RIP
(Routing Information Protocol)

Introduction

- Distance Vector Routing Protocol
- Open Standard Protocol
- RIP works well in small internetwork.
- Always count shortest path as a best path.
- It sends entire routing table to all active interfaces every **30** seconds.
- It uses only hop count to determine the best way to a remote network or destination, but it has maximum allowable hop count of **15** by default, meaning **16 is deemed unreachable.**
- RIP metric = Hop Count.
- Default Administrative Distance = **120**
- Version – **RIPv1**, **RIPv2, RIPng**

RIP Timers:

- Route update timer - **30 secs**
- Route invalid timer - **180 secs**
- Holddown timer - **180 secs**
- Route flush timer - **240 secs**

Comparison Between RIPv1, RIPv2 and RIPng

RIPv1

- Distance Vector Routing Protocol
- Supports maximum **15** hop counts
- Supports only **IPv4** networks
- Supports only **Classful** networks
- Broadcast Routing Advertisements
- No support for **VLSM** networks
- Supports No authentication

RIPv2

- Distance Vector Routing Protocol
- Supports maximum **15** hop counts.
- Support only **IPv4** networks
- Supports both **Classful & Classless** networks.
- Multicast Routing Advertisements
- Multicast Address - **224.0.0.9**
- Support for **VLSM** networks
- Support **MD5** authentication

RIPng (RIP next generation)

- Distance Vector Routing Protocol
- Supports maximum **15** hop counts
- Support for **IPv6** networks
- Supports both **Classful & Classless** networks.
- Multicast Routing Advertisements

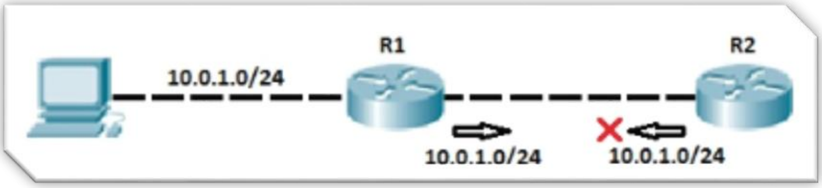
RIP Loop Prevention

- Split Horizon
- Route Poisoning
- Holddown

Split Horizon

Distance vector protocols are susceptible to routing loops. Split horizon is one of the features of distance vector routing protocols that prevents them. This feature prevents a router from advertising a route back onto the interface from which it was learned.

For example, consider the following network topology.



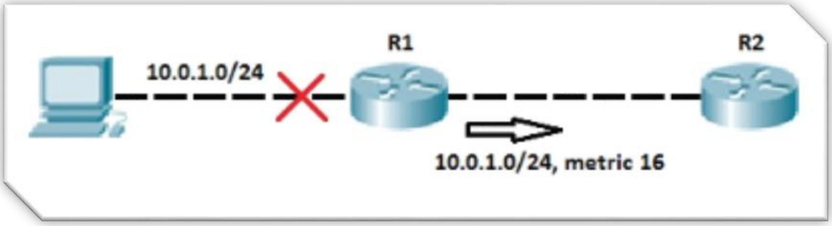
Router R2 has a route to the subnet 10.0.1.0/24 that is advertised to router R1 by using RIP. Router R1 receives the update and stores the route in its routing table. Router R1 knows that the routing update for

that route has come from R2, so it won't advertise the route back to router R2. Otherwise, if the network 10.0.1.0/24 goes down, router R1 could receive a route to the subnet 10.0.1.0/24 from R2. Router R1 now thinks that R2 has the route to reach the subnet, and uses that route. R2 receives the packets from R1 and sends them back to R2, because R2 thinks that R1 has a route to reach the subnet, thereby creating a routing loop.

Route Poisoning

Route poisoning is another method for preventing routing loops employed by distance vector routing protocols. When a router detects that one of its directly connected routes has failed, it sends the advertisement for that route with an infinite metric ("poisoning the route"). A router that receives the update knows that the route has failed and doesn't use it anymore.

Consider the following example.

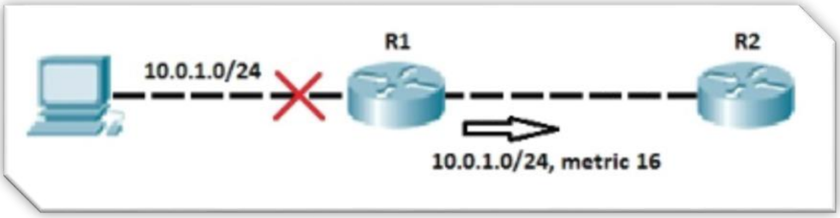


Router R1 is directly connected to the 10.0.1.0/24 subnet. Router R1 runs RIP and the subnet is advertised to R2. When the R1's Fa0/1 interface fails, an route advertisement is sent by R1 to R2, indicating that the route has failed. The route has a metric of 16, which is more than the RIP's maximum hop count of 15, so R1 considers the route to be unreachable.

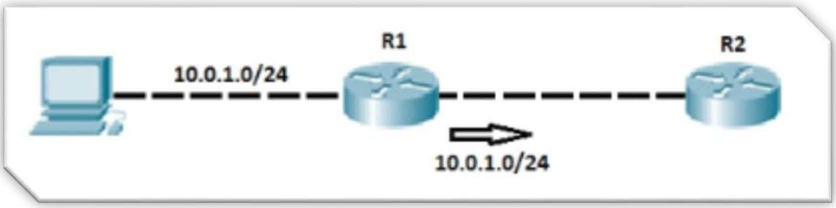
Holddown

Holddown is a loop-prevention mechanism employed by distance vector routing protocol. This feature prevents a router from learning new information about a failed route. When a router receives information about an unreachable route, a holddown timer is started. The router ignores all routing updates for that route until the timer expires (by default, 180 seconds in RIP). Only updates allowed during that period are updates sent from the router that originally advertised the route. If that router advertises the update, the holddown timer is stopped and the routing information is processed.

An example will help you understand the concept better. Consider the following network topology.



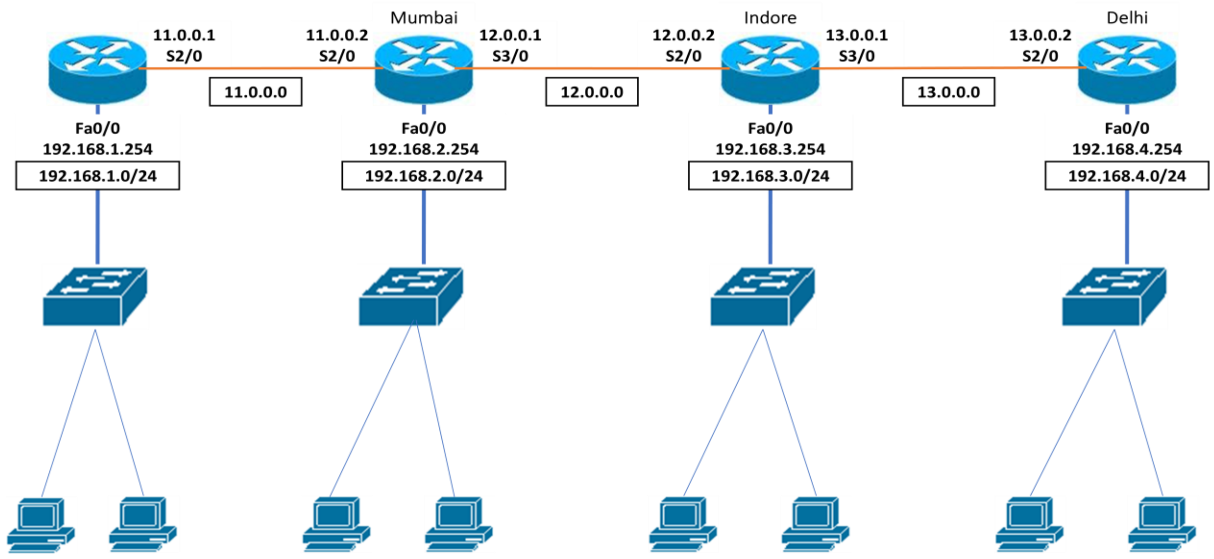
Router R1 has advertised its directly connected subnet 10.0.1.0/24 through RIP. After some period of time, the interface Fa0/1 on R1 fails and router R1 sends the poisoned route to R2. R2 receives the routing update, marks the route as unreachable and starts the holddown timer. During that time all updates from any other routers about that route are ignored to prevent routing loops. If interface Fa0/1 on R1 comes back up, R1 again advertises the route. R2 process that update even if the holddown timer is still running, because the update is sent by the same router that originally advertised the route.



RIP Configuration

Router(config)# router rip
Router(config-router)# version   <1/2>
Router(config-router)# network   <network address>

To list Routing Table:
Router#show ip route



An Internetwork

RIP Configuration for the topology “An Internetwork”

Bhopal Router:

Bhopal> enable
Bhopal# show ip route
Bhopal# configure terminal
Bhopal(config)# router rip

Bhopal(config-router)# version 2
Bhopal(config-router)# network 192.168.1.0
Bhopal(config-router)# network 11.0.0.0

Mumbai Router:

Mumbai> enable
Mumbai # show ip route
Mumbai # configure terminal
Mumbai (config)# router rip
Mumbai (config-router)# version 2
Mumbai (config-router)# network 192.168.2.0
Mumbai (config-router)# network 11.0.0.0
Mumbai (config-router)# network 12.0.0.0

Indore Router:

Indore> enable
Indore# show ip route
Indore# configure terminal
Indore(config)# router rip
Indore(config-router)# version 2
Indore(config-router)# network 192.168.3.0
Indore(config-router)# network 12.0.0.0
Indore(config-router)# network 13.0.0.0

Delhi Router

Delhi> enable
Delhi# show ip route
Delhi# configure terminal
Delhi(config)# router rip
Delhi(config-router)# version 2
Delhi(config-router)# network 192.168.4.0
Delhi(config-router)# network 13.0.0.0