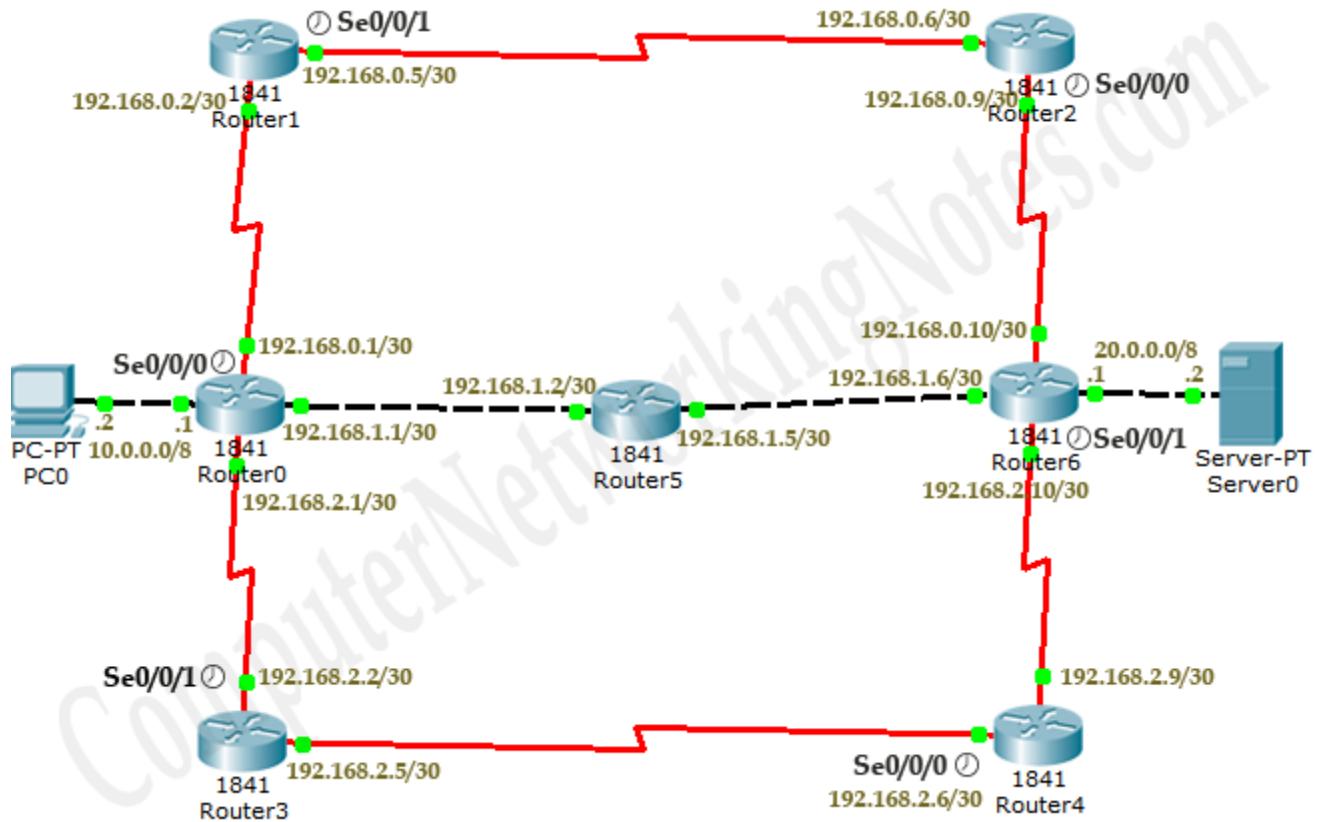


Configuration of OSPF Routing Protocol

Create a topology as illustrate in following figure.

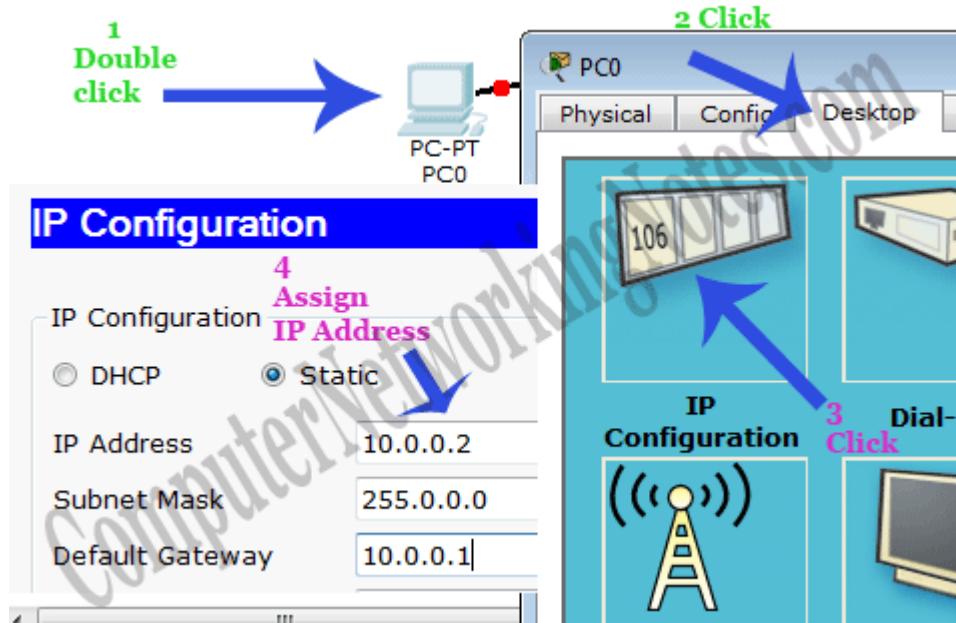


Device	Interface	IP Configuration	Connected with
PC0	Fa0/0	10.0.0.2/8	Router0's Fa0/0
Router0	Fa0/0	10.0.0.1/8	PC0's Fa0/0
Router0	Fa0/1	192.168.1.1/30	Router5's Fa0/1
Router5	Fa0/1	192.168.1.2/30	Router0's Fa0/1
Router5	Fa0/0	192.168.1.5/30	Router6's Fa0/0
Router6	Fa0/0	192.168.1.6/30	Router5's Fa0/0
Router6	Fa0/1	20.0.0.1/8	Server0's Fa0/0
Server0	Fa0/0	20.0.0.2/8	Router6's Fa0/1
Router0	Serial 0/0/0 (DCE)	192.168.0.1/30	Router1's Se0/0/0
Router1	Serial 0/0/0	192.168.0.2/30	Router0's Se0/0/0
Router1	Serial 0/0/1 (DCE)	192.168.0.5/30	Router2's Se0/0/1
Router2	Serial 0/0/1	192.168.0.6/30	Router1's Se0/0/1
Router2	Serial 0/0/0 (DCE)	192.168.0.9/30	Router6's Se0/0/0
Router6	Serial 0/0/0	192.168.0.10/30	Router2's Se0/0/0
Router0	Serial 0/0/1	192.168.2.1/30	Router3's Se0/0/1
Router3	Serial 0/0/1 (DCE)	192.168.2.2/30	Router0's Se0/0/1

Router3	Serial 0/0/0	192.168.2.5/30	Router4's Se0/0/0
Router4	Serial 0/0/0 (DCE)	192.68.2.6/30	Router3's Se0/0/0
Router4	Serial 0/0/1	192.168.2.9/30	Router6's Se0/0/1
Router6	Serial0/0/1 (DCE)	192.168.2.10/30	Router4's Se0/0/1

Assign IP address to PC

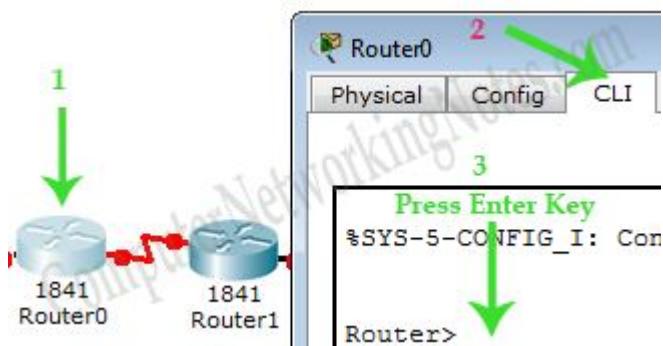
Double click **PC0** and click **Desktop** menu item and click **IP Configuration**. Assign IP address **10.0.0.2/8** to **PC0**.



Repeat same process for Server0 and assign IP address 20.0.0.2/8.

Assign IP address to interfaces of routers

Double click **Router0** and click **CLI** and press Enter key to access the command prompt of **Router0**.



Four interfaces FastEthernet0/0, FastEthernet0/1, Serial 0/0/0 and Serial0/0/1 of Router0 are used in this topology. By default interfaces on router are remain administratively down during the start up.

We need to configure IP address and other parameters on interfaces before we could actually use them for routing. Interface mode is used to assign the IP address and other parameters. Interface mode can be accessed from global configuration mode. Following commands are used to access the global configuration mode.

```
Router>enable  
Router# configure terminal  
Enter configuration commands, one per line. End with CNTL/Z.  
Router(config)#
```

From global configuration mode we can enter in interface mode. From there we can configure the interface. Following commands will assign IP address on FastEthernet0/0 and FastEthernet0/1.

```
Router(config)#interface fastEthernet 0/0  
Router(config-if)#ip address 10.0.0.1 255.0.0.0  
Router(config-if)#no shutdown  
Router(config-if)#exit  
Router(config)#interface fastEthernet 0/1  
Router(config-if)#ip address 192.168.1.1 255.255.255.252  
Router(config-if)#no shutdown  
Router(config-if)#exit  
Router(config)#
```

interface fastEthernet 0/0 command is used to enter in interface mode.

ip address 10.0.0.1 255.0.0.0 command would assign IP address to interface.

no shutdown command would bring the interface up.

exit command is used to return in global configuration mode.

Serial interface needs two additional parameters **clock rate** and **bandwidth**. Every serial cable has two ends DTE and DCE. These parameters are always configured at DCE end.

We can use **show controllers interface** command from privilege mode to check the cable's end.

```
Router#show controllers serial 0/0/0
Interface Serial0/0/0
Hardware is PowerQUICC MPC860
DCE V.35, clock rate 2000000
[Output omitted]
```

Fourth line of output confirms that DCE end of serial cable is attached. If you see DTE here instead of DCE skip these parameters.

Now we have necessary information let's assign IP address to serial interfaces.

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 0/0/0
Router(config-if)#ip address 192.168.0.1 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#bandwidth 64
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 192.168.2.1 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
```

Router#configure terminal Command is used to enter in global configuration mode.

Router(config)#interface serial 0/0/0 Command is used to enter in interface mode.

Router(config-if)#ip address 192.168.0.1 255.255.255.252 Command assigns IP address to interface. For serial link we usually use IP address from /30 subnet.

Router(config-if)#clock rate 64000 In real life environment this parameter controls the data flow between serial links and need to be set at service provider's end. In lab environment we need not to worry about this value. We can use any valid clock rate here.

Router(config-if)#bandwidth 64 Bandwidth works as an influencer. It is used to influence the metric calculation of OSPF or any other routing protocol which uses bandwidth parameter in

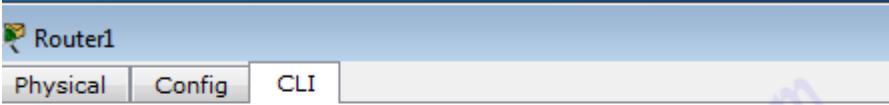
route selection process. Serial interface has default bandwidth of 1544Kbps. To explain, how bandwidth influence route selection process we will configure (64Kbps) bandwidth on three serial DCE interfaces of our network; R0's Se0/0/0, R1's Se0/0/1 and R2's Se0/0/0.

Router(config-if)#no shutdown Command brings interface up.

Router(config-if)#exit Command is used to return in global configuration mode.

We will use same commands to assign IP addresses on interfaces of remaining routers.

Router1



The screenshot shows the Cisco Router1 configuration interface. At the top, there is a toolbar with icons for Physical, Config (which is selected), and CLI. Below the toolbar, the title "IOS Command Line Interface" is displayed. The main area contains the command-line session:

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 0/0/0
Router(config-if)#ip address 192.168.0.2 255.255.255.252
Router(config-if)#exit
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 192.168.0.5 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#bandwidth 64
Router(config-if)#no shutdown
Router(config-if)#exit
```

Router2

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 0/0/0
Router(config-if)#ip address 192.168.0.9 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#bandwidth 64
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config-if)#
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 192.168.0.6 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
Router#
```

As I mention earlier, serial interface has a default bandwidth of 1544Kbps. If we don't assign any custom bandwidth, router would use default bandwidth. To see this feature in action we will not assign bandwidth on remaining routers.

Router6

Physical

Config

21

IOS Command Line Interface

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 0/0/0
Router(config if)#ip address 192.168.0.10 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 192.168.2.10 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 192.168.1.6 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface fastethernet 0/1
Router(config-if)#ip address 20.0.0.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
```

Router5

 Router5

Physical

Config

CLI

IOS Command Line Interface

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 192.168.1.5 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface fastethernet 0/1
Router(config-if)#ip address 192.168.1.2 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#[
```

Router3

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 0/0/0
Router(config-if)#ip address 192.168.2.5 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 192.168.2.2 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#

```

Router4

Physical | Config | CLI

IOS Command Line Interface

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface serial 0/0/0
Router(config-if)#ip address 192.168.0.10 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 192.168.2.10 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface fastethernet 0/0
Router(config-if)#ip address 192.168.1.6 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface fastethernet 0/1
Router(config-if)#ip address 20.0.0.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit

```

Great job we have finished our half journey. Now routers have information about the networks that they have on their own interfaces. Routers will not exchange this information between them on their own. We need to implement OSPF routing protocol that will insist them to share this information.

To be on same track I have uploaded my practice topology. Use this if you want to skip above IP configuration part.

Configure OSPF routing protocol

Enabling OSPF is a two steps process:-

- Enable OSPF routing protocol from global configuration mode.
- Tell OSPF which interfaces we want to include.

For these steps following commands are used respectively.

```
Router(config)# router ospf process_ID  
Router(config-router)# network IP_network_# [wild card mask] Area Number area number
```

Router(config)# router ospf process ID

This command will enable OSPF routing protocol in router. Process ID is a positive integer. We can use any number from 1 to 65,535. Process ID is locally significant. We can run multiple OSPF process on same router. Process ID is used to differentiate between them. Process ID need not to match on all routers.

Router(config-router)# network IP_network_# [wildcard_mask] area [area number]

Network command allows us to specify the interfaces which we want to include in OSPF process. This command accepts three arguments network number, wildcard mask and area number.

Network number

Network number is network ID. We can use any particular host IP address or network IP address. For example we can use 192.168.1.1 (host IP address) or we can use 192.168.1.0 (Network IP address). While targeting a specific interface usually we use host IP address (configured on that interface).

While targeting multiple interfaces, we use network IP address. So any interface that belongs to specified network ID will be selected.

Wildcard mask

Wildcard mask are used with network ID to filter the interfaces. Wildcard mask is different from subnet mask. Subnet mask is used to separate the network portion and host portion in IP address. While wildcard mask is used to match corresponding octet in network portion. Wildcard mask tells OSPF the part of network address that must be matched. Wildcard masks are explained with examples in access list tutorials of this category.

Key points

0 (Decimal – octet format) Wildcard mask indicates that corresponding octet in network address must be matched exactly.

255 (Decimal – octet format) Wildcard mask indicates that we don't care about corresponding octet in network address.

For example

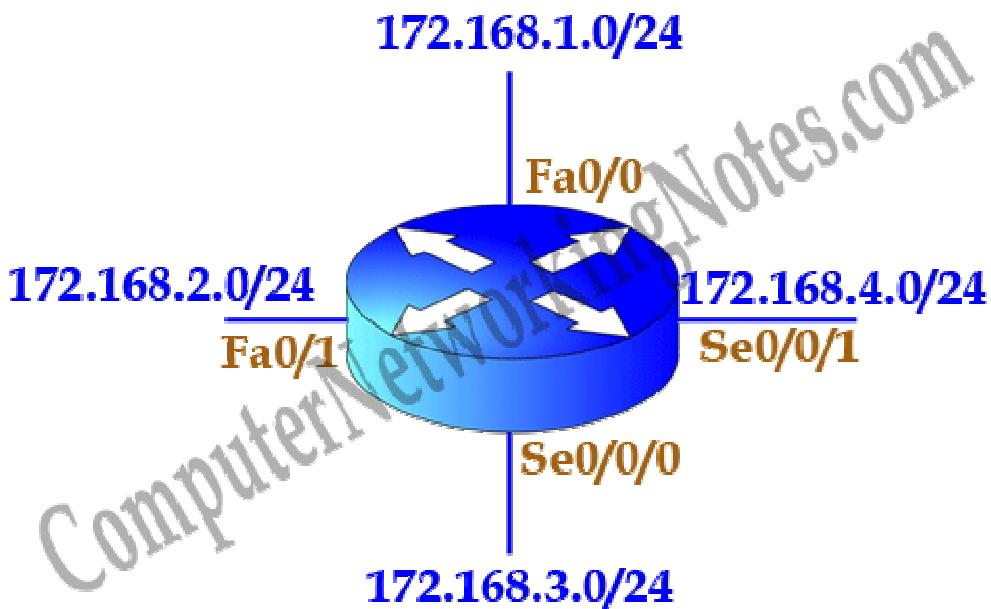
10. 10. 0. 0	0. 0. 255. 255	Valid match examples 10.10.0.1, 10.10.10.10, 10.10.253.253 Invalid match examples 10.0.0.1, 1.10.10.10.10, 10.1.253.253
Exact match	Ignore Everything	

0 (Binary – bit format) Wildcard mask indicates that corresponding bit in network address must be matched exactly.

255 (Binary – bit format) Wildcard mask indicates that we don't care about corresponding bit in network address.

	<i>Match</i>	<i>Ignore</i>	
192.168.0.0	11000000.10101000.00000000.00000000	00000000.00000000.00000000.11111111	Any interface configured with host address between 192.168.0.0 and 192.168.0.255 will be selected.
192.168.0.x	11000000.10101000.00000000.XXXX0000		

OSPF is a classless protocol. With wildcard we can also filter Subnetted networks. In classes implementation usually we use Subnetted networks. For example consider following figure



We have four networks 172.168.1.0/24, 172.168.2.0/24, 172.168.3.0/24 and 172.168.4.0/24 subnetted from single class B network 172.168.0.0/16. Classful configuration does not understand the concept of subnetting. In classful configuration all these networks belong to a single network. Classful configuration works only within default boundary of mask. Default boundary of this address is 16 bits. So a classful routing protocol will match only first 16 bits (172.168.x.y) of network address. A classful routing protocol such as RIP cannot distinguish between different Subnetted networks.

A classless routing protocol such as OSPF goes beyond the default boundary of mask and work well with Subnetted networks. With wildcard mask we can easily filter Subnetted networks.

With wildcard we are no longer limited with default boundaries. We can match Subnetted networks as well as default networks.

For example we want to exclude serial interfaces in above configuration. We can use a wildcard mask of 0.0.0.255 to match the subnet mask of /24.

```
Router(config-router)# network 172.168.1.0 0.0.0.255
Router(config-router)# network 172.168.2.0 0.0.0.255
```

Above commands will ask router to match /24 bits of address instead of default /16 bits.

Now router will look for 172.168.1.x and 172.168.2.x network. Our serial interfaces have 172.168.3.0/24 and 172.168.4.0/24 networks which do not fall in these search criteria.

Let's take one more example, if we use following network command, which interfaces would be selected.

```
Router(config-router)# network 192.168.0.0 0.0.0.3
```

Match *Ignore*

192.168.0.0/30 11000000.10101000.00000000.000000 00
 00000000.00000000.00000000.000000 11

 11000000.10101000.00000000.000000 XX

Any interface configured with host address between 192.168.0.0 and 192.168.0.3 will be selected. In this example valid host addresses are 192.168.0.1 and 192.168.0.2

In this case valid host IP addresses are 192.168.0.1 and 192.168.0.2. So any interface that has these IP address would be selected. /30 network is usually used for serial link connection which need only two valid host IP addresses; one for each end.

If you are unfamiliar with wildcard mask, I suggest you to check our tutorials on access lists configuration in this category. In those tutorials wildcard masks are explained in detail with examples.

For this tutorial let's move on third argument. Third argument which network command accept is area number. This parameter say router to put matched interface in specified area. OSPF areas are explained in second part this article.

Now we know the essential commands for configuration. Let's implement them in our network.

OSPF configuration

Router0

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 10.0.0.0 0.255.255.255 area 0
Router(config-router)#network 192.168.0.0 0.0.0.3 area 0
Router(config-router)#network 192.168.1.0 0.0.0.3 area 0
Router(config-router)#network 192.168.2.0 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#

```

Router1

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 10
Router(config-router)#network 192.168.0.0 0.0.0.3 area 0
Router(config-router)#network 192.168.0.4 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#

```

Router2

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 20
Router(config-router)#network 192.168.0.4 0.0.0.3 area 0
Router(config-router)#network 192.168.0.8 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#

```

Router6

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 60
Router(config-router)#network 20.0.0.0 0.255.255.255 area 0
Router(config-router)#network 192.168.0.8 0.0.0.3 area 0
Router(config-router)#network 192.168.2.8 0.0.0.3 area 0
Router(config-router)#network 192.168.1.4 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#

```

Router5

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 50
Router(config-router)#network 192.168.1.0 0.0.0.3 area 0
Router(config-router)#network 192.168.1.4 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#

```

Router4

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 40
Router(config-router)#network 192.168.2.8 0.0.0.3 area 0
Router(config-router)#network 192.168.2.4 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#

```

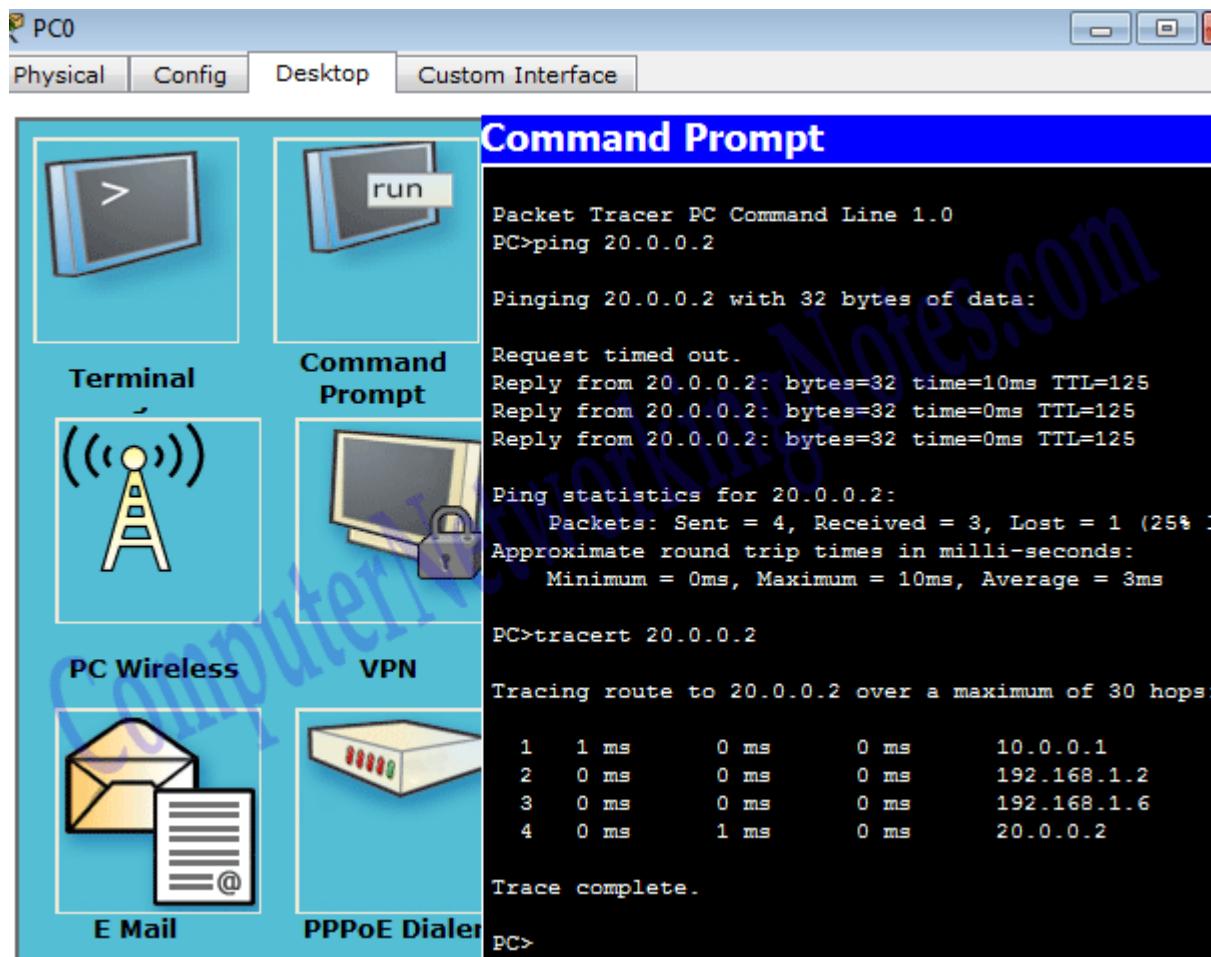
Router3

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router ospf 30
Router(config-router)#network 192.168.2.0 0.0.0.3 area 0
Router(config-router)#network 192.168.2.4 0.0.0.3 area 0
Router(config-router)#exit
Router(config)#

```

That's it. Our network is ready to take the advantage of OSPF routing. To verify the setup we will use **ping** command. ping command is used to test the connectivity between two devices. We have two routes between source and destination. tracert command is used to know the route which is used to get the destination.

Access the command prompt of PC1 and use ping command to test the connectivity from Server0. After that use **tracert** command to print the taken path.



Great! We have successfully implemented OSPF routing in our network. For cross check we have uploaded a configured topology on our server. You can use that if not getting same output.

Summary

Command	Description
Router(config)#router ospf 10	Enable OSPF routing protocol under process ID 10.
Router(config-router)#network 10.10.0.0 0.0.255.255 area 0	Enable OSPF with area 0 on matching interface.
Router(config)#interface loopback 0	Create a Loopback interface and move in sub interface configuration mode
Router(config-if)#ip address 192.168.250.250 255.255.255.0	Assign IP address to loopback interface.
Router(config-router)#router-id 1.1.1.1	Set 1.1.1.1 as router ID
Router(config)#interface serial 0/0	Inter in sub interface configuration mode

Router(config-if)#ip ospf priority 100	Used to influence DR/BDR selection process. Valid range is 0 to 255. 0 makes router ineligible for DR/BDR while 255 makes router guaranteed DR/BDR. Higher priority value means higher chance of becoming DR/BDR.
Router(config-if)#bandwidth 256	Used to influence route metric cost. Cost is the inverse of bandwidth. Higher bandwidth has lower cost. Bandwidth is defined in Kbps. 256 means 256 Kbps.
Router(config-if)#ip ospf hello-interval timer 15	Set hello interval timer to 15 seconds. Hello timer must be match on both routers in order become neighbors.
Router(config-if)#ip ospf dead-interval 60	Set dead interval timer to 60 seconds. Dead interval timer must be match on both routers in order to become neighbor
Router#show ip route	Display all routes from routing table
Router#show ip route ospf	Display all routers learned through OSPF from routing table
Router#show ip ospf	Display basic information about OSPF
Router#show ip ospf interface	Display information about all OSPF active interfaces
Router#show ip ospf interface serial 0/0/0	Display OSPF information about serial 0/0/0 interface
Router#show ip ospf neighbor List all	OSPF neighbors with basic info
Router#show ip ospf neighbor detail	List OSPF neighbors with detail info
Router#show ip ospf database	Display data for OSPF database
Router#clear ip route *	Clear all routes from routing table.
Router#clear ip route 10.0.0.0/8	Clear particular route from routing table
Router#clear ip ospf counters	Clear OSPF counters
Router#debug ip ospf events	Display all ospf events
Router#debug ip ospf packets	Display exchanged OSPF packets
Router#debug ip ospf adjacency	Display DR/BDR election process state