LAB 8

Basics of RIP and its Implementation using Cicso Packet Tracer

The **Routing Information Protocol** (**RIP**) is one of the oldest distance-vector routing protocols which employ the hop count as a routing metric. RIP prevents routing loops by implementing limit on the number of hops allowed in a path from source to destination. The maximum number of hops allowed for RIP is 15, which limits the size of networks that RIP can support. A hop count of 16 is considered an infinite distance and the route is considered unreachable. RIP implements the split horizon, route poisoning and holddown mechanisms to prevent incorrect routing information from being propagated.

Originally, each RIP router transmitted full updates every 30 seconds. In the early deployments, routing tables were small enough that the traffic was not significant. As networks grew in size, however, it became evident there could be a massive traffic burst every 30 seconds, even if the routers had been initialized at random times. It was thought, as a result of random initialization, the routing updates would spread out in time, but this was not true in practice.

RIP defines two types of messages.

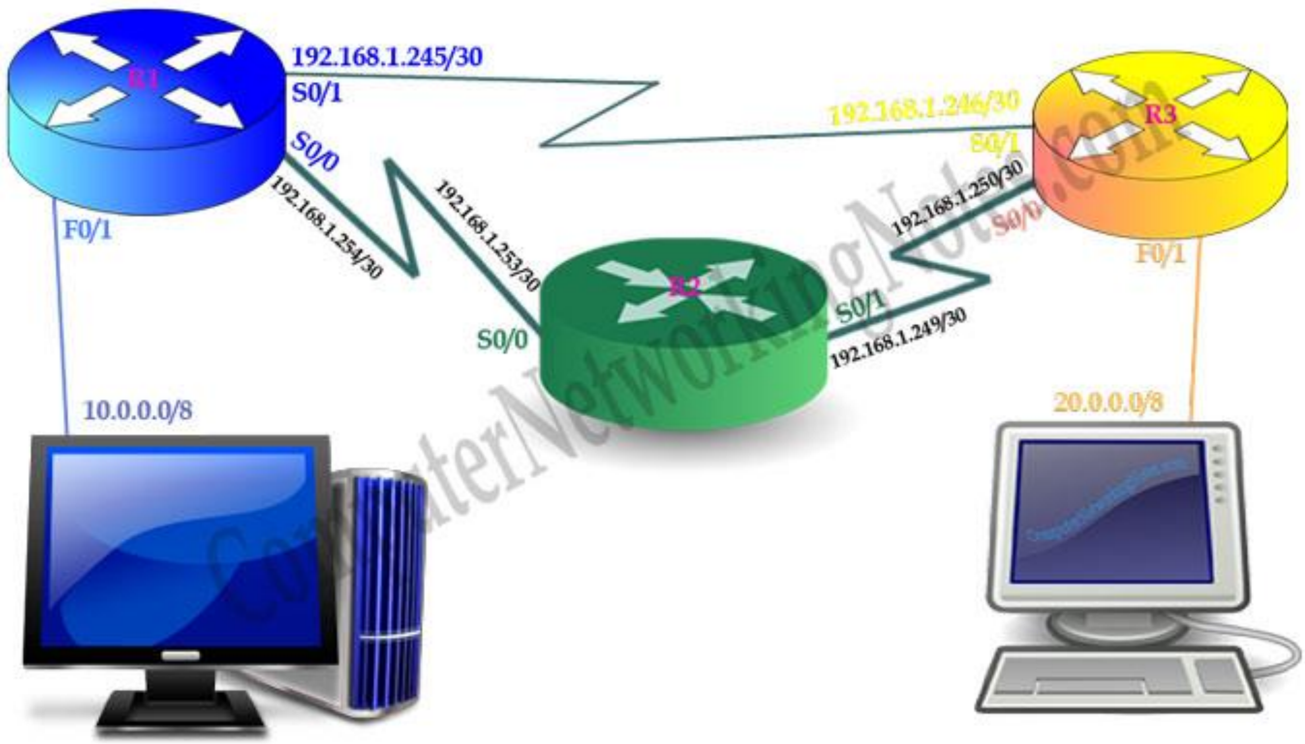
1. Request Message
2. Response Message

When a RIP router comes online, it sends a broadcast Request Message on all of its RIP enabled interfaces. All the neighbouring routers which receive the Request message respond back with the Response Message containing their Routing table. The Response Message is also gratuitously sent when the Update timer expires. On receiving the Routing table, the router processes each entry of the routing table as per the following rules

1. If there are no route entries matching the one received then the route entry is added to the routing table automatically, along with the information about the router from which it received the routing table.
2. If there are matching entries but the hop count metric is lower than the one already in its routing table, then the routing table is updated with the new route.
3. If there are matching entries but the hop count metric is higher than the one already in its routing table, then the routing entry is updated with hop count of 16 (infinite hop). The packets are still forwarded to the old route. A Holddown timer is started and all the updates for that from other routers are ignored. If after the Holddown timer expires and still the router is advertising with the same higher hop count then the value is updated into its routing table. Only after the timer expires, the updates from other routers are accepted for that route.

Routing Metrics

We may have multiple links to the destination network. In this situation router may learn multiple routes form same routing protocol. For example in following network we have two routes between PC0 and PC1.



Route 1

PC0 [10.0.0.0/8] ⬄ Router0 [Serial0/1 – 192.168.1.254] ⬄ Router2 [Serial 0/1 – 192.168.1.253] ⬄ PC1 [20.0.0.0/8]

Route 2

PC0 [10.0.0.0/8] ⬄ Router0 [Serial0/0 – 192.168.1.249] ⬄ Router1 [Serial 0/0 – 192.168.1.250] ⬄ Router1 [Serial 0/1 – 192.168.1.246] ⬄ Router2 [Serial 0/0 – 192.168.1.245] ⬄ PC1 [20.0.0.0/8]

In this situation Router uses metric to select the best path. Metric is a measurement which is used to select the best path from multiple paths learned by a routing protocol. RIP use hop count as metric to determine the best path. Hops are the number of layer3 devices which a packet crossed before reaching at destination.

RIP is a distance vector routing protocol. It uses distance [accumulated metric value] and direction [vector] to find and select the best path for destination network. We have explained this process with example in our first part of this article.

Okay now understand the concept of metric; tell me which route will Router0 use to reach at 20.0.0.0/8 network?

If it selects S0/1 [192.168.1.245/30] route, it has to cross single layer 3 device.

If it takes S0/0 [19.168.1.254/30] route, it will have to cross two layer 3 devices [Router1 and latter Router2] to get the destination network.

So it will take the first route to get 20.0.0.0/8 network.

Routing by Rumor

Sometimes RIP is also known as routing by rumor protocol. Because it learns routing information from directly connected neighbors and assume that these neighbors might have learned from their neighbors.

Advertising Updates

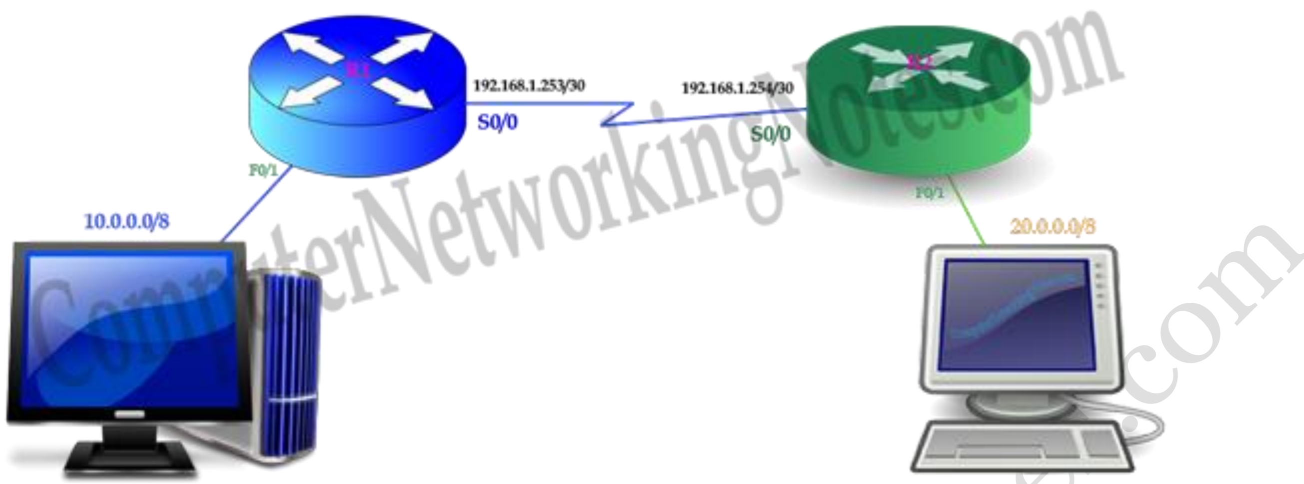
RIP periodically broadcast routing information from all of its ports. It uses local broadcast with destination IP of 255.255.255.255. While broadcasting it doesn't care who listen these broadcast or not. It doesn't use any mechanism to verify the listener. RIP assumes that if any neighbor missed any update, it will learn from next update or from any other neighbors.

Passive interface

By default RIP broadcasts are sent from all interfaces. RIP allows us to control this behavior. We can configure which interface should send RIP broadcast or which not. Once we mark any interface as passive interface, RIP will stop sending updates from that interface.

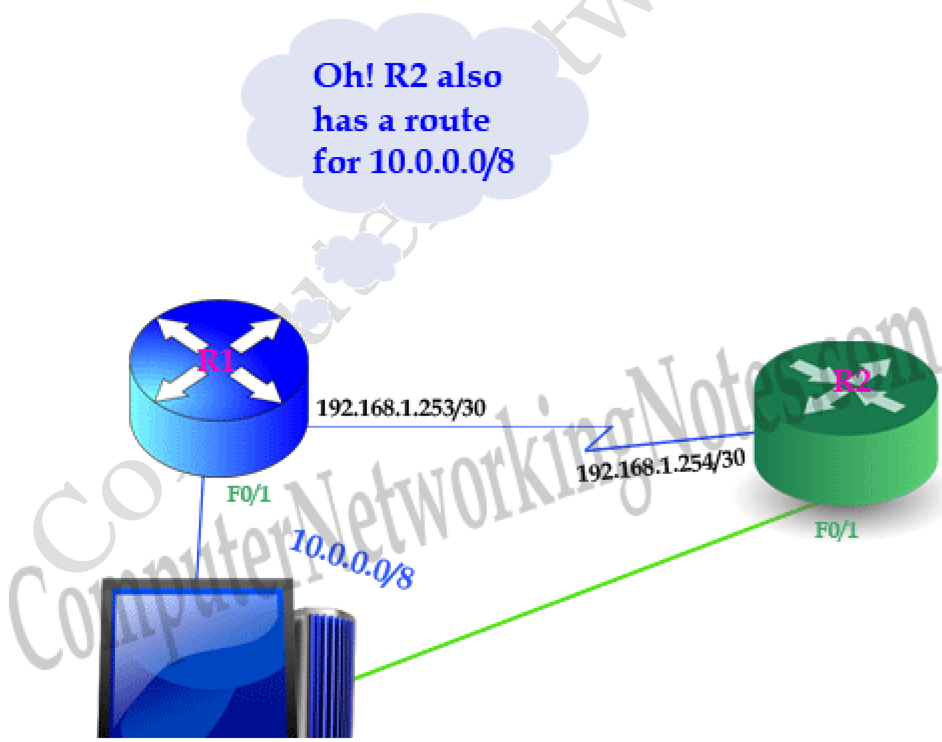
Split Horizon

Split horizon is a mechanism that states if a router receives an update for a route on any interface, it will not propagate the same route information back to the sender router on same port. Split horizon is used to avoid routing loops.
To understand this function more clearly let's take an example. Following network is using RIP protocol. R1 is advertising 10.0.0.0/8. R2 is receiving this information on S0/0 port.



Once R2 learnt about 10.0.0.0/8 network it will include it in its next routing update. Without split horizon it will advertise this route information back to R1 on S0/0 port.

Well R1 will not place this route in routing table because it has higher distance. But at the same it will not ignore this update. It will assume that R1 know a separate route to reach at 10.0.0.0/8 network, but that route has higher distance than the route I know. So I will not use that route to reach at 10.0.0.0/8 so far my route is working. But I can use that route if my route goes down. So it could work as a backup route for me.



This assumption creates a serious network issue. For example what happen if R1's F0/0 interface goes down? R1 has a direct link for 10.0.0.0/8 so it will immediately learn about this change.

In this situation if R1 receives a packet for 10.0.0.0/8, instead of dropping that packet, it will forward it from S0/0 to R2. Because R1 thinks R2 has an alternative route to reach at 10.0.0.0/8.

R2 will return this packet back to R1. Because R2 think R1 has a route to reach at 10.0.0.0/8.

This will create a network loop where actual route is down but R1 is thinking that R2 has a route for destination while R2 is thinking that R1 has a way to reach the destination. Thus this packet will be circle around between R1 and R2 endlessly. To prevent this issue RIP uses hop (router) count mechanism.

Hop count

RIP counts every hop (router) which a packet crossed to reach the destination. It limits the number of hop to 15. RIP uses TTL filed of packet to trace the number of hops. For each passing hop RIP decrement the TTL value by 1. If this value reaches to 0, packet will be dropped.

This solution only prevents a packet from trapping into the loop. It does not solve routing loop problem.

Split horizon solves this problem. If split horizon is enabled, learner router will never broadcast the same route information back to the sender. In our network R2 learned 10.0.0.0/8 network information from R1 on S0/0, so it will never broadcast network 10.0.0.0/8 information back to R1 on S0/0.

This solves our problem. If R1's F0/0 interface is down, both R1 and R2 would realize that there is no alternative route to reach at the 10.0.0.0/8 network.

To avoid network loops RIP deploy two more functions route poison and hold down timers.

Route poisoning

Route poisoning works in inverse mode of split horizon. When a router notices that any one of its directly connected route has failed, it will poison that route. By default a packet can travel only 15 hops in RIP. Any route beyond the 15 hops is invalid route for RIP. In a route failure condition, RIP assign a value higher than 15 to that specific route. This procedure is known as route poisoning. Poisoned route will be broadcast from all active interfaces. Receiving neighbor will ignore the split horizon rule by broadcasting the same poisoned route back to the sender. This process insure that every router update about a poisoned route.

RIP Timers

For better network optimization RIP uses four types of timers.

Hold down timer

RIP use hold down timer to give the routers enough time for propagating the poisoned route information in network. When router receives a poisoned route, it freezes that route in its routing table for a period of hold timer. During this period router will not use this route for routing. Hold–down period will be aborted only if router receives an update with same or better information for route. Default hold timer value is 180 seconds.

Route Invalid Timer

This timer is used to keep the track of discovered routes. If router does not receive an update for a route in 180 seconds it will mark that route as invalid route and broadcast an update to all neighbors letting them know that route is invalid.

Route Flush Timer

This timer is used to set an interval for a route that becomes invalid and its removal from routing table. Before removing an invalid route from routing table, it must update the neighboring routers about the invalid route. This timer gives enough time to updates the neighbors before invalid route is removed from routing table. Default route flush timer is set to 240 seconds.

Update Timer

RIP broadcast routing updates in every 30 seconds. It will do this continuously, whether something is changed in routing information or not. Once 30 seconds are expired router running RIP will broadcast its routing information from all of its interfaces.

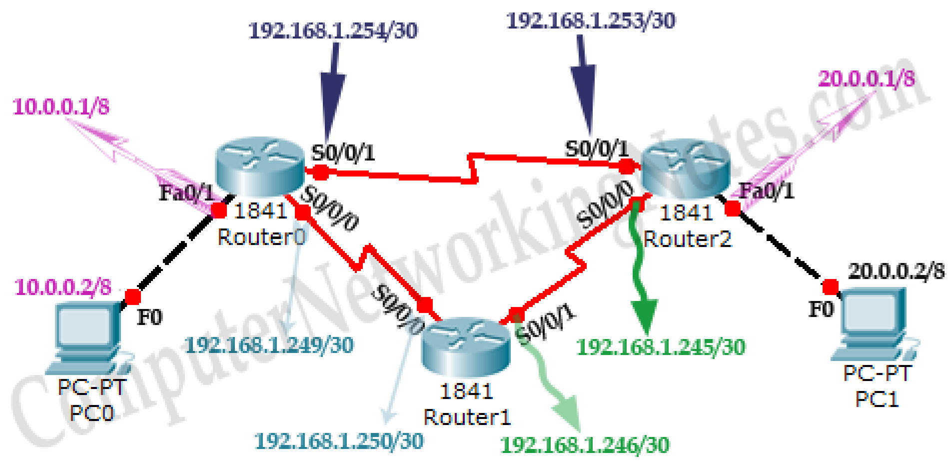
RIP is the oldest distance vector protocol. To meet the current network requirement it has been updated with RIPv2. RIPv2 is also a distance vector protocol with maximum 15 hop counts.

You can still use RIPv1 but it is not advisable. Following table lists key difference between RIPv1 and RIPv2.

Key differences between RIPv1 and RIPv2

RIPv1	RIPv2
It uses broadcast for routing update.	It use multicast for routing update.
It sends broadcast on 255.255.255.255 destination.	It sends multicast on 224.0.0.9 destination.
It does not support VLSM.	It supports VLSM.
It does not support any authentication.	It supports MD5 authentication
It only supports classful routing.	It supports both classful and classless routing.

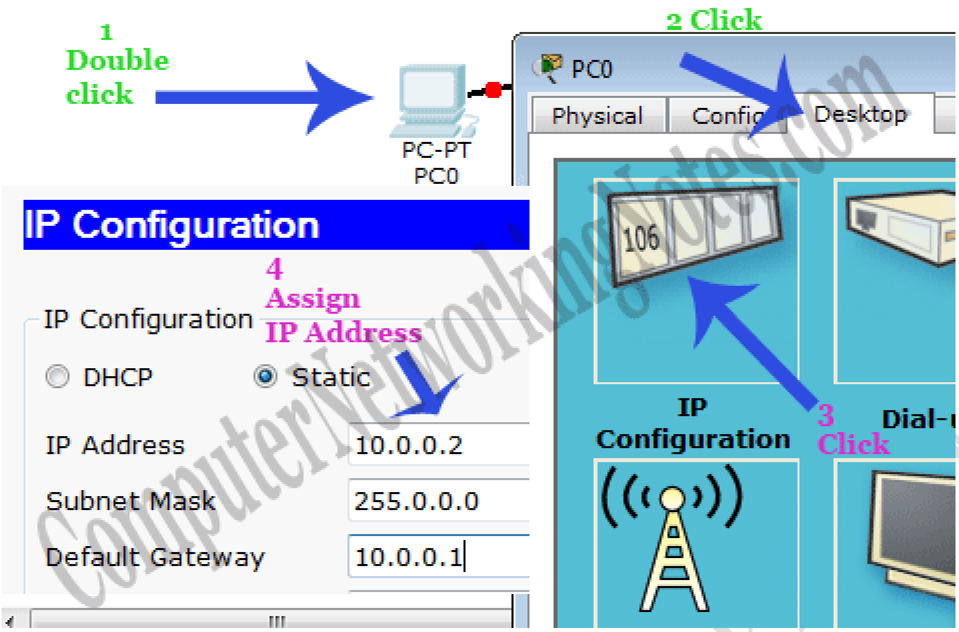
Create a topology as illustrate in following figure .



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

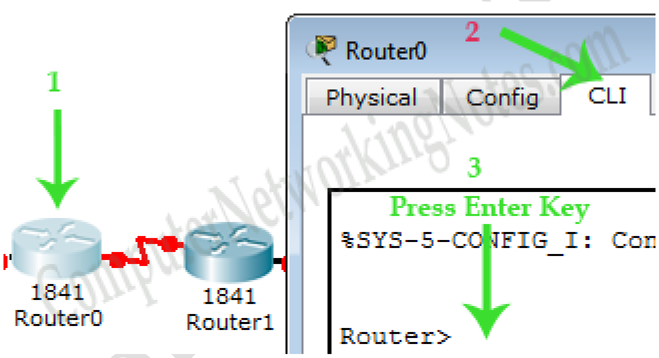
Double click **PC0** and click **Desktop** menu item and click **IP Configuration**. Assign IP address 10.0.0.2/8 to PC0.



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.

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.

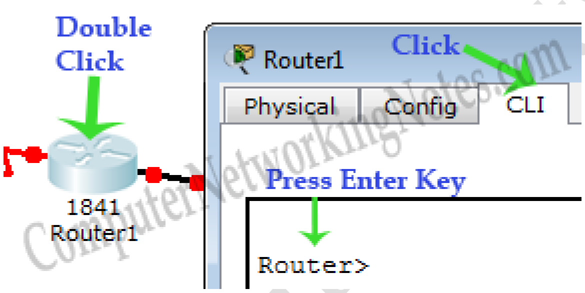
Router(config-if)#clock rate 64000 **And** *Router(config-if)#bandwidth 64* In real life environment these parameters control the data flow between serial links and need to be set at service providers end. In lab environment we need not to worry about these values. We can use these values.

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

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 RIP routing protocol that will insist them to share this information.

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

That's it. 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.

Good going we have successfully implemented RIP routing in our network.
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.

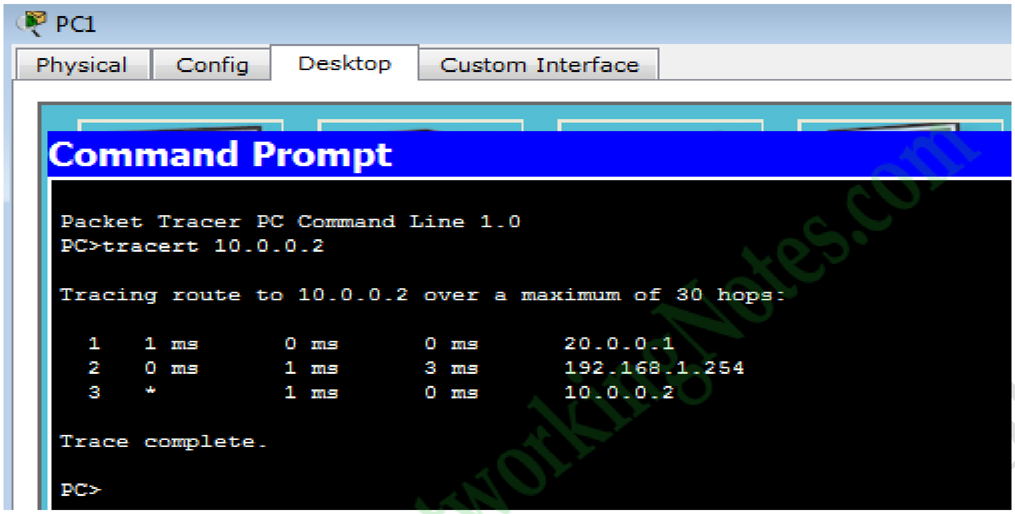
Route 1

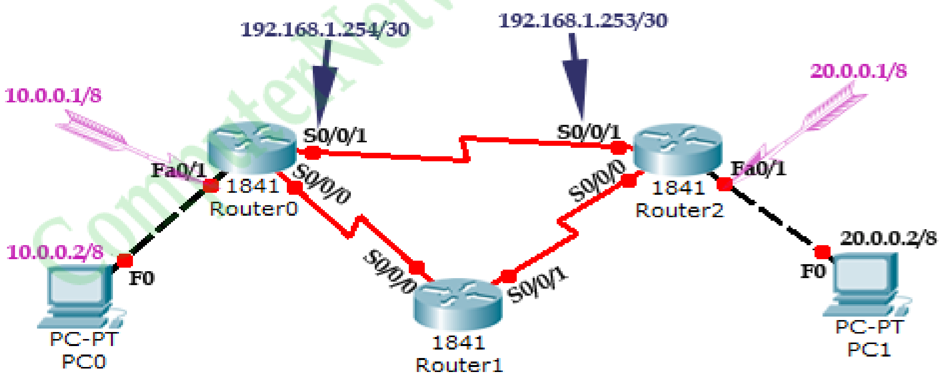
PC0 [Source / destination – 10.0.0.2] ⇔ Router0 [FastEthernet0/1 – 10.0.0.1] ⇔ Router0 [Serial0/0/1 – 192.168.1.254] ⇔ Router2 [Serial 0/0/1 – 192.168.1.253] ⇔ Router2 [FastEthernet0/0 – 20.0.0.1] ⇔ PC1 [Destination /source – 20.0.0.2]

Route 2

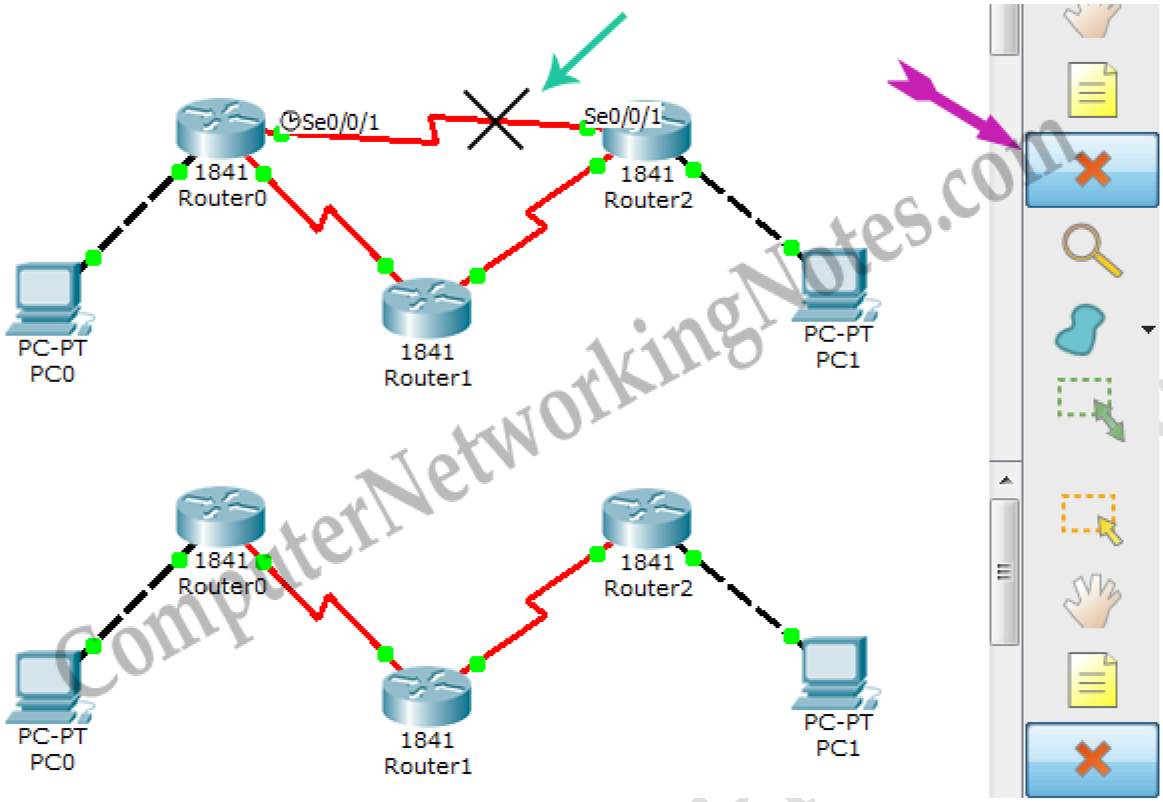
PC0 [Source / destination – 10.0.0.2] ⇔ Router0 [FastEthernet0/1 – 10.0.0.1] ⇔ Router0 [Serial0/0/0 – 192.168.1.249] ⇔ Router1 [Serial 0/0/0 – 192.168.1.250] ⇔ Router1 [Serial 0/0/1 – 192.168.1.246] ⇔ Router2 [Serial 0/0/0 – 192.168.1.245] ⇔ Router2 [FastEthernet0/0 – 20.0.0.1] ⇔ PC1 [Destination /source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



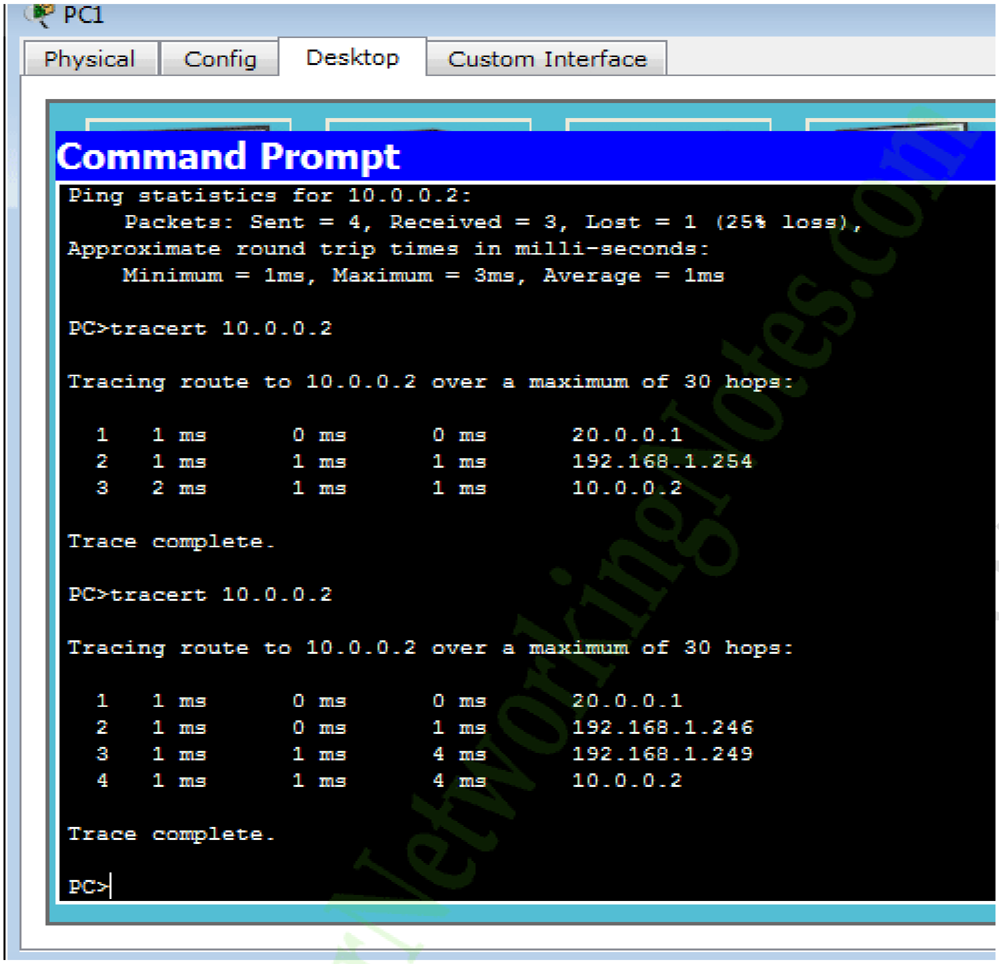


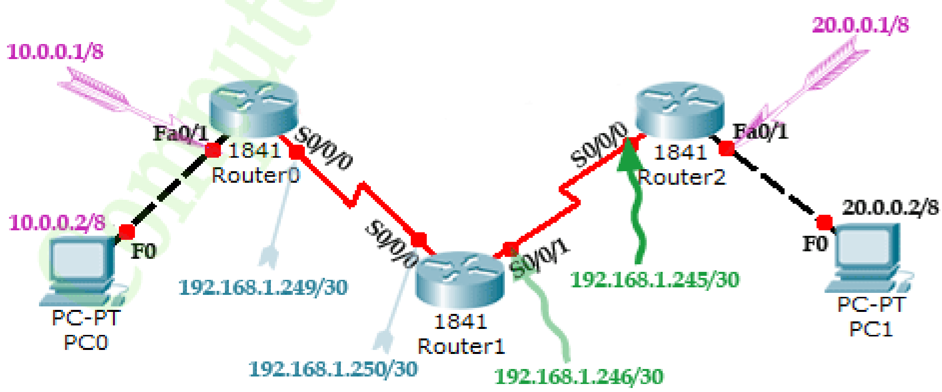
Now suppose route1 is down. We can simulate this situation by removing the cable attached between Router0 [s0/0/1] and Router2 [s0/0/1].



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
192.168.1.245/30
S0/0/0
1841
Router2
Fa0/1
20.0.0.1/8
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

LAB ASSIGMENT Questions: (Create Doc file for following ...)

Q.1 Create Topology in Cisco Packet Tracer. Implement RIP step by Step as described above. Show that it ping successfully. (i.e. screenshot of Topology and successful message Transfer.)
Q.2 what do you see when "Delaying IP RIP" command run on Router.
Q.3 which command is used to find RIP tables in Router. Perform it. Make screen shot of Table you found in each router.(for Topology created above)
Q.4 what do you mean by Split Horizon and Poison Reverse? Explain.