



Department of Computer Science and Engineering
Islamic University of Technology (IUT)
A subsidiary organ of OIC

Laboratory Report

CSE 4412: Data Communication and Networking Lab

Name : Mashrur Ahsan
Student ID : 200042115
Section : 1 (SWE)
Semester : Winter (4th)
Academic Year : 2021-22
Date of Submission : 07/03/2023
Lab No : 6

Title: Configuration of RIP in a network topology.

Objective:

1. Understand distance vector routing
2. Understand RIP
3. Understand the necessity of dynamic routing

Devices/ software Used:

1. Cisco Packet Tracer

Theory:

Distance Vector (DV) Routing

Distance vector is the "**Dynamic Routing**" protocol. A distance-vector protocol calculates the distance and direction of the vector of the next hop from the information obtained by the neighboring router. One router counts as one hop. It determines the **best path** for data to travel between network nodes.

Every node in the network should have information about its **neighboring node**. Each node in the network is designed to share information with all the nodes in the network. The nodes share the information with the neighboring node from time to time as there is a change in network topology.

Each router maintains a table that lists the distance to all other nodes in the network, as well as the next hop router that should be used to reach those nodes.

Each router periodically sends its entire routing table to its neighboring routers, and updates its own table based on the information received from its neighbors. This process continues until all routers have converged on a consistent set of routing tables.

Pros:

- Easy to implement for small networks
- Debugging's easy
- Very small room for redundancy

Cons:

- It takes considerable amount of time to update. (i.e., broken links)
- Producing a routing table can take a lot of time for large/complex networks
- Every change in the network creates traffic.

The **Routing Information Protocol** (RIP) implements the distance vector routing protocol to find the best path in the network along with some considerations.

Count to Infinity problem in DV routing

Distance Vector Routing Protocol uses the Bellman-Ford algorithm and cannot prevent routing loops (A network problem in which packets continue to be routed in an endless circle) and that's how the count-to-infinity problem occurs.

Routers may continue to increment their reported distance to a destination until it reaches infinity, which can take an indefinite amount of time.

Routing loops usually occur when an interface (link) goes down or two routers send updates simultaneously or when router receives false or inconsistent information about the state of the network from its neighboring routers causing it to update its routing table incorrectly.

Failure of the interface can happen because of several reasons like insufficient bandwidth to support the traffic volume, misconfigured duplex and speed settings, excessive buffering on interfaces, misconfigured Ether Channels, and faulty cables or hardware.

The count-to-infinity problem can be prevented by following the methods below:

- Route poisoning (Metric value between the routers updated to infinity)
- Split Horizon (Never sending routing info back where it came from)

Two node Loop problem in DV routing

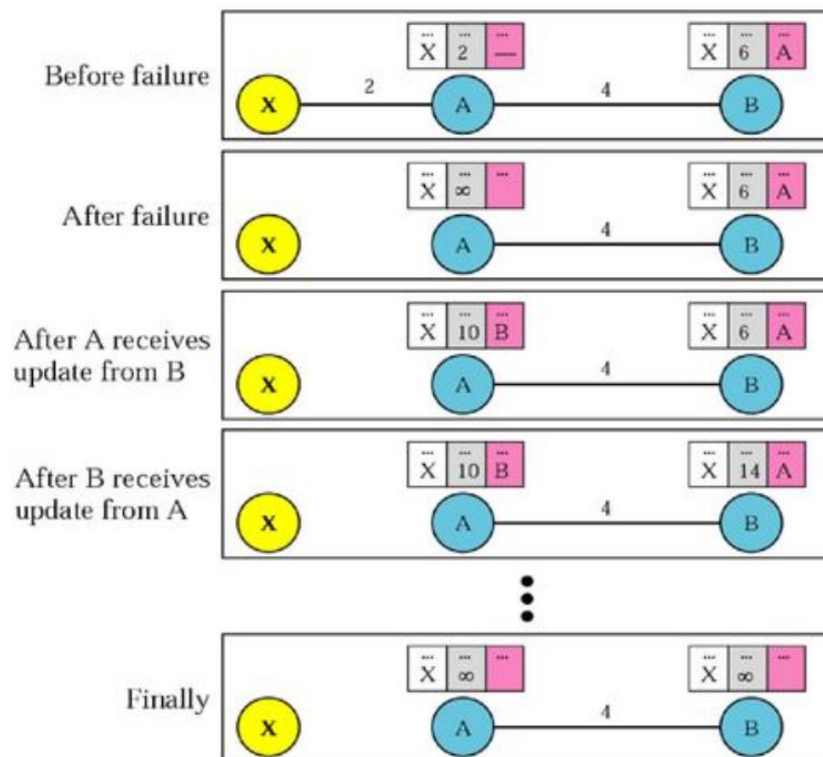
The two-node loop problem in Distance Vector (DV) routing occurs when two routers are directly connected to each other and there is a **loop between them**. In this scenario, each router will send info to its neighbor that it has the shortest path to the other router, causing a **"bouncing" effect** where the routing table of each router continually updates with conflicting information.

Let's say Router X and Router Y are directly connected to each other, and they both have a destination network that can only be reached through the other router.

Node X thinks that the route to A is via Y; node Y thinks that the route to A is via X. If there is a packet to A, Packet bounces between X and Y, **creating a two-node loop problem**.

When packet goes from X to Y, X will update its own routing table (include an entry with metric value of 1). Then Y will receive info from X and update its own routing table, indicating the shortest path to destination with a metric value of 2; since it has to go through X. Again, X will receive info from Y and will update its routing table like before. This **process will go on indefinitely**, with each router continually updating its routing table based on conflicting information from its neighbor. The result is a loop that can cause significant network congestion and instability.

To **prevent** the two-node loop problem, DV routing protocols often include a mechanism called **split horizon**, which prevents a router from advertising a route back to the same router that it learned the route from.



Split Horizon (one solution to instability)

Split Horizon is a method used by DVP that ensures that a router never sends routing information back in the direction in which it came from. It prevents network routing loops.

For example, let's say Router A and Router B are connected to each other and Router A has a destination network that can only be reached through Router B and vice-versa. When Router **B sends** its routing table to Router A, it includes an entry for the destination network with a metric value of 1 (since it is directly connected to Router A). However, Router A applies split horizon and does not advertise this route back to Router B. Instead, it advertises the route to **all other neighbors except Router B**. This prevents a loop from forming, as Router B will not receive the route back from Router A and will not update its routing table.

Pros:

- Relatively simple and easy to implement
- Improves network stability
- Prevent routing loops

Cons:

- Slows the convergence time of the network
- Increases network overhead (routers have to maintain additional info)

Poison Reverse

This technique is equivalent of “route poisoning” all possible reverse paths. Meaning, informing all routers that the path back to the original node for a particular packet has an infinite metric value.

For example, Router A will not advertise info to Router B router if split horizon is enabled. However, if **split horizon with poison reverse** is used, the route will be advertised to Router B but with the distance will be marked as infinite, indicating that the network is unreachable via this route.

For **RIP**, the metric value is set to 16, which is equivalent to infinity because the maximum RIP network hop count is 15.

Split horizon with poison reverse is **more effective** than simple split horizon in networks with **multiple routing paths**, although it results in greater network traffic. However, split horizon with poison reverse affords no improvement over simple split horizon in networks with only one routing path.

Pros:

- Fast Convergence (allows quick removal of invalid routes)
- Prevent routing loops
- Reduces network overhead when **used with split horizon**

Cons:

- Requires careful configuration
- May not be suitable for large networks (increases overhead and convergence)
- Limited scope (only works in case of link failures and topology change)

Routing Information Protocol (RIP)

Routing Information Protocol (RIP) is a distance vector routing protocol that is used in small to medium-sized networks. It uses hop count as a routing metric to find the best path between the source and the destination network.

Each RIP router **maintains a routing table**, which is a list of all the destinations the router knows how to reach.

Neighbors are the other routers to which a router is connected directly. The neighbors, in turn, pass routing information onto their nearest neighbors, and so on, until all RIP hosts within the network have the same knowledge of routing paths. This shared knowledge is known as **convergence**.

Hop count is the number of routers in between the source and destination network. The path with the lowest hop count is considered as the **best route** to reach a network and therefore placed in the routing table. RIP prevents routing loops by limiting the number of hops allowed in a path from source and destination. The maximum hop count allowed for RIP is **15** and a hop count of **16** is considered as network unreachable.

By default, each router **broadcasts/updates** its entire routing table to its closest neighbors **every 30 seconds**.

Invalid Timer determines how long a router waits before declaring a route invalid if it has not received an update about that route. By default, it's set to **180 seconds**.

Hold-Down Timer prevents routers from accepting new information about a route after declaring it invalid. The hold-down timer is set to **180 seconds** by default.

Flush Timer determines how long a router waits before removing a route from its routing table after declaring it invalid. The flush timer is set to **240 seconds** by default.

Route Time-Out Timer determines how long a router waits before removing a route from its routing table after declaring it invalid and holding it down. The route time-out timer is set **to 120 seconds** by default.

RIP mainly handle its stuff through **periodic updates**. However, RIP also supports **triggered** updates, which are sent immediately in response to changes in the network topology.

There are **3 types** of Routing Information Protocol:

- RIPv1
- RIPv2
- RIPvng

Comparison	RIP v1	RIP v2	RIPng
Updates	Broadcast (Sends it to everyone)	Multicast (Sends to a particular group)	Multicast (Sends to a particular group)
Addressing	Classful (No VLSM)	Classless protocol updated supports classful (VLSM)	Classless updates are sent (VLSM)
IPv Support	IPv4	IPv4	IPv6 only
Authentication Support	None (For updated messages)	Supports RIPv2 update messages	Supports RIPv2 update messages
Action	Broadcasts at 255.255.255.255	Multicasts at 224.0.0.9	Multicast at FF02::9
Hop count limit	15	255	More than RIPv1
Network support	Small	Larger	Larger

RIP v1 is known as Classful Routing Protocol because it doesn't send information of subnet mask in its routing update. So, it can lead to inefficient use of IP address space.

RIP v2 is known as Classless Routing Protocol because it sends information of subnet mask in its routing update. Allowing more efficient use of IP address space.

VLSM → Variable Length Subnet Mask

Here's a picture which will help us understand how RIP works:

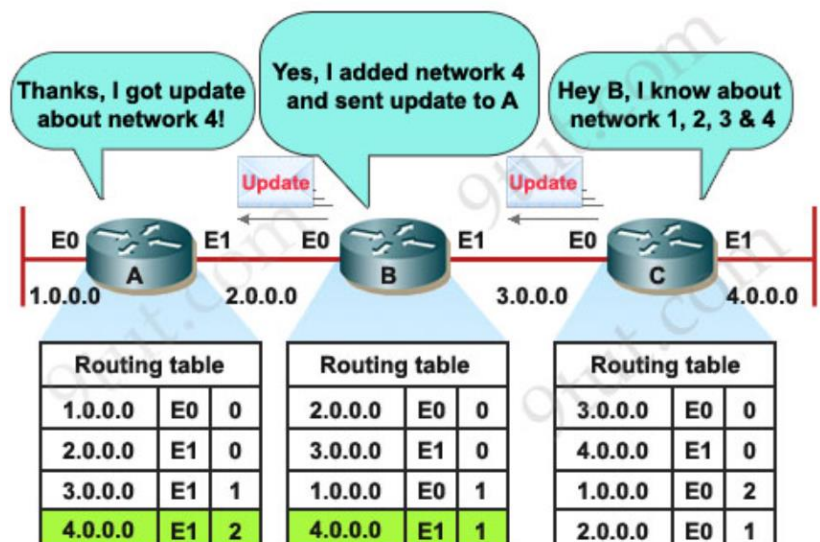
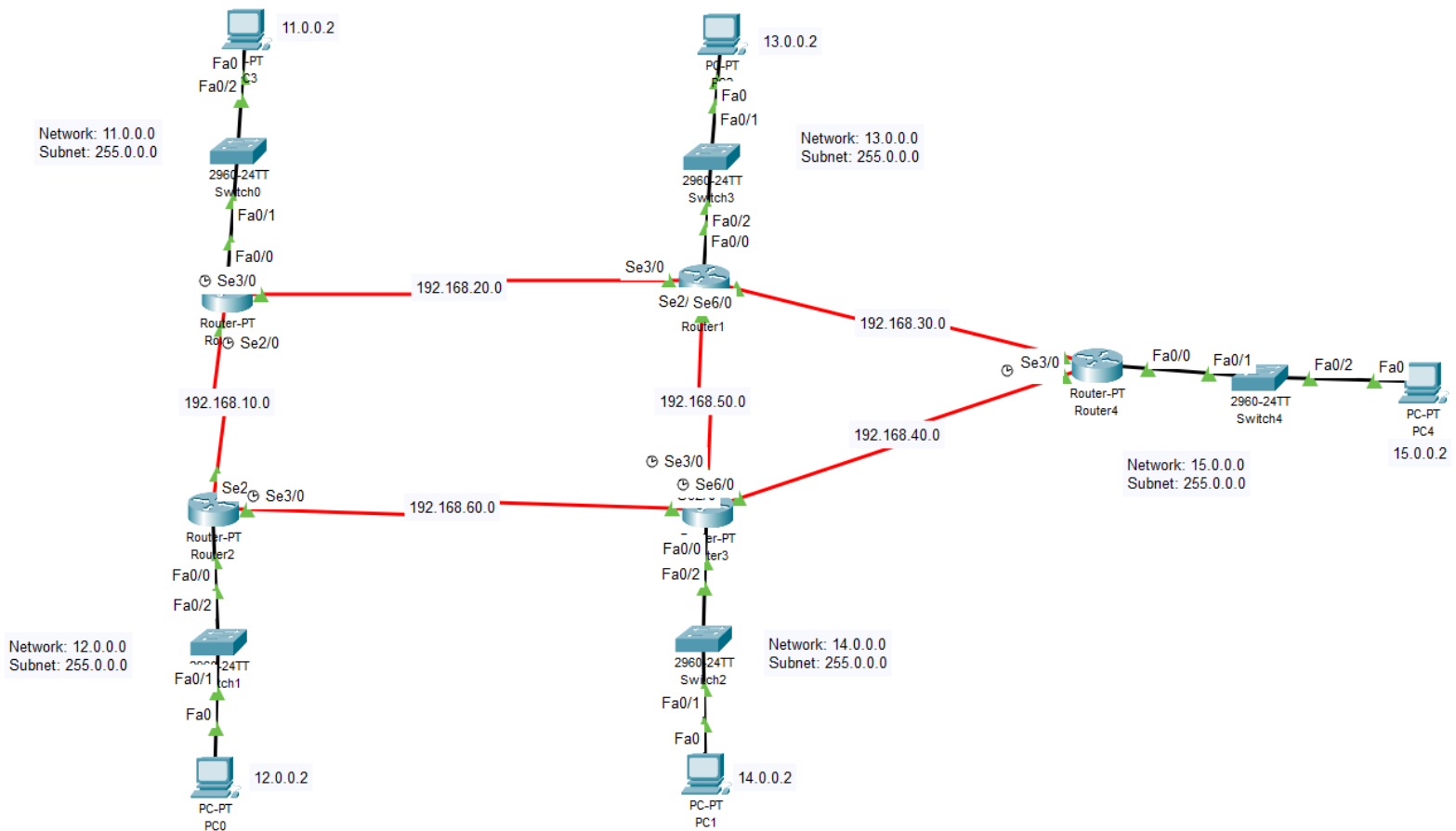


Diagram of the experiment:



Configuration of Routers:

For router4, the configuration would look like this:

```
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 15.0.0.0
Router(config-router)#network 192.168.30.0
Router(config-router)#network 192.168.40.0
Router(config-router)#
```


For router1, the configuration would look like this:

```
Router(config)#router rip
Router(config-router)#network 13.0.0.0
Router(config-router)#network 192.168.30.0
Router(config-router)#network 192.168.50.0
Router(config-router)#network 192.168.20.0
Router(config-router)#
```

Observation:

Why the first attempt may fail when simulating a Routing Information Protocol (RIP) in Cisco Packet Tracer, but the subsequent attempts are successful?

When a router first starts up, it has no information about the network topology, so it broadcasts its routing table to all its neighbors. The neighboring routers then update their routing tables based on the information they receive.

During this initial broadcast, it's possible that not all the routers have received the routing information yet, and so their routing tables may be incomplete or inconsistent. This can result in routing loops or incorrect routes, causing the first attempt to fail.

However, once the routers have exchanged their routing information and updated their routing tables, subsequent attempts to simulate the RIP protocol should be successful. This is because the routers now have a more complete and accurate view of the network topology, and can make better routing decisions.












Necessity of Dynamic Routing:

Dynamic routing is necessary in **large and complex** networks because it provides a way for routers to **automatically learn and adapt to changes** in network topology, traffic, and other conditions. In dynamic routing, routers exchange information with each other to determine the best paths to different destinations in the network, and they can adjust their routing tables in real-time based on this information.














In contrast, static routing, where routes are manually configured on each router, can be time-consuming and error-prone, especially in large networks. Static routing also does not provide the same level of **fault tolerance, efficiency, scalability** and **adaptability** as dynamic routing.

After setting up the RIP routing algorithm, if the Serial port Se3/0 of Router 4 is switched off then what are the changes occurred in Routing information of the routers.

While the Se3/0 is on:

Simulation Panel				
Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC1	 ICMP
	0.001	PC1	Switch2	 ICMP
	0.002	Switch2	Router3	 ICMP
	0.003	Router3	Router4	 ICMP
	0.004	Router4	Switch4	 ICMP
	0.005	Switch4	PC4	 ICMP
	0.006	PC4	Switch4	 ICMP
	0.007	Switch4	Router4	 ICMP
	0.008	Router4	Router3	 ICMP
	0.009	Router3	Switch2	 ICMP
	0.010	Switch2	PC1	 ICMP

While the Se3/0 is off:

Simulation Panel				
Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC1	 ICMP
	0.001	PC1	Switch2	 ICMP
	0.002	Switch2	Router3	 ICMP
	0.003	Router3	Router1	 ICMP
	0.004	Router1	Router4	 ICMP
	0.005	Router4	Switch4	 ICMP
	0.006	Switch4	PC4	 ICMP
	0.007	PC4	Switch4	 ICMP
	0.008	Switch4	Router4	 ICMP
	0.009	Router4	Router1	 ICMP
	0.010	Router1	Router3	 ICMP
	0.011	Router3	Switch2	 ICMP
	0.012	Switch2	PC1	 ICMP

If a serial port of a router running the Routing Information Protocol (RIP) is switched off, the other routers in the network will receive the updated routing information from the affected router, and the routing table of each router will be updated accordingly.

In the case of a serial port like **Se3/0** being switched off, the router will no longer be able to reach the networks that were previously reachable through that port. As a result, the router will remove the corresponding routes from its routing table and send out triggered updates to its neighboring routers.

The neighboring routers will then remove the routes that went through the affected router from their routing tables and recalculate their own routing tables accordingly.

Since all of the routers have enough knowledge about the network topology, the packet takes the next best route available. It looks for the best available route in its routing table. It was possible because Distance Vector (DV) routing protocol was followed here.

Challenges:

- It took a lot of time to gather all the information regarding everything described above.
- Familiarity issues.