Module 7

Routing and Congestion Control

Lesson 2

RIP – Routing Information Protocol

Specific Instructional Objectives

On completion of this lesson, the students will be able to:

- Explain the operation of the RIP protocol
- State the function of different fields of RIP packet format
- State the RIP routing table format
- Explain the use of different timers used in RIP
- Explain the solution to different problems encountered in RIP

7.2.1 Introduction

The **Routing Information Protocol (RIP)** is one of the most commonly used Interior Gateway Protocol on internal networks which helps a router dynamically adapt to changes of network connections by communicating information about which networks each router can reach and how far away those networks are. Although RIP is still actively used, it is generally considered to have been obsolete by Link-state routing protocol such as OSPF.

RIP was first developed in 1969 as a part of ARAPNET. One of the important thing to note about RIP is that is was built and widely adopted before a formal standard is written. It is a distance-vector protocol, which employs **Hop Count** as the metric. In the RIP metric, a router defined to be one hop from directly connected networks, two hops from networks that are reachable from one other router and so on. Thus, the Hop count along the path refers to the routers the datagram passes though while going from source to destination. The maximum number of hops allowed with RIP is 15. It runs above Network layer of the Internet protocol suite, using **UDP port 520** to carry its data.

RIP uses a distributed version of **Bellman-Ford algorithm.** *Bellman-Ford algorithm* computes single-source shortest paths in a weighted graph (where some of the edge weights may be negative). Bellman Ford runs in O(VE) time, where V and E are the number of vertices and edges.

The algorithm is distributed because it involves a number of nodes (routers) within an Autonomous system. It consists of the following steps:

- Each node calculates the distances between itself and all other nodes within the AS and stores this information as a table.
- Each node sends its table to all neighbouring nodes.
- When a node receives distance tables from its neighbours, it calculates the shortest routes to all other nodes and updates its own table to reflect any changes.

The main disadvantages of Bellman-Ford algorithm in this setting are

- Does not scale well
- Changes in network topology are not reflected quickly since updates are spread node-by-node.
- Counting to infinity

Few modifications, which will be discussed later in this section, are made in Bellman-ford algorithm to overcame the abovementioned disadvantages.

RIP partitions participiants (node within the AS) into *active* and *passive* (slient) nodes. Active routers advertise their routes to others; passive node just listen and updates their routes based on the advertisements. Passive nodes donot advertise. Only routers can run RIP in active mode; other host run RIP in passive mode. A router running in active mode broadcasts a messege or advertisement every 30 seconds. The message contains information taken from the router's current routing database. Each message consists of pairs, where each pair contains a IP network address and a integer distance to that network. All active and passive nodes listen to the advertisements and updates their route tables. Lets discuss an example for better understanding. Consider the Autonomous system consisting of 4 routers (R1, R2, R3, R4) shown in Fig. 7.2.1.

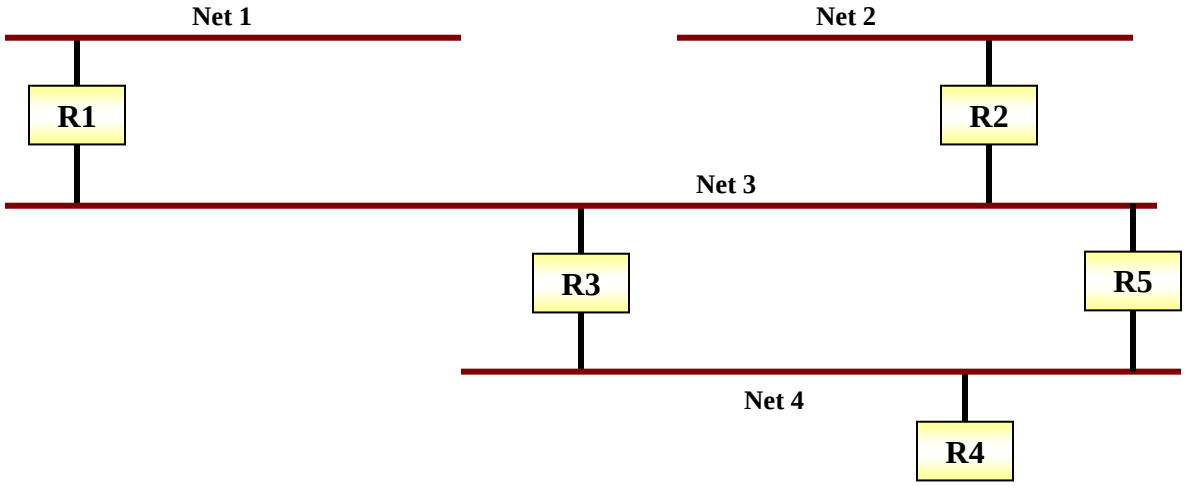

Figure 7.2.1 Example of an autonomous system

R2 will broadcast a message on network 3 (Net 3) containing a pair (2, 1), meaning that it can reach network 2 at a cost of 1. Router R1 and R3 will receive this broadcast and install a route for network 2 (Net 2) in their respective routing tables, through R2 (at a cost of 2, as now there are two routers in between either (R1 or R2) or (R2 and R3). Later on Router R3 will broadcast a message with pair (2, 2) on network 4 (Net 4). Eventually all router will have a entry for Network 2 (Net 2) in their routing tables, and same is the case with the routes for other networks too.

RIP specifies that once a router learns a route from another router, it must keep that route until it learns a better one. In our example, if router R3 and R5 both advertise network 2 (Net 2) or network 1 (Net 1) at cost of 2; router R2 will install a route through the one that happens to advertise first. Hence, to prevent routes from oscillating between two or more equal cost paths, RIP specifies that existing routes should be retained until a new route has strictly lower cost.

In this section we shall discuss the most important features of RIP. First we will have a look at the basic functioning of RIP, and then we shall discuss table format and the various timers used in RIP. After that we shall focus on the problem of Slow Convergence and some of its solutions. Then we shall have a look at the Message Format of RIP. Finally, we shall discuss RIP Version 2 and it's Message Format.

Table 7.2.1 A distance vector routing table

Destination Address	Hop Count	Next Router	Other Information
115.2.1.00	4	132.35.27.1	
126.3.56.6	5	176.21.11.3	
165.11.12.3	7	173.23.12.5	
188.22.33.2	6	130.22.34.7	
195.23.12.8	3	201.23.11.5	

7.2.2 Routing Table Format

As RIP is a distance vector routing protocol, it represents the routing information in terms of the cost of reaching the specific destination. Circuit priorities are represented using numbers between 1 and 15. This scale establishes the order of use of links. The router decides the path to use base on the priority list.

Once the priorities are established, the information is stored in a RIP routing table. Each entry in a RIP routing table provides a variety of information, including the ultimate destination, the next hop on the way to that destination, and a metric. The metric indicates the distance in number of hops to the destination. Other information can also be present in the routing table, including various timers associated with the route; these timers will be discussed in the next section. A distance vector routing table is shown in Table 7.2.1.

RIP maintains only the best route to a destination thus whenever new information provides a better route, it would replaces the old route information. Network topology alterations can provoke changes to routes, causing, for example, a new route to become the best route to a particular destination.

When network topology changes occur, they are reflected in routing update messages. For example, when a router detects a link or router failure, it recalculates its routes and sends routing update messages. Each router receiving a routing update message that includes a change updates its tables and propagates the change.

7.2.3 RIP Timers

Like other routing protocols, RIP uses certain timers to regulate its performance. The biggest drawback to a RIP router is the broadcast it makes. RIP uses numerous timers to regulate its performance. These include a routing-update timer, a route-timeout timer, and a route-flush timer. The routing-update timer clocks the interval between periodic routing updates, each router periodically transmits it entire routing table to all the other routers on the network. Generally, it is set to 30 seconds, with a small random amount of time added whenever the timer is reset. This is done to help prevent congestion, which could result from all routers simultaneously attempting to update their neighbors. The update timer ensures that each router will send a complete copy of its routing table to all neighbors every 30 seconds. While this alone is not a major detriment to network traffic, the routers also transmit a route response packet.

This is controlled by the *route invalid timer* (or *route-timeout timer*), which determines how much time must expire without a router having heard about a particular route before that route is considered invalid. Each routing table entry has a route-timeout timer associated with it. When the route-timeout timer expires, the route is marked invalid and neighbors are notified of this fact. Typical initial value of route invalid timer is 90 sec.

This notification of invalid route must occur prior to expiration of the *route flush timer*. When the route flush timer expires, the route is removed from the routing table. Typical initial value for route flush timer is 270 seconds.

Hence, routing update timer determines what the clock interval between two routing updates; route invalid timer determines when a route should be marked as Invalid, without having heard about the same; and finally router flush timer determines when to remove a route from the table.

7.2.4 Hop-Count Limit

RIP prevents routing loops from continuing indefinitely by implementing a limit on the number of hops allowed in a path from the source to a destination. The maximum number of hops in a path is 15. If a router receives a routing update that contains a new or changed entry, and if increasing the metric value by 1 causes the metric to be infinity (that is, 16), the network destination is considered unreachable. The downside of this stability feature is that it limits the maximum diameter of a RIP network to less than 16

hops**.** An example would if Router 2's link to Network A is via Router 1's link i.e. R2 has learned about a route to network A from R1 initially.

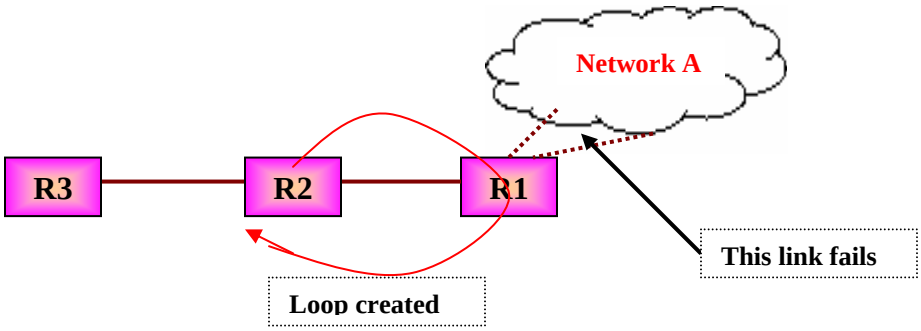

Figure 7.2.2 Count to infinity problem

If Router 1's link to network A fails, R1 will update its routing table immediately to make the distance 16 (infinite). In the next broadcast, R1 will report the higher cost route. Now suppose r2 advertise a route to Network A via R1 in its normal advertisement message, just after R1's connection to network A fails. If so R1 will receive this update message and sees that Router 2 has a two-hop link (which is actually via Router 1) to Network A, according to the normal vector-distance algorithm it will install a new route to network A via R2, of length 3.

After this, it would began advertising it has a three-hop link to Network A and then route all traffic to Network A through R2. This would create a routing loop, since when Router 2 (R2) sees that Router 1 get to Network A in three hops, it alters it own routing table entry to show it has a four-hop path to Network A.

This is known as *Count-to Infinity problem*, i.e. bad news travel slowly through the network and to advertise a bad news throughout the entire network will take a long time. This problem is also called as *slow convergence problem*.In the next section we shall discuss some of the possible solutions to this slow convergence problem

7.2.5 Solution To Slow Convergence Problem.

In this section we shall discuss some of the solutions to slow converge problem, which makes operations of RIP more stable. Some of these solutions are *hold-downs*, *split horizons*, *poison reverse updates* and *triggered updates*.

Hold-Downs

Hold-downs prevent inappropriately reinstating a route that has gone bad when routers broadcast their regular update messages.

When a route is down, neighbor routers will detect it and attempt to broadcast route changes after they have calculated the new routes. This triggered route updates may not arrive at certain network devices and those devices may broadcast a regular update message stating that the route that has gone down is still good to devices that has just been notified of the network failure. As such, the latter devices contains incorrect routing information which they may potentially further advertise.

Let us examine this problem with an example, say initially all Routers (R1, R2 and R3) knows about a route to network A through Router 1 (R1). Now if the Router 1 (R1) link for network A goes down, and say the link failure message from Router 1 (R1) reaches Router 2 (R2) but not yet reached the Router 3 (R3). At this point Router 2 (R2) has no entry in its table for a route to network A. Now if a regular update message from Router 3 (R3), about the reachability information for network A, i.e. the out-dated information, reaches Router 2 (R2). Then Router 2 (R2) will think as if the route to Network A is Up and working, so both the routers- R3, R2 will have wrong information about the network.

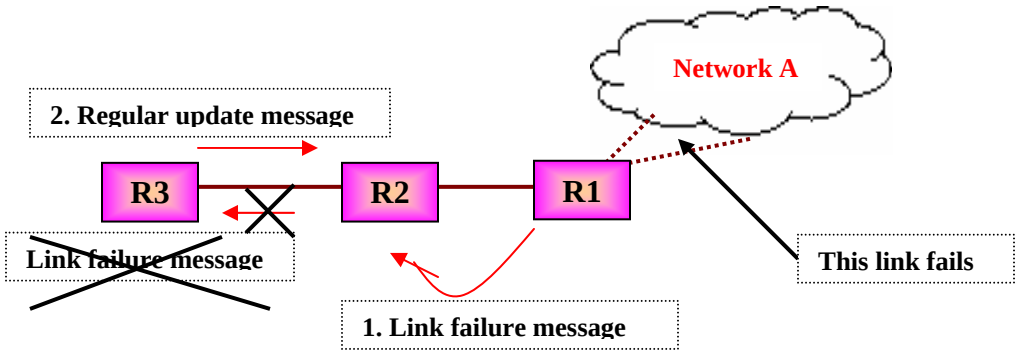

Figure 7.2.3 Hold down, solution to Slow Convergence problem

To solve the abovementioned problem, a technique known as *Hold Down* is considered. Hold downs tell routers to hold on to any changes that might affect recently removed routes for a certain period of time, usually calculated just to be greater than the period of time necessary to update the entire network with a route change. This prevents count-to-infinity problem. As per our example, it means that once R2 has removed the route to Network A, after receiving a link failure message from R1, It will not change or add any new route to network A, until a certain amount of time has passed. This time duration is known as *Hold Down time.* Typically hold down time is around 60 sec. So the idea is to wait long enough to ensure that all machines receive the bad news (link failure news) and not mistakenly accepts a message that is out dated.

Split Horizons

It is never useful to send information about a route back in the direction from which it came and thus split horizons is used to prevent updates that are redundant to the network. For this purpose Router records the interface over which it received a particular route and does not propagates its information about that route back to the same interface.

Let us consider an example in which Router 1 advertises that it has a route to Network A. If Router 2 is sending traffic to Network A via Router 1, there is no reason for Router 2 to include the route info in its update back to Router 1, because Router 1 is closer to Network A.

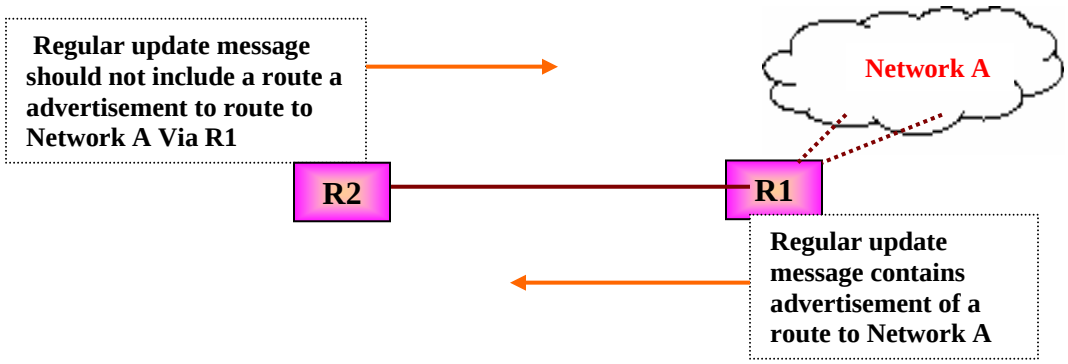

Figure 7.2.4 Split Horizon, solution to Slow Convergence problem

Without split horizon rule in place, Router 2 would continue to inform Router 1 that it can actually get to Network A through 2 hops which is via Router 1. If there is a failed direct connection to Network A, Router 1 may direct traffic to Router 2 thinking it's an alternative route to Network A and thus causing a routing loop.

Split horizon in this instance serve as an additional algorithm to achieve stability.

Poison Reverse Updates

This is yet another technique used to solve the slow convergence problem. Larger routing loops prevented using poison reverse updates. Once a connection disappears, the router advertising the connection retains the entry for several update periods, and include an infinite cost in the broadcast. The updates are sent to remove downed route and place it in hold-down.

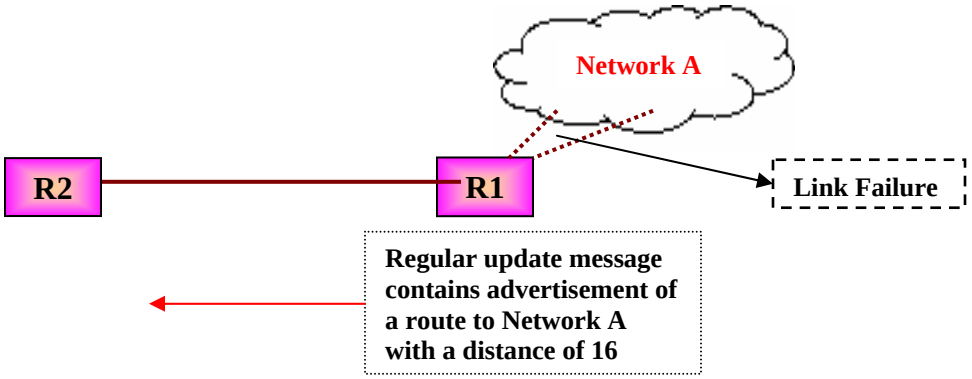

Figure 7.2.5 Poison Reverse, other solution to Slow Convergence problem

To make Poison reverse more efficient, it must be combined with *Triggered Updates*. Triggered updates force a router to send an immediate broadcast when receiving bad news, instead of waiting for the next periodic broadcast. By sending an update immediately, a router minimizes the time it is vulnerable to believing in good news.

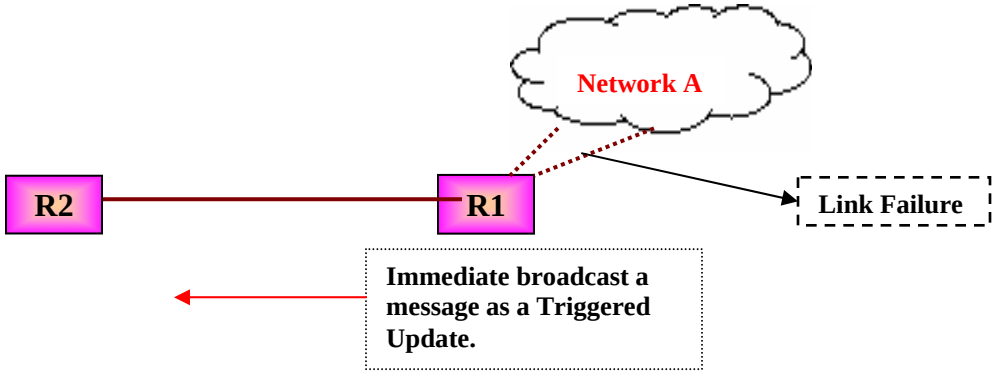

Figure 7.2.6 Poison Reverse along with triggered Update

7.2.6 RIP Message Format

The following section focuses on the RIP packet format. RIP messages can be broadly classified into two types: routing information messages and messages used to request information. Both uses same format, which consists of fixed header information followed by optional list of network and distance pairs. Figure 7.2.7 illustrates the IP RIP packet format.

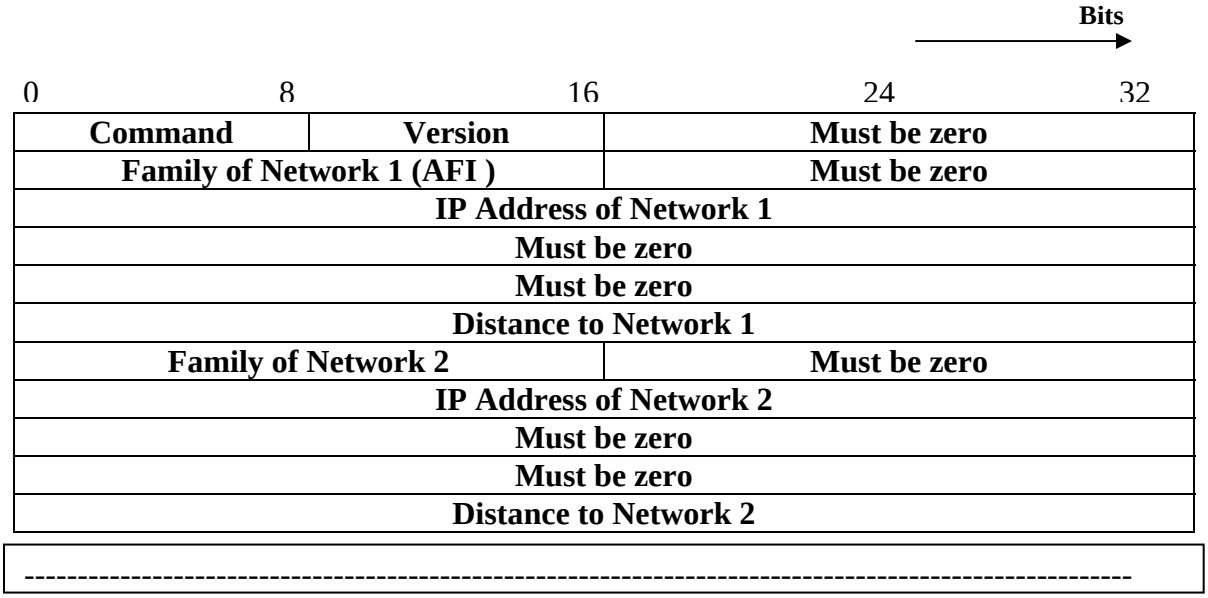

Figure 7.2.7 RIP Message

After first 32- bit header, the RIP message contains a sequence of pairs, where each pair consists of a network IP address and a integer distance to that network.

The following descriptions summarize the IP RIP packet format fields illustrated in Figure 5.7:

Command: Indicates whether the packet is a request or a response. The request asks that a router send all or part of its routing table. The response can be an unsolicited regular routing update or a reply to a request. Responses contain routing table entries. Multiple RIP packets are used to convey information from large routing tables. COMMAND specifies an operation according to the Table 7.2.2:

Table 7.2.2 Meaning for different values of command field

Value in command field	Meaning
1	Request for partial or full routing information
2	Response containing network distance pair from sender's routing table
3	Turn on Trace mode (obsolete now)
4	Turn off Trace mode (obsolete now)
5	Reserved for SUN Microsystems internal use

Version number—Specifies the RIP version used. This field can signal different potentially incompatible versions.

Zero—This field is not actually used by RFC 1058 RIP; it was added solely to provide backward compatibility with prestandard varieties of RIP. Its name comes from its defaulted value: zero.

Address-family identifier (AFI)—Specifies the address family used. RIP is designed to carry routing information for several different protocols. Each entry has an address-family identifier to indicate the type of address being specified. The AFI for IP is 2.

Address—Specifies the IP address for the entry.

Metric—Indicates how many internetwork hops (routers) have been traversed in the trip to the destination. This value is between 1 and 15 for a valid route, or 16 for an unreachable route.

A router or host can ask another router for routing information by sending a *request* command. Router replies to request using *Response* command. In most of the cases router broadcast unsolicited response messages periodically.

RIP messages do not contains explicit length field. RIP assumes that underlying delivery mechanism will tell the receiver about the incoming message length. As RIP operates on UDP port 520, so it depends on UDP for this purpose.

7.2.7 RIP version 2

Most of the slow convergence problems are handled by split horizon, poison reverse, and triggered updates. However, RIP cannot increase network diameter or disseminate network bit masks needed to properly interpret routes thus it is a poor choice for modern network. An updated version of RIP, known as RIPv2, solves this problem.

RIP Version 2 (RIPv2) RIP Version 2 adds a "network mask" and "next hop address" field to the original RIP packet while remaining completely compatible with RIP. Thus RIPv2 routers can coexist with RIP routers without any problems.

The subnet mask field contains the network bit mask associated with the destination; it also allows the implementation of CIDR addressing. This will allow RIP to function in a variety of environments, which may implement variable subnet masks on a network.

The "next hop address" field provides the address of the gateway thus allowing optimization of routes in an environment which uses multiple routing protocols thus having to ability to understand other routing protocol which may provide a better route path to a destination.

Authentication is another improvement RIPv2 offers over RIP-1. It defines password authentication mechanism for RIPv2 routers to prevent accidental updates for misconfigured hosts.

In addition to the above, RIPv2 uses multicasting instead of broadcast to reduce the load on systems that do not want RIPv2 updates and for sharing information which RIP-1 routers will not hear. In multicasting, only a set of hosts listening on a specific IP multicast address will hear the information. Other remaining fields like routing domain and route tag are presently still limited in usage.

RIP 2 Message Format

The RIP 2 specification (described in RFC 1723) allows more information to be included in RIP packets and provides a simple authentication mechanism that is not supported by RIP. Figure 7.2.8 shows the IP RIP 2 packet format. Functions of different fields are summarized below:

Command—Indicates whether the packet is a request or a response. The request asks that a router send all or a part of its routing table. The response can be an unsolicited regular routing update or a reply to a request. Responses contain routing table entries. Multiple RIP packets are used to convey information from large routing tables.

Version—Specifies the RIP version used. In a RIP packet implementing any of the RIP 2 fields or using authentication, this value is set to 2.

Address-family identifier (AFI)—Specifies the address family used. RIPv2's AFI field functions identically to RIP's AFI field, with one exception: If the AFI for the first entry in the message is 0xFFFF, the remainder of the entry contains authentication information. Currently, the only authentication type is simple password.

Route tag—Provides a method for distinguishing between internal routes (learned by RIP) and external routes (learned from other protocols).

IP address—Specifies the IP address for the entry.

Subnet mask—Contains the subnet mask for the entry. If this field is zero, no subnet mask has been specified for the entry.

Next hop—Indicates the IP address of the next hop to which packets for the entry should be forwarded.

Metric—Indicates how many internetwork hops (routers) have been traversed in the trip to the destination. This value is between 1 and 15 for a valid route, or 16 for an unreachable route.

Command	**Version**	**Must be zero**
Family of Network 1 (AFI)		**Route Tag**
IP Address of Network 1		
Subnet mask for Network 1		
Next hop Field		
Distance to Network 1 (Metric)		
Family of Network 2		**Route Tag**
IP Address of Network 2		
Subnet mask for Network 2		
Next hop Field		
Distance to Network 2 (Metric)		

Figure 7.2.8 RIP2 packet format

Fill In The Blanks

1. RIP and OSPF are _____ Gateway Protocol.
2. RIP is abbreviated as _____.
3. RIP uses a _____ vector algorithm
4. RIP employs _____ as the metric
5. The maximum number of hops allowed with RIP is _____.
6. RIP runs above Network layer, using _____ at port _____ to carry its data.
7. RIP uses a distributed version of _____ algorithm
8. Active routers _____ their routes to others; passive node just _____ and updates their routes based on the advertisements.
9. _____ nodes donot advertise.
10. Command field in RIP header equal to one means _____
11. RIPv2 adds a _____ and _____ field to the original RIP packet

Answers.

1. Interior
2. Routing Information Protocol
3. Distance
4. Hop count
5. 15
6. User datagram protocol (UDP) , 520
7. Bellman-Ford
8. advertise, listen
9. Passive
10. Request for partial or full routing information
11. network mask, next hop address

Short Answer Questions

1. Describe Bellman-Ford Algorithm.

Ans: The algorithm is distributed because it involves a number of nodes (routers) within an Autonomous system. It consists of the following steps:

1) Each node calculates the distances between itself and all other nodes within the AS and stores this information as a table.
2) Each node sends its table to all neighbouring nodes.
3) When a node receives distance tables from its neighbours, it calculates the shortest routes to all other nodes and updates its own table to reflect any changes.

2. What are the disadvantages of Bellman Ford Algorithm?

Ans: The main disadvantages of Bellman-Ford algorithm in this setting are:

1) Does not scale well
2) Changes in network topology are not reflected quickly since updates are spread node-by-node.
3) Counting to infinity

3. Describe various Timer used in RIP.

Ans: RIP uses numerous timers to regulate its performance. These include a routing-update timer, a route-timeout timer, and a route-flush timer.

1. The **routing-update timer** clocks the interval between periodic routing updates, each router periodically transmits it entire routing table to all the other routers on the network. Generally, it is set to 30 seconds, with a small random amount of time added whenever the timer is reset.
2. The **route invalid timer** (or **route-timeout timer**), which determines how much time must expire without a router having heard about a particular route before that route is considered invalid.
3. This notification of invalid route must occur prior to expiration of the **route flush timer**. When the route flush timer expires, the route is removed from the routing table.

4. Explain the Count to Infinity problem.

Ans: **Count-to Infinity problem** is a problem that bad news travel slowly through the network and to advertise a bad news throughout the entire network will take a long time. This problem arises because routing update messages propagates slowly across the network. Choosing a small infinity limit (16) reduces it but doesn't eliminate it. This problem causes inconsistencies in the routing tables of different routers. This problem is also called as slow convergence problem.

5. Name few solutions to Slow Convergence Problem.

Ans: Some of the solutions to slow convergence problem are

- Hold-downs
- Split horizons
- Poison reverse updates
- Triggered updates.

6. Explain Hold Down Solution

Ans: Hold-Down is one of the solutions to Slow Convergence problem. Hold-downs prevent inappropriately reinstating a route that has gone bad when routers broadcast their regular update messages. Hold downs tell routers to hold on to any changes that might affect recently removed routes for a certain period of time, usually calculated just to be greater than the period of time necessary to update the entire network with a route change. This prevents count-to-infinity problem. This time duration is known as **Hold Down time**. Typically hold down time is around 60 sec. So the idea is to wait long enough to ensure that all machines receive the bad news (link failure news) and not mistakenly accepts a message that is out dated

7. Explain Split Horizon Technique.

Ans: It is never useful to send information about a route back in the direction from which it came and thus split horizons is used to prevent updates that are redundant to the network. For this purpose Router records the interface over which it received a particular route and does not propagates its information about that route back to the same interface. This change is known as Split Horizon Technique.

8. Explain Poison reverse and Triggered updates technique.

Ans: Once a connection disappears, the router advertising the connection retains the entry for several update periods, and include an infinite cost in the broadcast. The updates are sent to remove downed route and place it in hold-down. This sending of update immediately is known as **poison reverse**.

To make Poison reverse more efficient, it must be combined with **Triggered Updates**. Triggered updates force a router to send an immediate broadcast when receiving bad news, instead of waiting for the next periodic broadcast. By sending an update immediately, a router minimizes the time it is vulnerable to believing in good news.