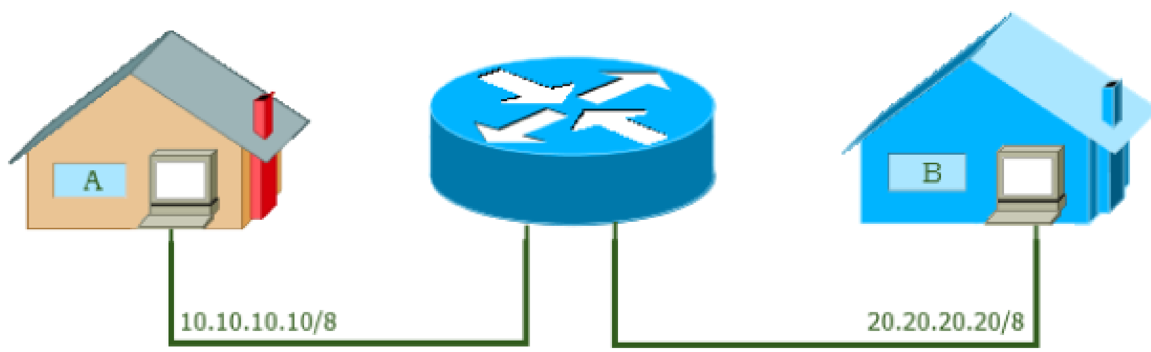
Lab 5 - RIP Conceptual Notes and Configuration Exercise – Dynamic Routing

Basic of IP Routing

What is IP routing?

IP routing is a process of transferring data from one network to another as IP packets. By default, hosts of different networks cannot communicate with each other. If two hosts located in different IP networks want to communicate with each other, they use IP routing. Routers provide IP routing. A router is a specialized device that connects different IP networks. Let's take a simple example. Suppose two IP hosts; 10.10.10.10/8 and 20.20.20.20/8 want to communicate. Since they both belong to the different IP networks, they need a router to communicate.

The following image shows this example.



The complete IP routing process relies on two types of protocols; routed protocols and routing protocols.

Routing protocols v/s Routed or Routable protocols

A routed protocol is used to encapsulate the data that is exchanged between the source host and the destination host. In IP routing, the **IP protocol** is used as the routed protocol. By using the IP protocol, a source host packs data pieces and adds the source address and the destination address on each data piece. A data piece with both addresses (source and destination) is known as the **IP packet**.

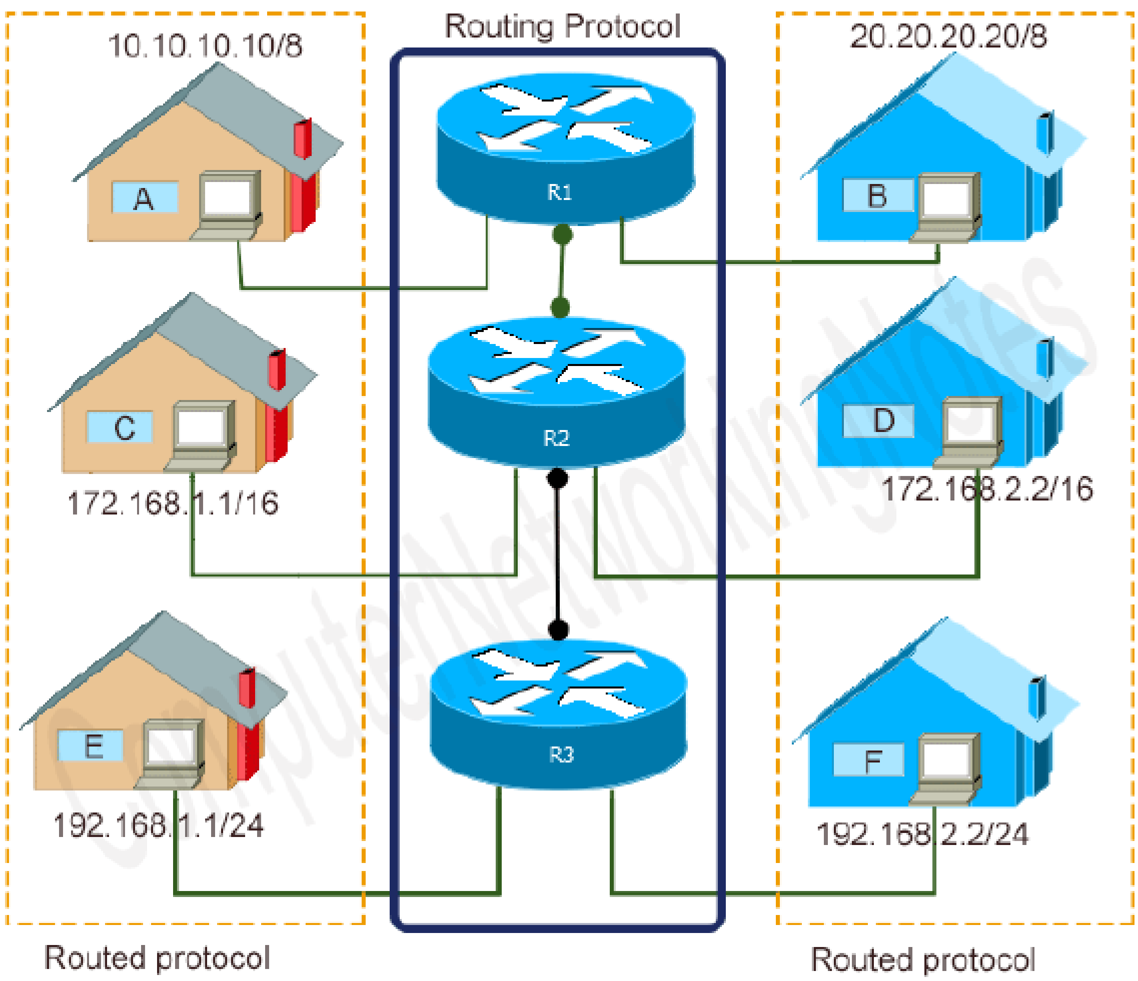
Any router that works on a path that connects the source host to the destination host uses both (source and destination) addresses to find out where the packet came from and where it will go.

Routing protocols

Routers use a routing protocol for the following purposes.

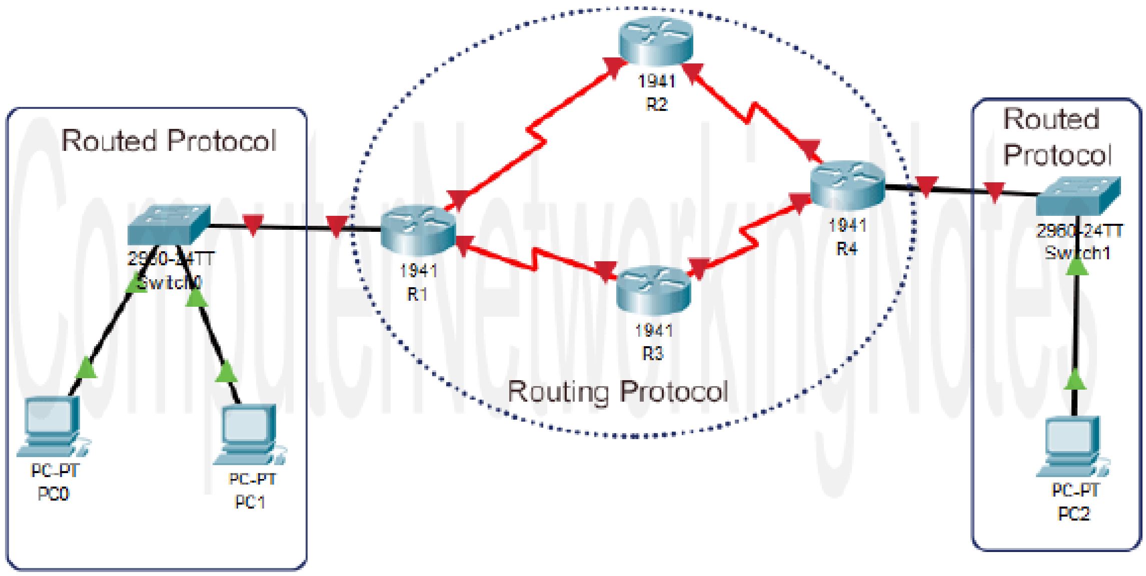
- To figure out all available paths of the network. A router stores these paths in a table known as the **routing table**.
- To select the best and fastest path to get a destination host. When a router receives an IP packet, the router checks its routing table and compares all available paths to get the destination network of the received IP packet and selects the fastest path from all available paths.

RIP, IGRP, EIGRP, and OSPF are examples of routing protocols.



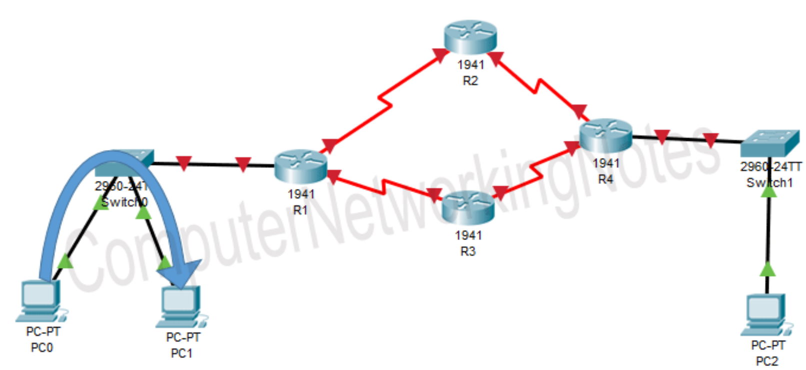
How does IP routing work?

The process of IP routing begins when a host creates a data packet for a host that is located in another network and ends when that destination host receives the packet. To understand this process in detail, let's take a simple example. The following image shows the layout of a network. In this network, PC0 and PC1 are connected to PC2 via four routers; R1, R2, R3, and R4.



Suppose, an application running on PC0 wants to send some data to PC1. The application calls the IP protocol of PC0 and hands that data over to the IP protocol. The IP protocol packs data into packets and adds source and destination addresses to each packet. After this, the IP protocol uses another protocol known as **ARP** protocol to figure out whether the destination address (PC1) is located in the local network (the same IP network) or is located in the remote network (another IP network). If the destination address is located in the same IP (local) network, the IP protocol sends packets directly to the destination host.

The entire routing process is controlled by the routed (IP) protocols of PC0 and PC1.



Now suppose that the same application wants to send data to PC2. The same process is repeated until the packet forwarding decision is made by the IP protocol. This time, since the destination host (PC2) is located in the remote network (another IP network), the IP protocol sends packets to the default gateway.

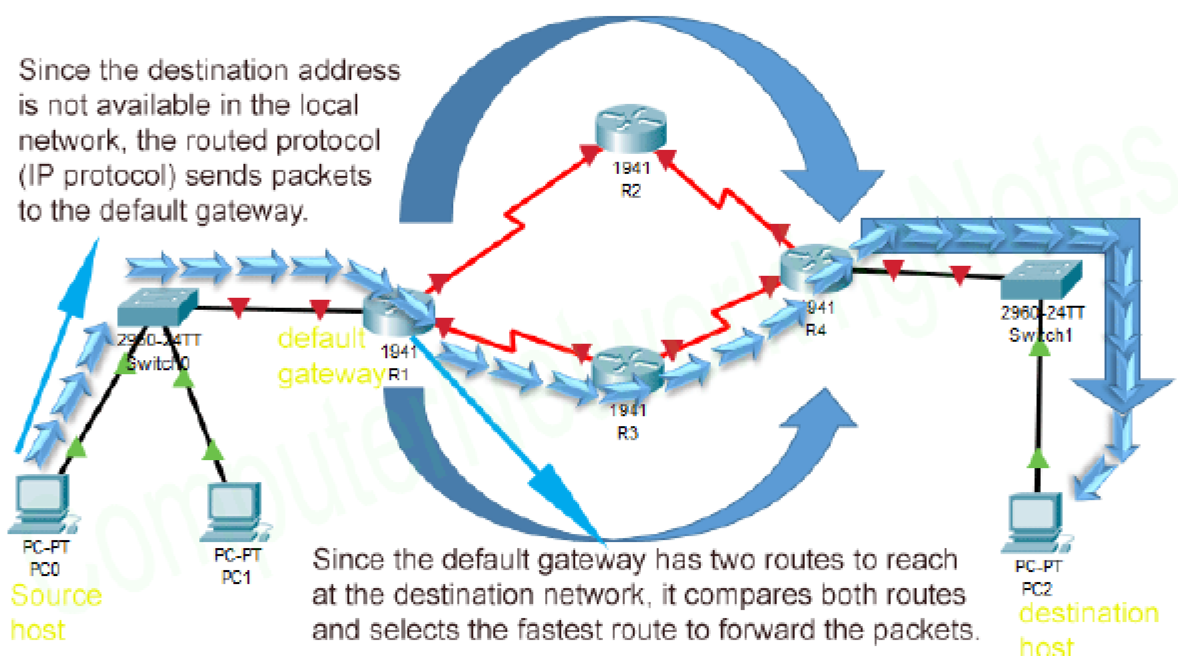
A **default gateway** is an interface of a router that connects the local network with the remote network. By default, an IP protocol forwards all packets to the default gateway except the packets that belong to the local IP network.

The default gateway router not only keeps records of all remote networks but also keeps records of all available paths for each remote network. A router maintains these records in the routing table. A typical routing table entry consists of two pieces; the network address and the interface on which that network is available.

When a router receives a packet on any of its interfaces, it reads the destination network of that packet and finds that network in the routing table. If the routing table contains a record for that network, the router uses that record to forward the packet. If the routing table doesn't contain a record for that network, the router discards that packet.

If multiple paths to a remote network exist, the router chooses the fastest path from them.

In the diagram shown below the default gateway router R1 has two paths to reach to PC2's network. When it receives packets for PC2 from PC0, it compares both paths and chooses the fastest path to forward packets.



PC2 receives packets from its default gateway router R4. The entire routing process is controlled by both types of protocols; routed and routing protocols.

Basic Routing Concepts and Protocols

Features of routing protocols

Routers use routing protocols: -

- To know all the available paths of the network
- To select the best and fastest path for each destination in the network
- To select a single and fastest path if more than one path exists for a single destination

Functions of routing protocols

The main functions of routing protocols are the following.

- Advertise local routing information to neighboring routers
- Calculate the best route for each subnet of the network
- Provide a virtual map of all routes of the network
- Calculate the cost of each route and help the router choose the best and fastest route
- Detect any change in the network and update all routers about that change

Types of routing protocols

There are three types of routing protocols: distance-vector, link-state, and hybrid. RIPv1 and IGRP are examples of distance-vector routing protocols while OSPF is an example of a link state routing protocol. Examples of hybrid routing protocols include RIPv2, EIGRP, and BGP.

Distance-vector routing protocols

Routers running distance-vector routing protocols periodically broadcast routing and reachability information from all active interfaces. They also receive the same information from their neighbors on their active interfaces.

Distance-vector protocols use timers to broadcast routing information. Once their periodic timer expires, they broadcast their routing information from all active interfaces, no matter whether the routing information has changed since the previous broadcast or not.

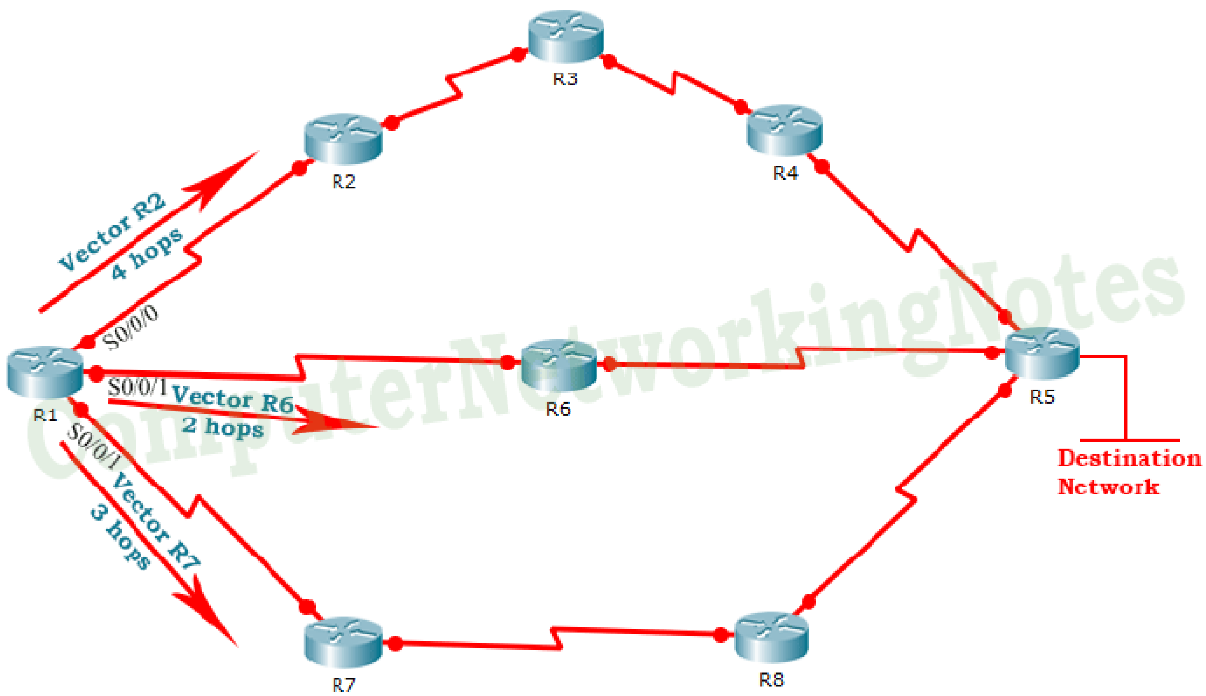
Calculating/selecting the best route

Distance-vector protocols use distance and direction to calculate and select the best route for each subnet of the network. Distance is the number of routers that a packet crosses to reach its destination.

Distance is measured in terms of hops. Each instance where a packet goes through a router is called a hop. For example, if a packet crosses four routers to reach its destination, the number of hops is 4. The route with the least number of hops is selected as the best route.

The vector indicates the direction that a packet uses to reach its destination.

The following figure shows an example of a network running distance vector protocol.



In this network, the router R1 has three routes to the destination network. These routes are the following.

1. The four-hop route (distance) through R2 (vector)

2. The two-hop route (distance) through R6 (vector)

3. The three-hop route (distance) through R7 (vector)

Since the second route has the lowest hop count, the router R1 uses this route to forward all packets of the destination network.

Key points: -

- Distance-vector protocols do not perform any mechanism to know who their neighbors are.
- Distance-vector protocols learn about their neighbors by receiving their broadcasts.
- Distance-vector protocols do not perform any formal handshake or hello process with neighbors before broadcasting routing information.
- Distance-vector protocols do not verify whether neighbors received routing updates or not.
- Distance-vector protocols assume that if a neighbor misses an update, it will learn about the change in the next broadcast update.

Link-state routing protocols

Unlike distance-vector routing protocols, the link-state routing protocols do not share routing and reachability information with anyone. Routers running link-state protocols share routing information only with neighbors. To discover neighbors, link-state protocols use a special protocol known as the hello protocol.

After discovering all neighbors, the link-state protocols create three separate tables. One of these tables keeps track of directly attached neighbors, one determines the topology of the entire internetwork, and one is used as the routing table.

From all available routes, to select the best route for each destination of the network, the link state protocols use an algorithm called the Shortest Path First (SPF) algorithm.

Differences between distance--vector routing protocols and link--state routing protocols

Unlike distance-vector routing protocols that broadcast the entire routing table periodically whether there are any changes or not, link-state routing protocols do not exchange routing information periodically. They exchange information only when they detect any change in the network.

Distance-vector protocols use local broadcasts, which are processed by every router on the same segment, while link-state protocols use multicasts which are processed only by the routers running the link-state protocol.

Distance-vector protocols do not verify routing broadcasts. They don't care whether the neighboring routers received them or not. Link-state protocols verify routing updates. A destination router, when receiving a routing update, will respond to the source router with an acknowledgment.

Hybrid routing protocols

Hybrid routing protocols are the combination of both distance-vector and link-state protocols. Hybrid routing protocols are based on distance-vector routing protocols but contain many of the features and functions of link-state routing protocols.

Hybrid routing protocols are built upon the basic principles of a distance-vector protocol but act like a link-state routing protocol. Hybrid protocols use a Hello protocol to discover neighbors and form neighbor relationships. Hybrid protocols also send updates only when a change occurs.

Hybrid routing protocols reduce the CPU and memory overhead by functioning like a distance vector protocol when it comes to processing routing updates; but instead of sending out periodic updates like a distance-vector protocol, hybrid routing protocols send out incremental, reliable updates via multicast messages, providing a more network- and router friendly environment.

How RIP Routing Protocol Works

What is the RIP routing protocol?

When an IP packet arrives on an interface of the router, the router reads the destination address of the IP packet and searches the destination address in the routing table. A routing table entry contains two important pieces of information: the destination subnet and the local interface that is connected with that destination.

If the router finds an entry for the destination address in the routing table, the router forwards the incoming packet from the interface that is associated with the destination address in the entry. If the router does not find an entry for the destination address in the routing table, it immediately discards the incoming packet.

There are two ways to add entries in the routing table: manual and dynamic. In the manual method, we manually add entries for all network paths in the routing table. In dynamic routing, we configure and activate a routing protocol and the routing protocol automatically discovers all network paths and adds them to the routing table.

RIP (Routing Information Protocol) is a dynamic routing protocol. Once configured and activated, it not only automatically discovers all network paths but also adds them to the routing table.

How does RIP routing protocol work?

RIP requires information about locally available networks. On the first step, we add this information and activate the RIP routing protocol on routers of the network. Once configured and activated, each router sends the routing update out of all active interfaces every 30 seconds.

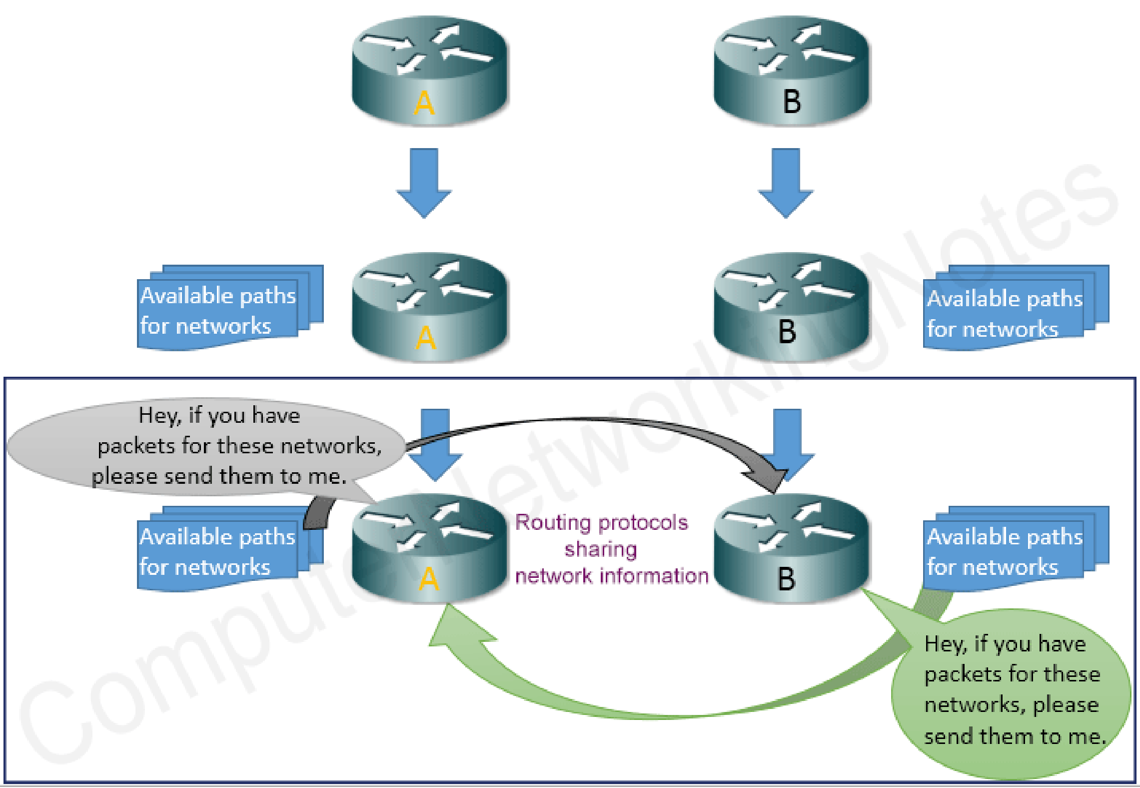
Each router also receives routing updates from its neighboring routers. A routing update contains the entire routing table of the sending router. Routers compare the received routing tables with their routing tables. If they find any new route in the received routing tables, they add them to their routing tables.

In the next routing update, routers advertise the updated routing tables. Over time, as each router learns more routes, they advertise about those routes as well. By the end of the process, all routers know about all routes.

Let's understand this process in detail through a simple example.

In a network, two routers: **A** and **B** are connected. An administrator configures the RIP routing protocol on both routers. After configuration, the RIP routing protocol of both routers automatically exchanges the information of locally available networks.

The following image shows this process.



If RIP detects any change in locally available networks' information, it updates the other router about this change in the next update. This way, an administrator only needs to provide information about locally available networks once. After that, the RIP protocol automatically manages all changes in the network.

RIP Routing broadcasts

To share the paths' information, the RIP protocol uses broadcast messages. RIP protocol periodically reads the routing table and shares it with neighbors through a broadcast message. Upon receiving a broadcast message from a neighbor, the RIP protocol reads the broadcast message and updates the routing table accordingly.

For example, if the broadcast message contains information about a new path, the RIP protocol adds that path in the routing table or if the broadcast message contains information that an existing path has gone down, the RIP protocol removes that path from the routing table or marks that path unusable in the routing table.

When a router running RIP protocol broadcasts the routing table, it not only broadcasts the information about the locally connected networks but also broadcasts the information about the networks that it has learned from its neighbors through the previously received broadcasts.

This update sequence eventually allows all routers to learn all paths. Let's understand this process through an example. Suppose, in a network, four routers: A, B, C, and D are connected in a sequence. All four routers are using the RIP routing protocol. Networks 10.0.0.0/8, 20.0.0.0/8, 30.0.0.0/8, and 40.0.0.0/8 are locally connected to the routers A, B, C, and D respectively.

The routing update sequence goes in the following way.

Router **A** broadcasts information of the network **10.0.0.0/8** to Router **B**.

Router **B** broadcasts information of the network **10.0.0.0/8** to Router **A** and **C**.

Router **C** broadcasts information of the network **30.0.0.0/8** to Router **B** and **D**.

Router **D** broadcasts information of the network **40.0.0.0/8** to Router **C**.

All routers after receiving broadcast update their routing tables, respectively.

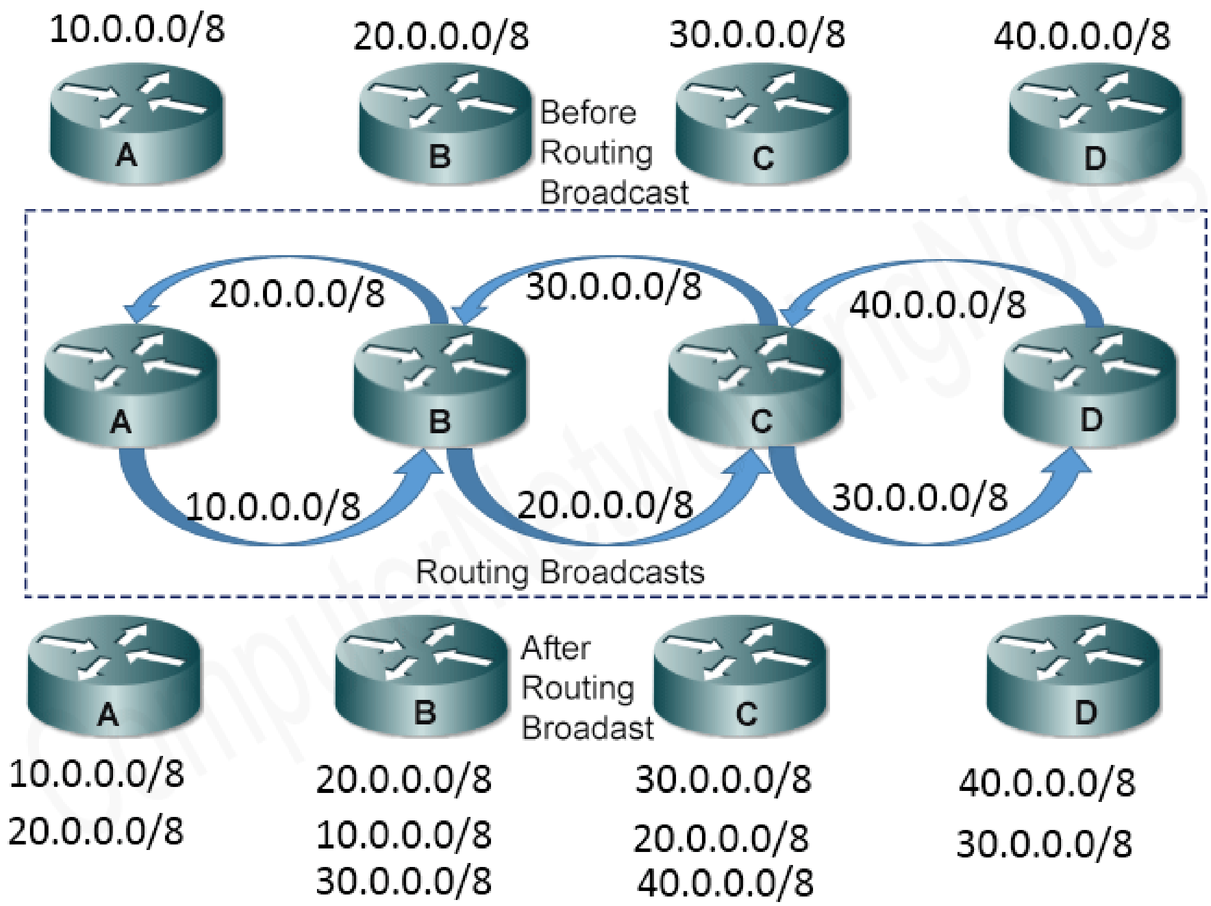
Router **A** adds an entry in the routing table that indicates the network **20.0.0.0/8** is reachable through Router **B**.

Router **B** adds an entry in the routing table that indicates the networks **10.0.0.0/8** and **30.0.0.0/8** are reachable through Router **A** and **C**, respectively.

Router **C** adds an entry in the routing table that indicates the networks **40.0.0.0/8** and **20.0.0.0/8** are reachable through Router **B** and **D**, respectively.

Router **D** adds an entry in the routing table that indicates the network **30.0.0.0/8** is reachable through Router **C**.

The following image shows this process.



After the next routing update:-

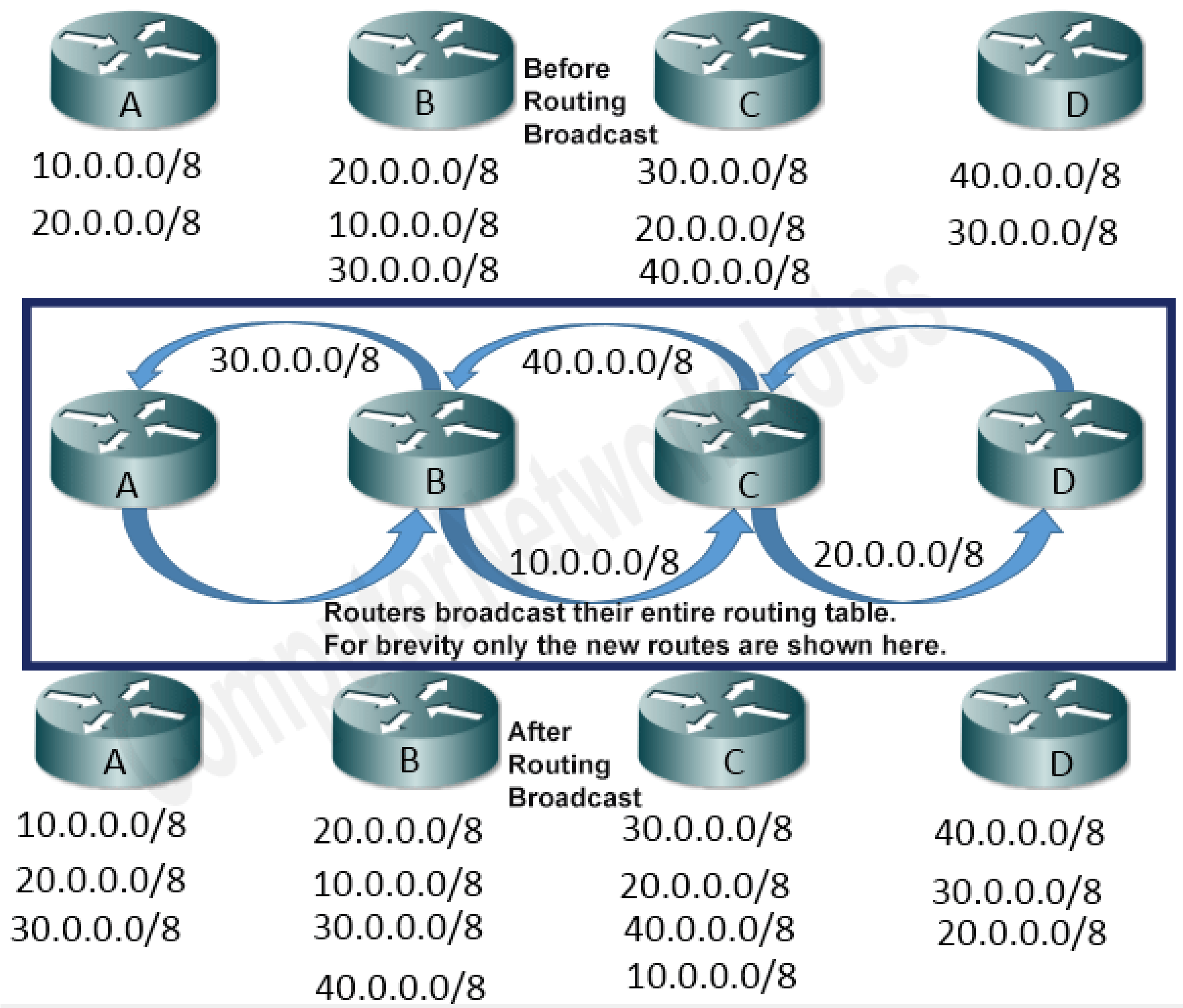
Router **A** adds an entry in the routing table that indicates the network **30.0.0.0/8** is reachable through Router **B**.

Router **B** adds an entry in the routing table that indicates the network **40.0.0.0/8** is reachable through Router **C**.

Router **C** adds an entry in the routing table that indicates the network **10.0.0.0/8** is reachable through Router **B**.

Router **D** adds an entry in the routing table that indicates the network **20.0.0.0/8** is reachable through Router **C**.

The following image shows routing tables before and after the second routing broadcast.

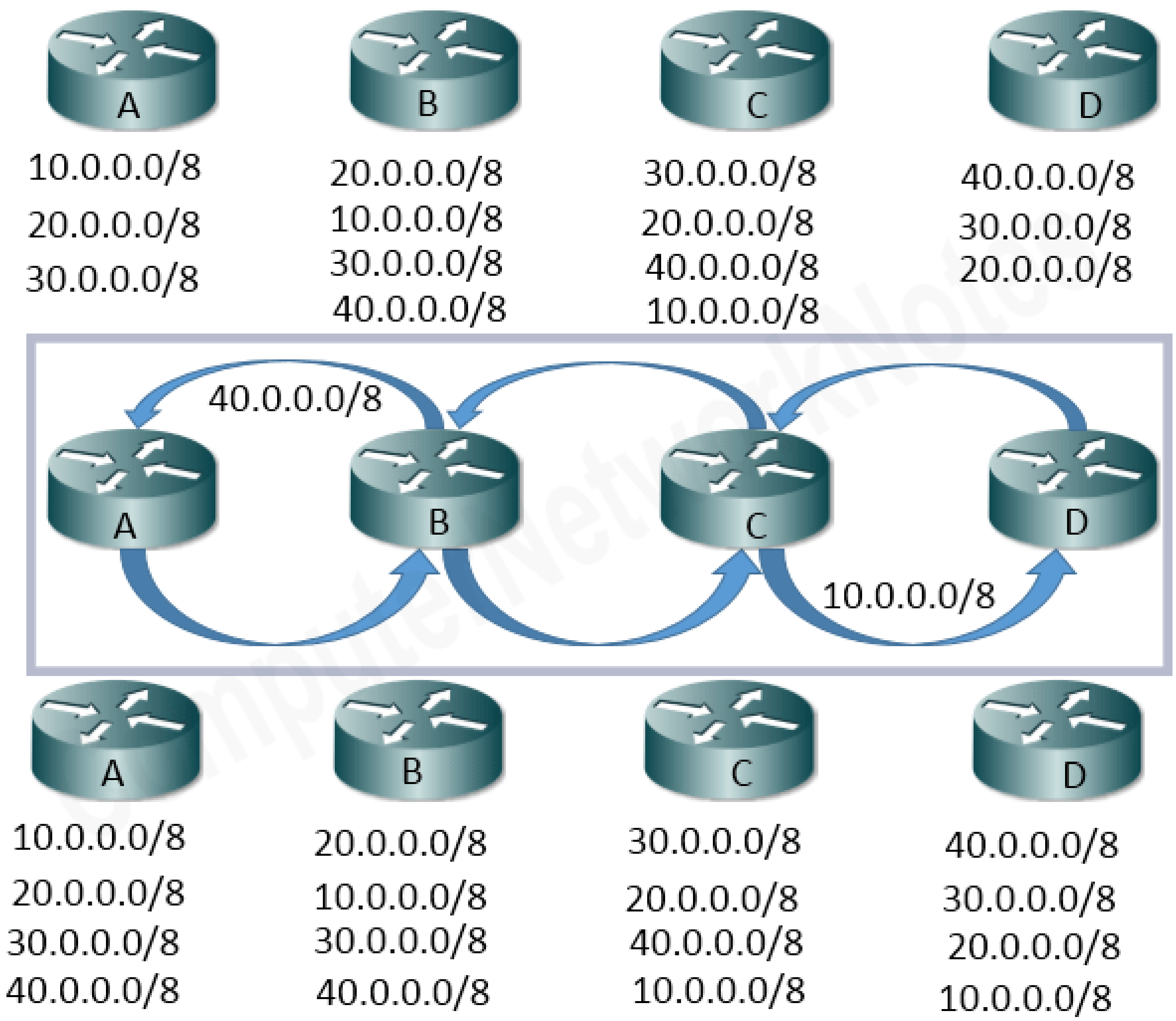


After the next routing update:-

Router **A** adds an entry in the routing table that indicates the network **40.0.0.0/8** is reachable through Router **B**.

Router **D** adds an entry in the routing table that indicates the network **10.0.0.0/8** is reachable through Router **C**.

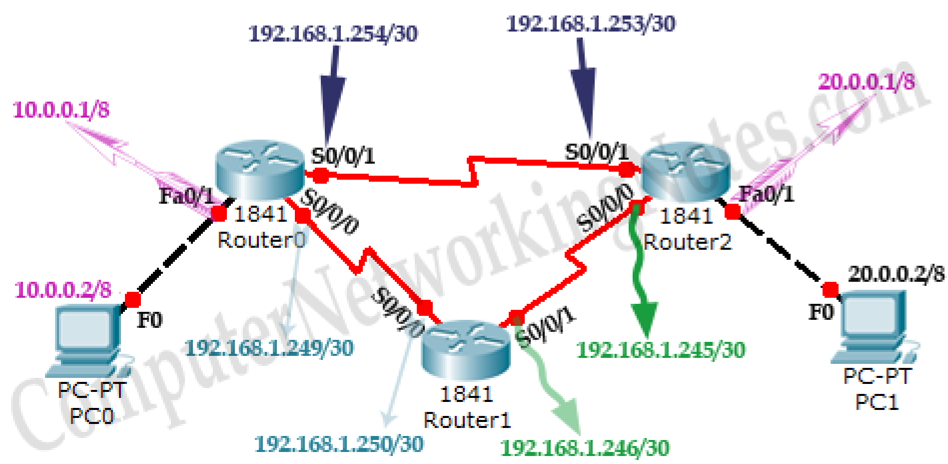
The following image shows routing tables before and after the third routing broadcast.



The situation in which all routers know all paths of the network is called **convergence**. After the convergence, the RIP routing protocol actively monitors all paths. If it detects any change in any path, it updates neighboring routers about that change in the next broadcast.

RIP Protocol Configuration

Create this topology in packet tracer

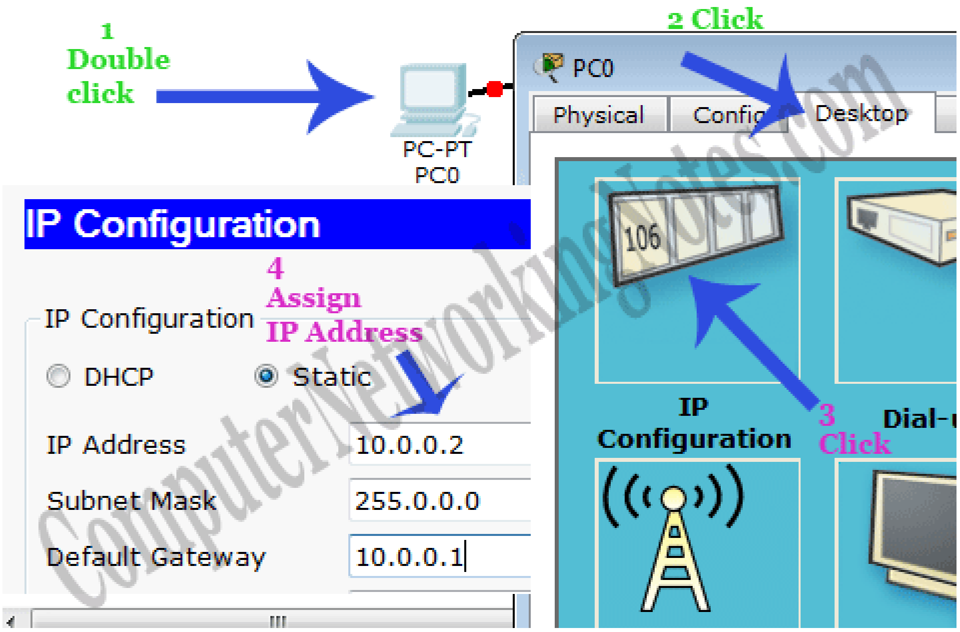


Initial IP configuration

Device	Interface	IP Configuration	Connected with
PC0	Fast Ethernet	10.0.0.2/8	Router0's Fa0/1
Router0	Fa0/1	10.0.0.1/8	PC0's Fast Ethernet
Router0	S0/0/1	192.168.1.254/30	Router2's S0/0/1
Router0	S0/0/0	192.168.1.249/30	Router1's S0/0/0
Router1	S0/0/0	192.168.1.250/30	Router0's S0/0/0
Router1	S0/0/1	192.168.1.246/30	Router2's S0/0/0
Router2	S0/0/0	192.168.1.245/30	Router1's S0/0/1
Router2	S0/0/1	192.168.1.253/30	Router0's S0/0/1
Router2	Fa0/1	20.0.0.1/30	PC1's Fast Ethernet
PC1	Fast Ethernet	20.0.0.2/30	Router2's Fa0/1

Assign IP address to PCs

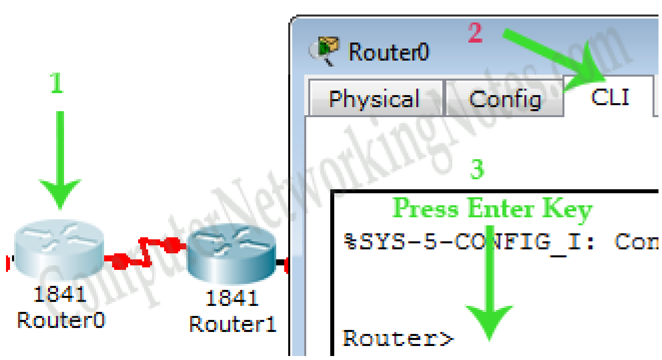
Double click **PC0** and click **Desktop** menu item and click **IP Configuration**. Assign IP address 10.0.0.2/8 to PC0.



Repeat same process for PC1 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**.



Three interfaces *FastEthernet0/0*, *Serial0/0/0* 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 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.

```
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 0/0*** command is used to enter in interface mode.

ip address ***10.0.0.1 255.0.0.0*** command will assign IP address to interface.

no shutdown command will 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 interface.

```
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 192.168.1.249 255.255.255.252
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 192.168.1.254 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#bandwidth 64
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#
```

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 ***192.168.1.249 255.255.255.252*** Command assigns IP address to interface. For serial link we usually use IP address from /30 subnet.

Router(config-if)#clock rate ***64000*** And **Router(config-if)#bandwidth** ***64*** In real life environment these parameters control the data flow between serial links and need to be set at service providers end. In lab environment we need not to worry about these values. We can use these values.

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. We need to provided clock rate and bandwidth only on DCE side of serial interface. Following command will assign IP addresses on interface of Router1.

Router1

```
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 192.168.1.250 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 192.168.1.246 255.255.255.252
Router(config-if)#clock rate 64000
Router(config-if)#bandwidth 64
Router(config-if)#no shutdown
Router(config-if)#exit
```

Use same commands to assign IP addresses on interfaces of Router2.

Router2

```
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 20.0.0.1 255.0.0.0
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial 0/0/0
Router(config-if)#ip address 192.168.1.245 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#interface serial 0/0/1
Router(config-if)#ip address 192.168.1.253 255.255.255.252
Router(config-if)#no shutdown
Router(config-if)#exit
```

Configure RIP routing protocol

Configuration of RIP protocol is much easier than you think. It requires only two steps to configure the RIP routing.

- Enable RIP routing protocol from global configuration mode.
- Tell RIP routing protocol which networks you want to advertise.

Let's configure it in Router0

Router0

```
Router0(config)#router rip
Router0(config-router)# network 10.0.0.0
Router0(config-router)# network 192.168.1.252
Router0(config-router)# network 192.168.1.248
```

router rip command tell router to enable the RIP routing protocol.

network command allows us to specify the networks which we want to advertise. We only need to specify the networks which are directly connected with the router.

That's all we need to configure the RIP. Follow same steps on remaining routers.

Router1

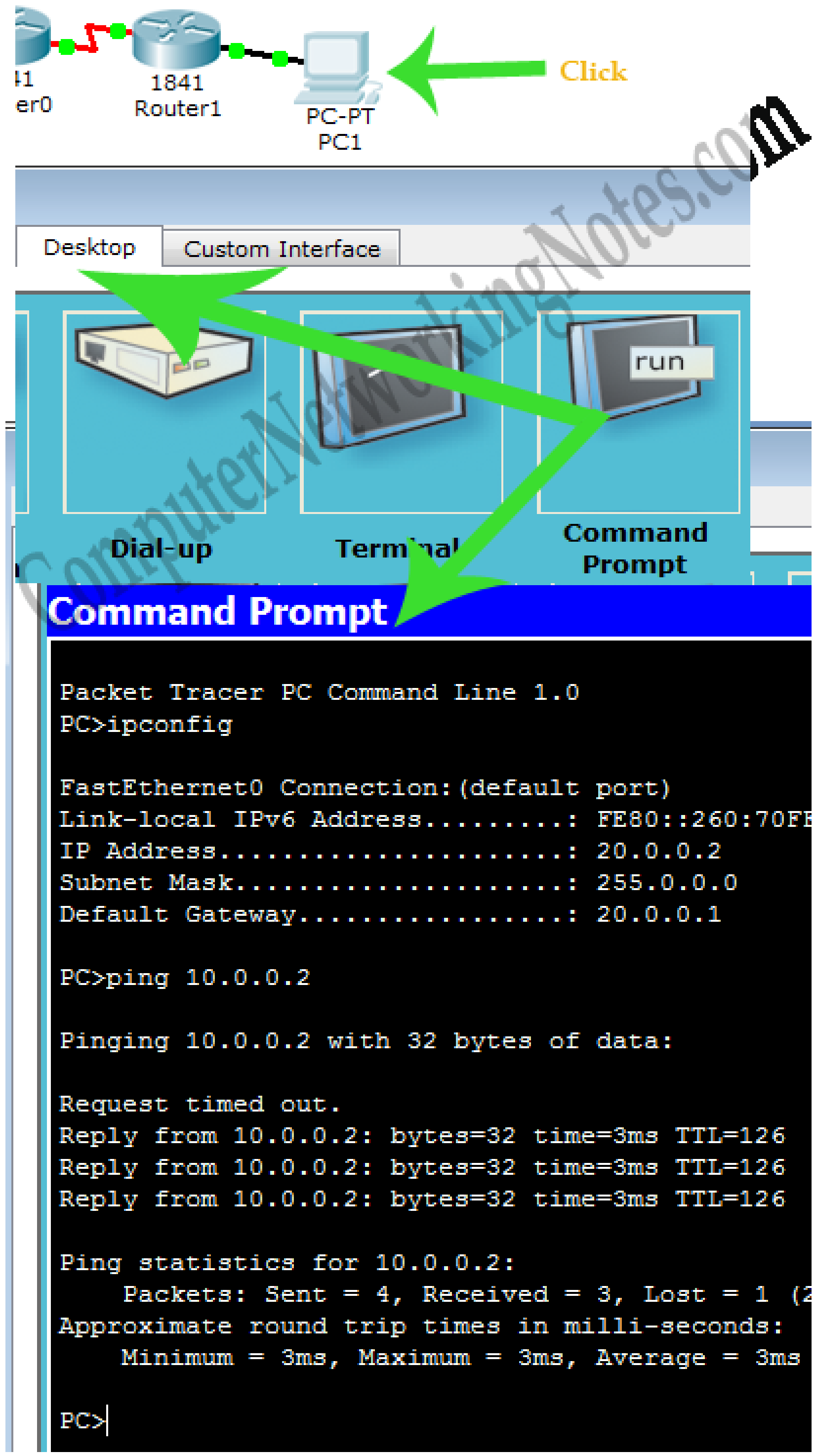
```
Router1(config)#router rip
Router1(config-router)# network 192.168.1.244
Router1(config-router)# network 192.168.1.248
```

Router2

```
Router2(config)#router rip
Router2(config-router)# network 20.0.0.0
Router2(config-router)# network 192.168.1.252
Router2(config-router)# network 192.168.1.244
```

Our network is ready to take the advantage of RIP routing. To verify the setup we will use ping command. ping command is used to test the connectivity between two devices.

Access the command prompt of **PC1** and use *ping* command to test the connectivity from **PC0**.



RIP protocol automatically manage all routes for us. If one route goes down, it automatically switches to another available. To explain this process more clearly we have added one more route in our network.

Currently there are two routes between PC0 and PC1.

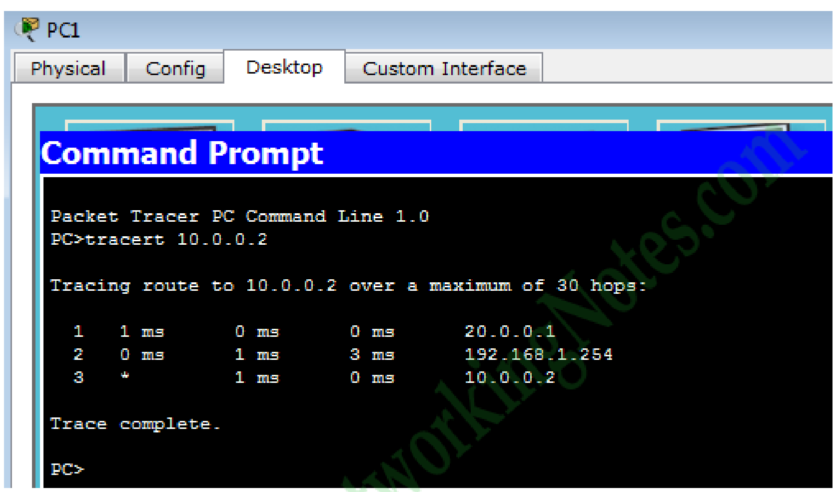
Route 1

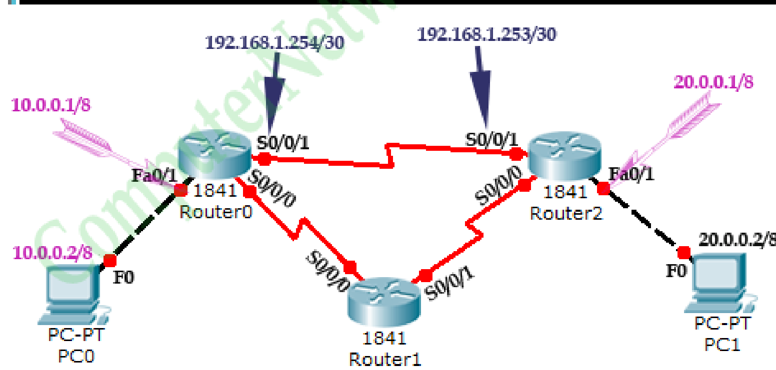
PC0 [Source / destination – 10.0.0.2] <==> Router0 [FastEthernet0/1 – 10.0.0.1] <==> Router0 [Serial0/0/1 – 192.168.1.254] <==> Router2 [Serial 0/0/1 – 192.168.1.253] <==> Router2 [FastEthernet0/0 – 20.0.0.1] <==> PC1 [Destination /source – 20.0.0.2]

Route 2

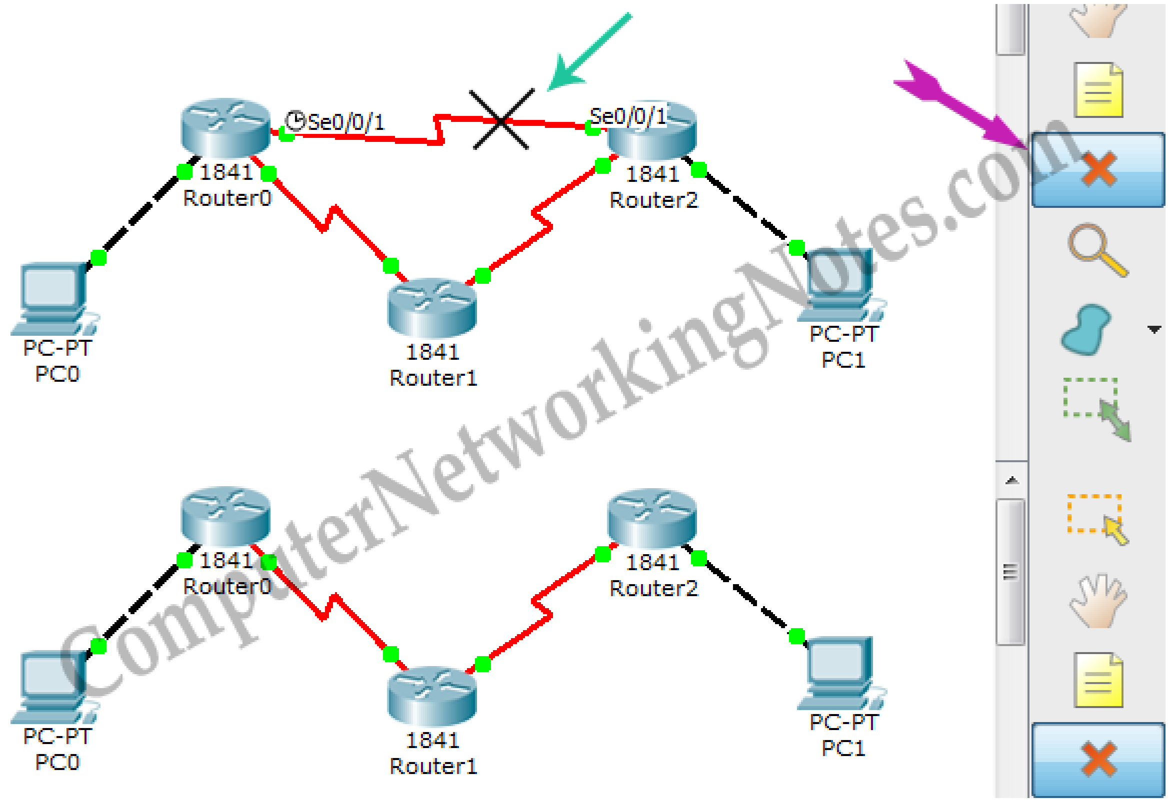
PC0 [Source / destination – 10.0.0.2] <==> Router0 [FastEthernet0/1 – 10.0.0.1] <==> Router0 [Serial0/0/0 – 192.168.1.249] <==> Router1 [Serial 0/0/0 – 192.168.1.250] <==> Router1 [Serial 0/0/1 – 192.168.1.246] <==> Router2 [Serial 0/0/0 – 192.168.1.245] <==> Router2 [FastEthernet0/0 – 20.0.0.1] <==> PC1 [Destination /source – 20.0.0.2]

By default RIP will use the route that has low hops counts between source and destination. In our network route1 has low hops counts, so it will be selected. We can use tracert command to verify it.



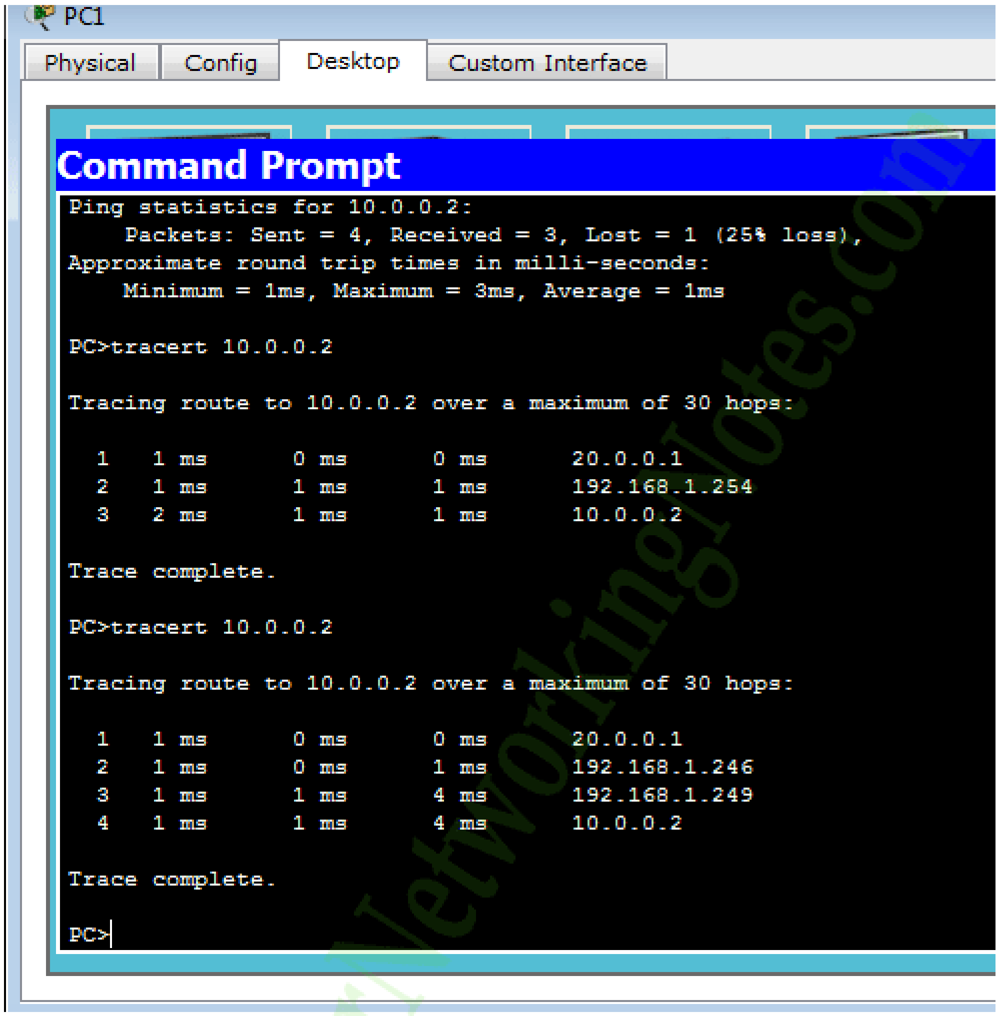


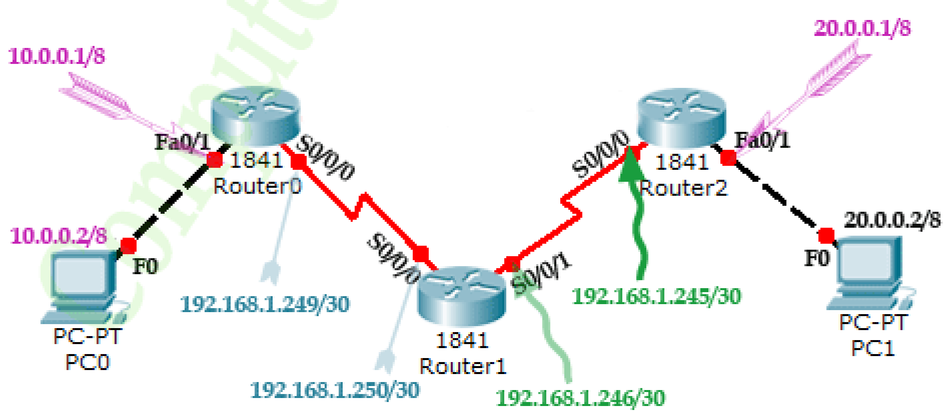
Now suppose route1 is down. We can simulate this situation by removing the cable attached between **Router0 [s0/0/1]** and **Router2 [s0/0/1]**



Okay our primary route went down. What will be happen now?

So far we are running RIP routing protocol and have another route to destination, there is no need to worry. RIP will automatically reroute the traffic. Use ***tracert*** command again to see the magic of dynamic routing.


PC1
Physical
Config
Desktop
Custom Interface
Command Prompt
Ping statistics for 10.0.0.2:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 1ms, Maximum = 3ms, Average = 1ms
PC>tracert 10.0.0.2
Tracing route to 10.0.0.2 over a maximum of 30 hops:
1 1 ms 0 ms 0 ms 20.0.0.1
2 1 ms 1 ms 1 ms 192.168.1.254
3 2 ms 1 ms 1 ms 10.0.0.2
Trace complete.
PC>tracert 10.0.0.2
Tracing route to 10.0.0.2 over a maximum of 30 hops:
1 1 ms 0 ms 0 ms 20.0.0.1
2 1 ms 0 ms 1 ms 192.168.1.246
3 1 ms 1 ms 4 ms 192.168.1.249
4 1 ms 1 ms 4 ms 10.0.0.2
Trace complete.
PC>


10.0.0.1/8
Fa0/1
1841
Router0
S0/0/0
10.0.0.2/8
F0
PC-PT
PC0
192.168.1.249/30
S0/0/0
1841
Router1
S0/0/1
192.168.1.250/30
192.168.1.246/30
S0/0/0
1841
Router2
192.168.1.245/30
20.0.0.1/8
Fa0/1
20.0.0.2/8
F0
PC-PT
PC1

RIP Routing protocol configuration commands summary

Command	Description
Router(config)#router rip	Enable RIP routing protocol
Router(config-router)#network a.b.c.d	Add a.b.c.d network in RIP routing advertisement
Router(config-router)#no network a.b.c.d	Remove a.b.c.d network from RIP routing advertisement
Router(config-router)#version 1	Enable RIP routing protocol version one (default)
Router(config-router)#version 2	Enable RIP routing protocol version two
Router(config-router)#no auto-summary	By default RIPv2 automatically summarize networks in their default classful boundary. This command will turn it off.
Router(config-router)#passive-interface s0/0/0	RIP will not broadcast routing update from this interface
Router(config-router)#no ip split-horizon	Disable split horizon (Enable by default)
Router(config-router)#ip split-horizon	Enable spilt horizon
Router(config-router)#timers basic 30 90 180 270 360	Allow us to set RIP timer in seconds. 30 (routing update), 90 (invalid timer), 180 (Hold timer), 270 (Flush timer), 360 (sleep timer)
Router(config)#no router rip	Disable RIP routing protocol
Router#debug ip rip	Used for troubleshooting. Allow us to view all RIP related activity in real time.
Router#show ip rip database	Display RIP database including routes