

ROUTING INFORMATION PROTOCOL (RIP)

A CASE STUDY REPORT

Submitted by

AADIT VINAYAK (RA2211029010012)

VEDANT PANDEY (RA2211029010013)

CHIRANJEEV KUMAR (RA2211029010019)

for the Course

21CSC302J- COMPUTER NETWORKS

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND

ENGINEERING with specialization in

COMPUTER NETWORKING



DEPARTMENT OF NETWORKING AND

COMMUNICATIONS

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603 203

NOVEMBER 2024

Department of Networking and Communications



Own Work Declaration Form

Degree/Course: **B. Tech in Computer Science & Engineering with Specialization in Computer Networking**

Student Name: **Aadit Vinayak, Vedant Pandey & Chiranjeev Kumar**

Registration Number: **RA2211029010012, RA2211029010013 & RA2211029010019**




Title of Work: **Routing Information Protocol (RIP)**

We hereby certify that this assessment complies with the University's Rules and Regulations relating to Academic misconduct and plagiarism, as listed in the University Website, Regulations, and the Education Committee guidelines.

We confirm that all the work contained in this assessment is our own except where indicated, and that we have met the following conditions:

- Clearly referenced / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc)
- Given the sources of all pictures, data etc. that are not my own
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g.fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course handbook / University website

We understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:	
We are aware of and understand the University's policy on Academic misconduct and plagiarism and we certify that this assessment is our own work, except where indicated by referring, and that we have followed the good academic practices noted above.	
Aadit Vinayak [RA2211029010012]	
Vedant Pandey [RA2211029010013]	
Chiranjeev Kumar [RA2211029010019]	



SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University by U.O. No. 1 of 130C Act, 1956

SRM Institute of Science and Technology

Kattankulathur – 603 203

BONAFIDE CERTIFICATE

Certified that 21CSC302J case study report titled “**ROUTING INFORMATION PROTOCOL (RIP)**” is the bonafide work of **AADIT VINAYAK (RA2211029010012)**, **VEDANT Pandey(RA2211029010013)**, **CHIRANJEEV KUMAR (RA2211029010019)** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr B. Yamini

**Course Handling Faculty Computer
Networks-21CSC302J**

Associate Professor

**Department of Networking and
Communications**



SIGNATURE

Dr Lakshmi M.

**Head of the Department
Department of Networking
and Communications**

Abstract

The Routing Information Protocol (RIP) is a foundational distance-vector routing protocol used in computer networks to enable efficient data packet routing between routers.

This case study provides a comprehensive examination of RIP, covering its fundamental mechanisms, historical development, and practical applications in modern networks. Introduced in the 1980s as one of the earliest protocols in the TCP/IP suite, RIP functions by allowing routers to communicate periodically to update their routing tables, using hop counts as the metric for path selection.

This study delves into RIP's operating principles, such as the use of hop limits, time-based updates, and split-horizon techniques, which collectively help prevent routing loops.

Additionally, the study explores the protocol's limitations, including its maximum hop count of 15, which restricts its suitability for large-scale networks. Comparisons with other protocols, such as Open Shortest Path First (OSPF) and Enhanced Interior Gateway Routing Protocol (EIGRP), highlight RIP's strengths in simplicity and ease of configuration as well as its weaknesses in scalability and convergence time.

Furthermore, the case study assesses RIP's adaptations, such as RIP version 2 (RIPv2), which introduced improvements like subnet mask support and multicast updates, enhancing its applicability in modern networks.

Concluding remarks discuss the relevance of RIP in smaller or less complex networks and consider the protocol's legacy in the evolution of network routing standards.

Table of Contents

Chapter Number	Topic	Page Number
	Abstract	(iii)
	List of Figures	(vi)
1	Introduction to RIP	1
2	Rip's Versions	2
3	Advantages and Disadvantages of Rip	3
4	Features of Rip	5
5	Rip Updating Algorithm	6
6	Topology	7
7	Rip Configuration	9
	7.i RIPv1 Configuration	11
	7.ii Limitations of RIPv1	12
	7.iii RIPv2 Configuration	15
	7.iv RIP Timers Configuration and Example	16
	7.v RIP Loop Avoidance Mechanisms	18
	7.vi RIP Passive Interfaces	19
	7.vii RIP Neighbours	20
	7.viii RIPv2 Authentication	21
	7.ix Altering RIP's Metric	22
	7.x Interoperating between RIPv1 and RIPv2	23
	7.xi Triggering RIP Updates	24
	7.xii Troubleshooting RIP	25
8	Scenario	26
9	Conclusion	30
10	References	31

LIST OF FIGURES

Figure Number	Figure Title	Page Number
6.1	Components Used in RIP Ring Configuration	7
6.2	RIP Ring Topology Configuration	7
6.3	Simple RIP Topology	8
6.4	Fully Functioning RIP Network	8
7.1	Two Router Simple RIP v1 Configuration	11
7.2	Three Router RIP v1 Configuration	12
7.3	Updated 3 Router RIP v1 Configuration	13
7.4	RIP v2 Configuration	15
7.5	RIP Timers Example	17
7.6	RIP Loop Avoidance	18
7.7	RIP Passive Interface	19
7.8	RIP Neighbours	20
7.9	Looped RIP Metric	22
7.10	Interoperation Between RIP v1 and v2	23
8.1	Sample Campus Network Simulation	27
8.2	Sample Hostel Network Layout	29

CHAPTER 1

Introduction to RIP

Routing Information Protocol (RIP)

DEFINITION:

Routing Information Protocol (RIP) is a widely-used dynamic routing protocol designed to help routers in a network exchange information about the best paths to reach various network destinations. RIP operates by using the hop count as its routing metric, where each router hop between the source and the destination network is counted as one. The path with the fewest hops is selected as the optimal route, thus aiming to simplify route determination.

DETAILS:

Routing Information Protocol (RIP) RIP is an intradomain (or interior) routing protocol, meaning it is typically implemented within a single autonomous system (AS), such as a corporate or campus network. RIP operates based on distance-vector routing principles, where each router only has awareness of its direct neighbours and updates its routing table by receiving distance vectors from these neighbouring routers. In RIP, the network is represented as nodes (routers and connected networks), and each link is a potential path.

The routing table in RIP consists of destination network addresses, associated metrics, and the next-hop router information. The metric, or cost, for reaching each destination is measured in hop counts. To prevent endless loops in path calculation, RIP imposes a maximum hop count of 15; any path with a hop count of 16 or more is considered unreachable. This limitation means that RIP is not suitable for large-scale or complex networks, as the protocol cannot support paths that exceed 15 hops.

CHAPTER 2

RIP's VERSIONS

There are two versions of routing information protocol.

- RIP Version 1
- RIP Version2

RIP Version1:

RIP v1 is known as classful Routing Protocol because it doesn't send information of subnet mask in its routing update. It is an open standard protocol means it works on the various vendors routers. It works on most of the router, it is classful routing protocol. Updates are broadcasted. Its administrative distance value is 120, it means it is not reliable. The lesser the administrative distance value the reliability is much more. Its metric is hop count and max hop count is 15. There will be total 16 routers in the network. When there will be the same number of hops to reach destination, Rip starts to perform load balancing. Load balancing means if there are three ways to reach the destination and each way has same number of routers then packets will be sent to each path to reach the destination. This reduces traffic and also the load is balanced. It is used in small companies; in this protocol routing tables are updated in each 30 sec, whenever link breaks rip traces out another path to reach the destination. It is one of the slowest protocols.^[1]

RIP Version-2:

RIP v2 is known as Classless Routing Protocol because it sends information of subnet mask in its routing update. Due to some deficiencies in the original RIP specification, RIP version 2 was developed in 1993. It supports classless Inter-Domain Routing (CIDR) and has ability to carry subnet information, its metric is also hop count and max hop count 15 is same as rip version 1. It supports authentication and does sub netting and multicasting. Auto summary can be done on every router. In RIPv2 Subnet masks are included in the routing update. RIPv2 multicasts the entire routing table to all adjacent routers at the address 224.0.0.9, as opposed to RIPv1 which uses broadcast (255.255.255.255).^[2]

CHAPTER 3

ADVANTAGES &DISADVANTAGES OF RIP

RIP VERSION 1

Advantages of RIP v1:

1. Easy to configure, static routers are complex.
2. Less overhead
3. No complexity.

Disadvantage of RIP v1:

1. Bandwidth utilization is very high as broadcast for every 30 seconds.
2. It works only on hop count.
3. It is not scalable as hop count is only 15. If there will be requirement of more routers in the network it would be a problem.
4. Convergence is very slow, wastes a lot of time in finding alternate path. [\[1\]](#)

RIP VERSION 2

Advantages of RIP v2:

1. It's a standardized protocol.
2. It's VLSM compliant.
3. Provides fast convergence.
4. It sends triggered updates when the network changes.
5. Works with snapshot routing - making it ideal for dial networks.

Disadvantage of RIP v2:

There lie some disadvantages as well:

1. Max hop count of 15, due to the 'count-to-infinity' vulnerability.
2. No concept of neighbours.
3. Exchanges entire table with all neighbours every 30 seconds (except in the case of a triggered update). [\[2\]](#)

CHAPTER 4

FEATURES OF RIP

- Periodic Network Updates:**

RIP routers share routing information every 30 seconds to keep routing tables updated. This ensures network routes are refreshed but can increase network traffic in larger environments.

- Broadcasted Routing Information:**

RIP uses broadcast messages to send routing updates to all routers within the same segment. While simple, this can lead to inefficiencies, as all routers receive updates even if only some need them.

- Full Routing Table Updates:**

Instead of incremental changes, RIP sends its entire routing table in each update. This design simplifies protocol processing but can consume considerable bandwidth, especially in networks with many routes.

- Routing on Rumours (Trust in Neighbouring Routers):**

RIP operates by trusting the routing information received from neighbouring routers, a concept known as "routing on rumours." While this simplifies the protocol, it may lead to issues like routing loops if incorrect data is shared.

- Maximum Hop Count Limit:**

RIP limits the hop count to 15, considering anything beyond that as unreachable to prevent routing loops. This limitation makes RIP suitable for small to medium-sized networks only.

- Distance Vector Algorithm:**

RIP uses a simple distance-vector algorithm where routers periodically share their routing tables with neighbors, enabling easy computation of the shortest path based on hop count.

- Support for RIPv2 Enhancements:**

RIPv2 improves on RIP by supporting subnet masks, multicast updates, and basic authentication, addressing some of RIP's original limitations and enhancing security. ^[3]

CHAPTER 5

RIP UPDATING ALGORITHM

The RIP updating algorithm processes incoming response messages to keep a router's routing table accurate^[6]. Here's how it works:

1. **Receive a Response RIP Message:**

Upon receiving a response from a neighbouring router, the router reviews each advertised route (destination and hop count).

2. **Increment the Hop Count:**

The router adds one hop to each advertised hop count, accounting for its own connection to the neighbour.

3. **Process Each Destination:**

For each advertised destination:

- **If the destination isn't in the routing table**, the router adds it, recording the destination, hop count, and next-hop router.
- **If the destination is already in the table**, two checks are performed:
 - **Same Next-Hop Router:** If the next-hop matches, the router updates the entry to reflect the latest hop count.
 - **Different Next-Hop and Smaller Hop Count:** If the next-hop differs but the advertised hop count is smaller, the router updates the route to use the new, shorter path.

4. **Finish Update:**

The router completes the update, ready to handle new messages.

CHAPTER 6

TOPOLOGY

One of the topologies for used making ring RIP configuration has been used here. *Figure 6.1* illustrates the RIP ring configuration with six routers and two end devices in a ring topology. The figure shows the initial network setup, followed by updates as routing information is exchanged, and finally the stabilized network with no further changes in routing tables. ^[5]



Figure 6.1 Components Used in RIP Ring Configuration

In *Figure 6.2* we see the depiction of a RIP ring topology configured with RIP version 1. It shows the connection status of all the routers after initial setup.

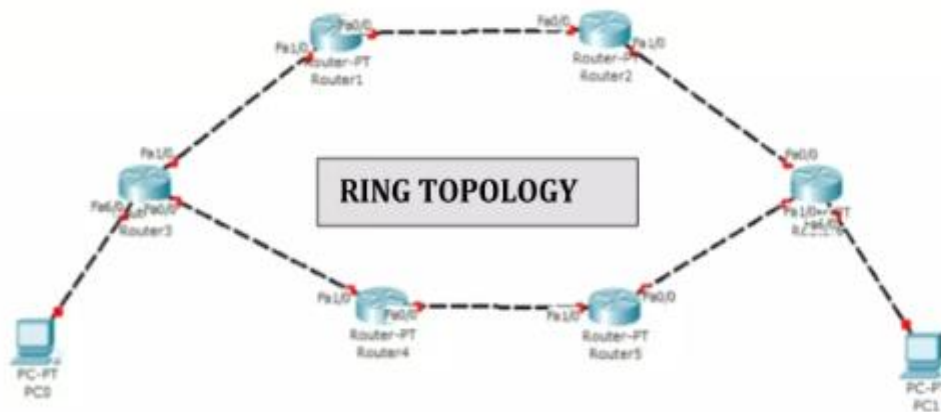


Figure 6.2 RIP Ring Topology Configuration

This is another example of RIP Topology .

Figure 6.3 shows a network with two routers and four PCs using a Serial DCE connection and copper crossover cable. The figure highlights the initial IP configurations, which will be followed by the exchanged updates and updated forwarding tables.

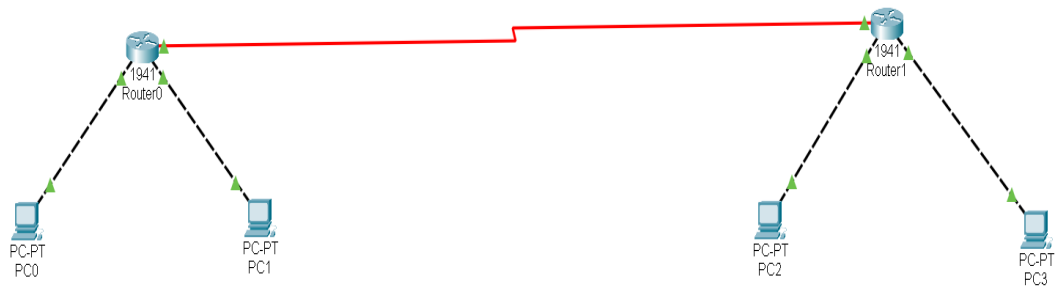


Figure 6.3 Simple RIP Topology

We now assign every interface with different IP address with their subnet mask.

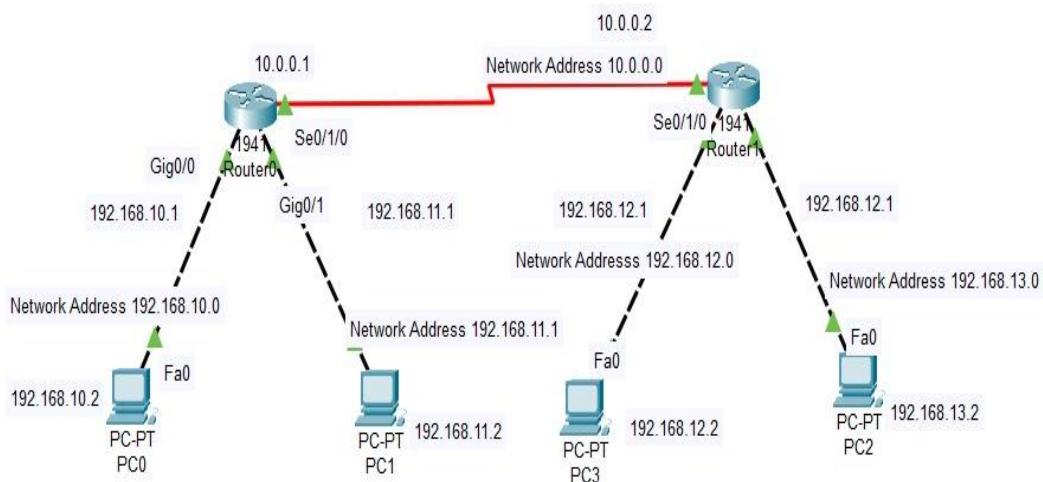


Figure 6.4 Fully Functioning RIP Network

Figure 6.4 demonstrates a fully functioning RIP network. After manually configuring RIP, routing updates are sent, and connectivity is verified by pinging all IP addresses. The routers initially show only direct connections, then display all network paths after RIP configuration. ^[5]

CHAPTER 7

RIP Configuration

RIP is a standardized Distance Vector protocol, designed for use on smaller networks. RIP was one of the first true Distance Vector routing protocols, and is supported on a wide variety of systems.

RIP adheres to the following Distance Vector characteristics:

- RIP sends out periodic routing updates (every 30 seconds)
- RIP sends out the full routing table every periodic update
- RIP uses a form of distance as its metric (in this case, hopcount)
- RIP uses the Bellman-Ford Distance Vector algorithm to determine the best “path” to a particular destination

Other characteristics of RIP include:

- RIP supports IP and IPX routing.
- RIP utilizes UDP port 520
- RIP routes have an administrative distance of 120.
- RIP has a maximum hop count of 15 hops.

Any network that is 16 hops away or more is considered unreachable to RIP, thus the maximum diameter of the network is 15 hops. A metric of 16 hops in RIP is considered a poison route or infinity metric.

If multiple paths exist to a particular destination, RIP will load balance between those paths (by default, up to 4) only if the metric (hopcount) is equal. RIP uses a round-robin system of load-balancing between equal metric routes, which can lead to pinhole congestion.

For example, two paths might exist to a particular destination, one going through a 9600 baud link, the other via a T1. If the metric (hopcount) is equal, RIP will load-balance, sending an equal amount of traffic down the 9600 baud link and the T1. This will (obviously) cause the slower link to become congested.

RIP Versions

RIP has two versions, Version 1 (RIPv1) and Version 2 (RIPv2).

RIPv1 (RFC 1058) is classful, and thus does not include the subnet mask with its routing table updates. Because of this, RIPv1 does not support Variable Length Subnet Masks (VLSMs). When using RIPv1, networks must be contiguous, and subnets of a major network must be configured with identical subnet masks. Otherwise, route table inconsistencies (or worse) will occur.

RIPv1 sends updates as broadcasts to address 255.255.255.255.

RIPv2 (RFC 2543) is classless, and thus does include the subnet mask with its routing table updates. RIPv2 fully supports VLSMs, allowing discontinuous networks and varying subnet masks to exist.

Other enhancements offered by RIPv2 include:

- Routing updates are sent via multicast, using address 224.0.0.9
- Encrypted authentication can be configured between RIPv2 routers
- Route tagging is supported (explained in a later section)

RIPv2 can interoperate with RIPv1. By default:

- RIPv1 routers will send only Version 1 packets
- RIPv1 routers will receive both Version 1 and 2 updates
- RIPv2 routers will both send and receive only Version 2 updates

We can control the version of RIP a particular interface will “send” or “receive.”

Unless RIPv2 is manually specified, a Cisco will default to RIPv1 when configuring RIP. [\[5\]](#) [\[7\]](#)

7.i RIPv1 Configuration

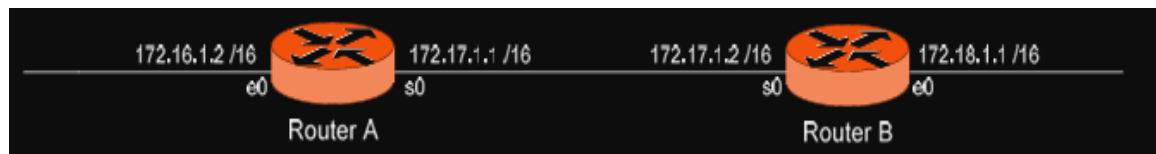


Figure 7.1 Two Router Simple RIPv1 Configuration

Figure 7.1 This is an example of RIPv1 where the networks are contiguous, and the subnet masks remain consistent. It is constructed using the following example:

Routing protocol configuration occurs in Global Configuration mode. On Router A, to configure RIP, we would type:

```
Router(config)# router rip
Router(config-router)# network 172.16.0.0
Router(config-router)# network 172.17.0.0
```

The first command, `router rip`, enables the RIP process.

The network statements tell RIP which networks you wish to advertise to other RIP routers. [\[1\]](#)

To configure Router B:

```
Router(config)# router rip
Router(config-router)# network 172.17.0.0
Router(config-router)# network 172.18.0.0
```

The routing table on Router A will look like:

```
RouterA# show ip route
```

<eliminated irrelevant header>

Gateway of last resort is not set

C 172.16.0.0 is directly connected, Ethernet0

C 172.17.0.0 is directly connected, Serial0

R 172.18.0.0 [120/1] via 172.17.1.2, 00:00:00, Serial0

The routing table on Router B will look like:

```
RouterB# show ip route
```

<eliminated irrevelevant header>

Gateway of last resort is not set

C 172.17.0.0 is directly connected, Serial0

C 172.18.0.0 is directly connected, Ethernet0

R 172.16.0.0 [120/1] via 172.17.1.1, 00:00:00, Serial0

7.ii Limitations of RIPv1

The example on the previous page works fine with RIPv1, because the networks are contiguous and the subnet masks are consistent. Consider the following example:

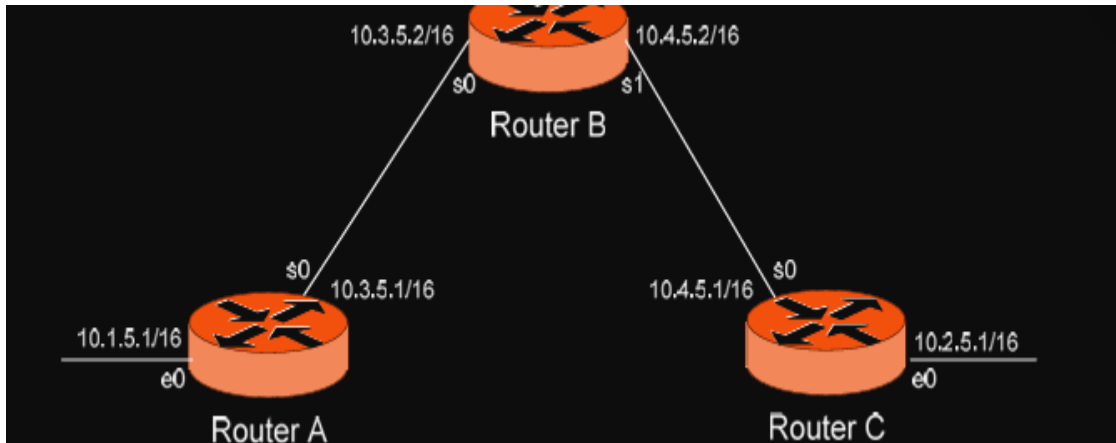


Figure 7.2 Three Router RIPv1 Configuration

The scenario in *Figure 7.2* continues to work with RIPv1, even though the major 10.0.0.0 network has been subnetted. As shown in *Figure 7.2*, the subnets are contiguous, meaning they belong to the same major network, and they share the same subnet mask.

When Router A sends a RIPv1 update to Router B via Serial0, it will not include the subnet mask for the 10.1.0.0 network. However, because the 10.3.0.0 network is in the same major network as the 10.1.0.0 network, it will not summarize the address. The route entry in the update will simply state “10.1.0.0”.

Router B will accept this routing update, and realize that the interface receiving the update (Serial0) belongs to the same major network as the route entry of 10.1.0.0. It will then apply the subnet mask of its Serial0 interface to this route entry.

Router C will similarly send an entry for the 10.2.0.0 network to Router B. Router B's routing table will thus look like:

```
RouterB# show ip route
Gateway of last resort is not set
10.0.0.0/16 is subnetted, 4 subnets
C   10.3.0.0 is directly connected, Serial0
C   10.4.0.0 is directly connected, Serial1
R   10.1.0.0 [120/1] via 10.3.5.1, 00:00:00, Serial0
```

Limitations of RIPv1 (continued)

Consider the following, slightly altered, example:

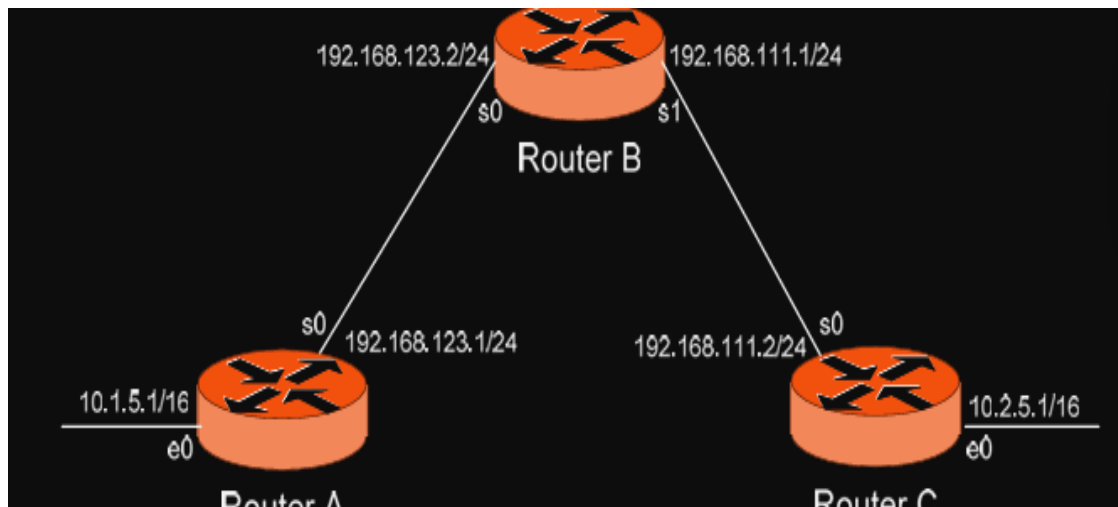


Figure 7.3 Updated 3 Router RIPv1 Configuration

In *Figure 7.3* that RIPv1 is configured correctly on all routers. Notice that our networks are no longer contiguous. Both Router A and Router C contain subnets of the 10.0.0.0 major network (10.1.0.0 and 10.2.0.0 respectively).

Separating these networks now are two Class C subnets (192.168.123.0 and 192.168.111.0).

Why is this a problem? Again, when Router A sends a RIPv1 update to Router B via Serial, it will not include the subnet mask for the 10.1.0.0 network. Instead, Router A will consider itself a border router, as the 10.1.0.0 and 192.168.123.0 networks do not belong to the same major network. Router A will summarize the 10.1.0.0/16 network to its classful boundary of 10.0.0.0/8.

Router B will accept this routing update, and realize that it does not have a directly connected interface in the 10.x.x.x scheme. Thus, it has no subnet mask to apply to this route. Because of this, Router B will install the summarized 10.0.0.0 route into its routing table.

Router C, similarly, will consider itself a border router between networks 10.2.0.0 and 192.168.111.0. Thus, Router C will also send a summarized 10.0.0.0 route to Router B. ^[1]

Router B's routing table will then look like:

```
RouterB# show ip route
```

```
Gateway of last resort is not set
```

```
C 192.168.123.0 is directly connected, Serial0
```

```
C 192.168.111.0 is directly connected, Serial1
```

```
R 10.0.0.0 [120/1] via 192.168.123.1, 00:00:00, Serial0 [120/1] via  
192.168.111.2, 00:00:00, Serial1
```

That's right, Router B now has two equal metric routes to get to the summarized 10.0.0.0 network, one through Router A and the other through Router C. Router B will now load balance all traffic to any 10.x.x.x network between routers A and C. Suffice to say, this is not a good thing.

It gets better. Router B then tries to send routing updates to Router A and Router C, including the summary route of 10.0.0.0/8. Router A's routing table looks like:

```
RouterA# show ip route
```

```
Gateway of last resort is not set
```

```
C 192.168.123.0 is directly connected,
```

```
Serial0 10.0.0.0/16 is subnetted,
```

```
1 subnet C 10.1.0.0 is directly connected, to Ethernet0
```

Router A will receive the summarized 10.0.0.0/8 route from Router B, and will reject it. This is because it already has the summary network of 10.0.0.0 in its routing table, and it's directly connected. Router C will respond exactly the same, and the 10.1.0.0/16 and 10.2.0.0/16 networks will never be able to communicate.

7.iii RIPv2 Configuration

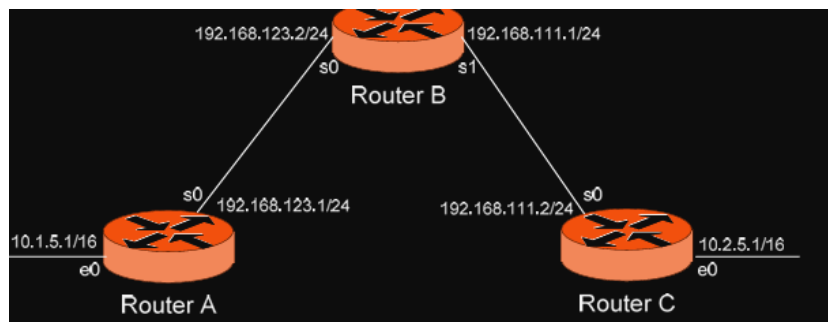


Figure 7.4 RIPv2 Configuration

Figure 7.4 RIPv2 overcomes the limitations of RIPv1 by including the subnet mask in its routing updates. By default, Cisco routers will use RIPv1. To change to Version 2, we must use commands

```
Router(config)# router rip
```

```
Router(config-router)# version 2
```

Thus, the configuration of Router A would be:

```
RouterA(config)# router rip
```

```
RouterA(config-router)# version 2
```

```
RouterA(config-router)# network 10.0.0.0
```

```
RouterA(config-router)# network 192.168.123.0
```

Despite the fact that RIPv2 is a classless routing protocol, we still specify networks at their classful boundaries, without a subnet mask.

However, when Router A sends a RIPv2 update to Router B via Serial0, by default it will still summarize the 10.1.0.0/16 network to 10.0.0.0/8. Again, this is because the 10.1.0.0 and 192.168.123.0 networks do not belong to the same major network. Thus, RIPv2 acts like RIPv1 in this circumstance unless you disable auto summarization:

```
RouterA(config)# router rip
```

```
RouterA(config-router)# version 2
```

```
RouterA(config-router)# no auto-summary
```

The no auto-summary command will prevent Router A from summarizing the 10.1.0.0 network. Instead, Router A will send an update that includes both the subnetted network (10.1.0.0) and its subnet mask (255.255.0.0). ^[2]

7.iv RIP Timers

RIP has four basic timers:

Update Timer (default **30 seconds**) – indicates how often the router will send out a routing table update.

Invalid Timer (default **180 seconds**) – indicates how long a route will remain in a routing table before being marked as invalid, if no new updates are heard about this route. The invalid timer will be reset if an update is received for that particular route before the timer expires.

A route marked as invalid is not immediately removed from the routing table. Instead, the route is marked (and advertised) with a metric of 16, indicating it is unreachable, and placed in a hold-down state.

Hold-down Timer (default **180 seconds**) – indicates how long RIP will “suppress” a route that it has placed in a hold-down state. RIP will not accept any new updates for routes in a hold-down state, until the hold-down timer expires. ^[6]

A route will enter a hold-down state for one of three reasons:

- The invalid timer has expired.
- An update has been received from another router, marking that route with a metric of 16 (or unreachable).
- An update has been received from another router, marking that route with a higher metric than what is currently in the routing table. This is to prevent loops.

Flush Timer (default 240 seconds) – indicates how long a route can remain in a routing table before being flushed, if no new updates are heard about this route. The flush timer runs **concurrently with the invalid timer**, and thus will flush out a route 60 seconds after it has been marked invalid.

RIP timers must be identical on all routers on the RIP network, otherwise massive instability will occur. ^[6]

RIP Timers Configuration and Example (continued)

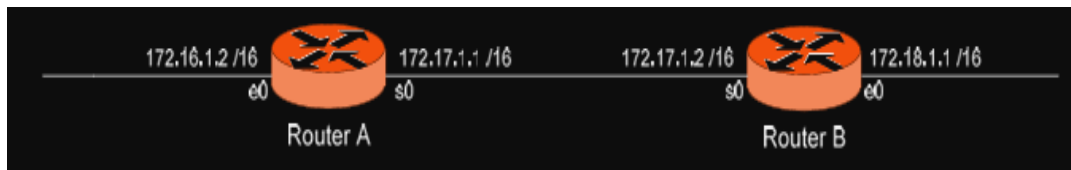


Figure 7.5 RIP Timers Example

Consider the *Figure 7.5*. Router A receives a RIP update from Router B that includes network 172.18.0.0. Router A adds this network to its routing table:

```
RouterA# show ip route
```

```
Gateway of last resort is not set
```

```
C 172.16.0.0 is directly connected, Ethernet0
```

```
C 172.17.0.0 is directly connected, Serial0
```

```
R 172.18.0.0 [120/1] via 172.17.1.2, 00:00:00, Serial0
```

Immediately, Router A sets an invalid timer of 180 seconds and flush timer of 240 seconds to this route, which run concurrently. If no update for this route is heard for 180 seconds, several things will occur:

- The route is marked as invalid in the routing table.
- The route enters a hold-down state (triggering the hold-down timer).
- The route is advertised to all other routers as unreachable.

The hold-down timer runs for 180 seconds after the route is marked as invalid. The router will not accept any new updates for this route until this hold-down period expires.

If no update is heard at all, the route will be removed from the routing table once the flush timer expires, which is 60 seconds after the route is marked as invalid. Remember that the invalid and flush timers run concurrently.

To configure the RIP timers:

```
Router(config)# router rip
```

```
Router(config-router)# timers basic 20 120 120 160
```

The timers basic command allows us to change the update (20), invalid (120), hold-down (120), and flush (240) timers. To return the timers back to their defaults: Router(config-router)# no timers basic.

7.v RIP Loop Avoidance Mechanisms

RIP, as a Distance Vector routing protocol, is susceptible to loops.

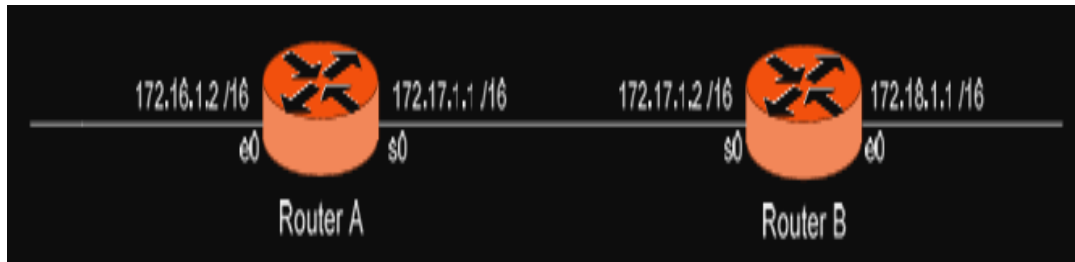


Figure 7.6 RIP Loop Avoidance

Let's assume no loop avoidance mechanisms are configured on either router, this is shown in *Figure 7.6*, if the 172.18.0.0 network fails, Router B will send out an update to Router A within 30 seconds (whenever its update timer expires) stating that route is unreachable (metric = 16).

But what if an update from Router A reaches Router B before this can happen? Router A believes it can reach the 172.18.0.0 network in one hop (through Router B). This will cause Router B to believe it can reach the failed 172.18.0.0 network in two hops, through Router A. Both routers will continue to increment the metric for the network until they reach a hop count of 16, which is unreachable. This behavior is known as counting to infinity. [4]

How can we prevent this from happening? There are several loop avoidance mechanisms:

Split-Horizon – Prevents a routing update from being sent out the interface it was received on. In our above example, this would prevent Router A from sending an update for the 172.18.0.0 network back to Router B, as it originally learned the route from Router B. Split-horizon is enabled by default on Cisco Routers.

Route-Poisoning – Works in conjunction with split-horizon, by triggering an automatic update for the failed network, without waiting for the update timer to expire. This update is sent out all interfaces with an infinity metric for that network.

Hold-Down Timers – Prevents RIP from accepting any new updates for routes in a hold-down state, until the hold-down timer expires. If Router A sends an update to Router B with a higher metric than what is currently in Router B's routing table, that route will be placed in a hold-down state. (Router A's metric for the 172.18.0.0 network is 1; while Router B's metric is 0). [3]

7.vi RIP Passive Interfaces

It is possible to control which router interfaces will participate in the RIP process.

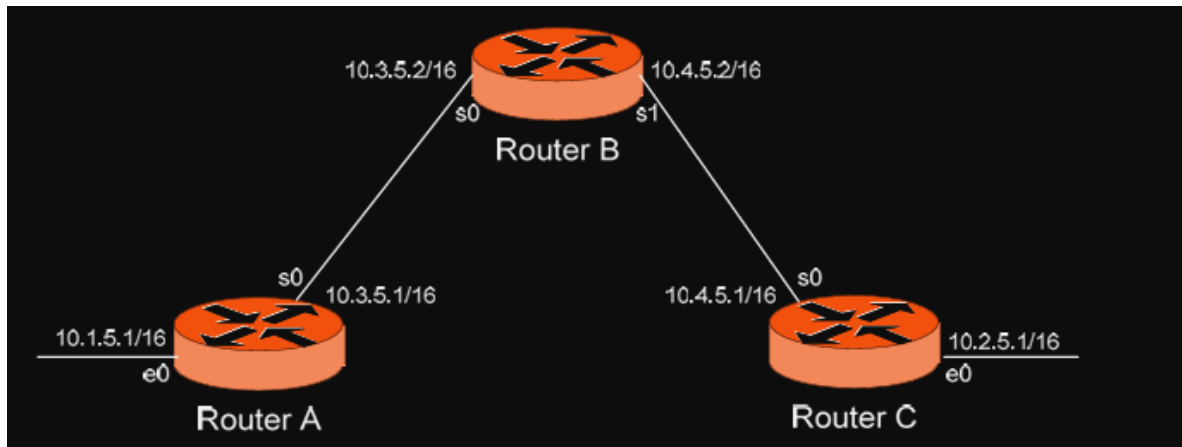


Figure 7.7 RIP Passive Interface

Consider the scenario in *Figure 7.7*, we can see Router C does not want to participate in the RIP domain. However, it still wants to listen to updates being sent from Router B, just not send any updates back to Router B:

```
RouterC(config)# router rip
RouterC(config-router)# network 10.4.0.0
RouterC(config-router)# network 10.2.0.0
RouterC(config-router)# passive-interface s0
```

The passive-interface command will prevent updates from being sent out of the Serial0 interface, but Router C will still receive updates on this interface.

[\[4\]](#) [\[6\]](#)

We can configure all interfaces to be passive using the passive-interface default command, and then individually use the no passive-interface command on the interfaces we do want updates to be sent out:

```
RouterC(config)# router rip
RouterC(config-router)# network 10.4.0.0
RouterC(config-router)# network 10.2.0.0
RouterC(config-router)# passive-interface default
RouterC(config-router)# no passive-interface e0
```

7.vii RIP Neighbours

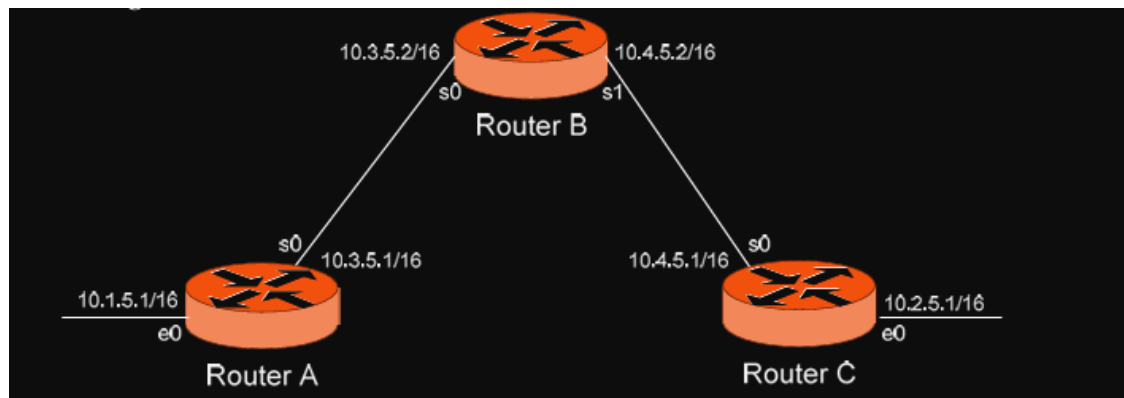


Figure 7.8 RIP Neighbours

Recall that RIPv1 sends out its updates as broadcasts, whereas RIPv2 sends out its updates as multicasts to the 224.0.0.9 address. In *Figure 7.8* we configure specific RIP neighbor commands, which will allow us to unicast routing updates to those neighbors. ^[3]

On Router B:

```
RouterB(config)# router rip
RouterB(config-router)# network 10.3.0.0
RouterB(config-router)# network 10.4.0.0
RouterB(config-router)# neighbor 10.3.5.1
RouterB(config-router)# neighbor 10.4.5.1
```

Router B will now unicast RIP updates to Router A and Router C.

However, Router B will still broadcast (if RIPv1) or multicast (if RIPv2) its updates, in addition to sending unicast updates to its neighbors. In order to prevent broadcast/multicast updates, we must also use passive interfaces;

```
RouterB(config)# router rip
RouterB(config-router)# passive-interface s0
RouterB(config-router)# passive-interface s1
RouterB(config-router)# neighbor 10.3.5.1
RouterB(config-router)# neighbor 10.4.5.1
```

The `passive-interface` commands prevent the updates from being broadcasted or multicasted. The `neighbor` commands still allow unicast updates to those specific neighbors.

7.viii RIPv2 Authentication

RIPv2 supports authentication to secure routing updates.

The first step is creating a shared authentication key that must be identical on both routers. This is accomplished in global configuration mode:

```
RouterA(config)# key chain MYCHAIN
RouterA(config-keychain)# key 1
RouterA(config-keychain-key)# key-string MYPASSWORD

RouterB(config)# key chain MYCHAIN
RouterB(config-keychain)# key 1
RouterB(config-keychain-key)# key-string MYPASSWORD
```

The first command creates a key chain called MYCHAIN. We must then associate a key to our keychain. Then we actually configure the shared key using the key-string command.

We then apply our key chain to the interface connecting to the other router:

```
RouterA(config)# interface s0
RouterA(config-if)# ip rip authentication key-chain MYCHAIN

RouterB(config)# interface s0
RouterB(config-if)# ip rip authentication key-chain MYCHAIN
```

If there was another router off of Router B's Ethernet port, we could create a separate key chain with a different key-string. Every router on the RIP domain does not need to use the same key chain, only interfaces directly connecting two (or more) routers. [\[2\]](#)

The final step in configuring authentication is identifying which encryption to use. By default, the key is sent in clear text:

```
RouterA(config)# interface s0
RouterA(config-if)# ip rip authentication mode text
```

Or we can use MD5 encryption for additional security:

```
RouterA(config)# interface s0
RouterA(config-if)# ip rip authentication mode md5
```

Whether text or MD5 is used, it must be the same on both routers.

7.ix Altering RIP's Metric

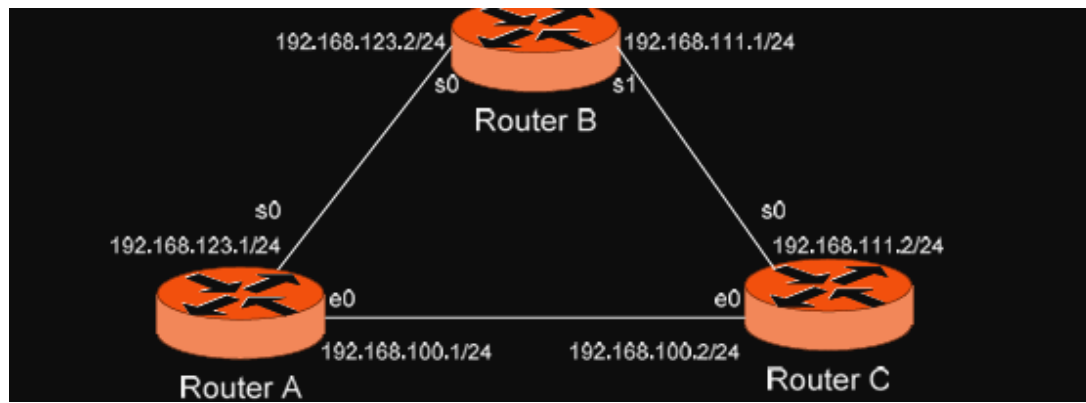


Figure 7.9 Looped RIP Metric

Consider the above example. In *Figure 7.9* we see Router B has two paths to get to the 192.168.100.0 network, via Router A and Router C. Because the metric is equal (1 hop), Router B will load balance between these two paths.

What if we wanted Router B to only go through Router A, and use Router C only as a backup? To accomplish this, we can adjust RIP's metric to make one route more preferred than the other. ^[6]

The first step is creating an access-list on Router B that defines which route we wish to alter:

```
RouterB(config)# ip access-list standard MYLIST
RouterB(config-std-nacl)# permit 192.168.100.0 0.0.0.255
```

Next, we tell RIP how much to offset this route if received by Router C: .

```
RouterB(config)# router rip
RouterB(config-router)# offset-list MYLIST in 4 s1
```

We specify an offset-list pointing to our access list named MYLIST. We will increase the routing metric by 4 for that route coming inbound to interface Serial 1.

Thus, when Router C sends an update to Router B for the 192.168.100.0 network, Router B will increase its metric of 1 hop to 5 hops, thus making Router A's route preferred.

We could have also configured Router C to advertise that route with a higher metric (notice the out in the offset-list command):

```
RouterC(config)# ip access-list standard MYLIST
RouterC(config-std-nacl)# permit 192.168.100.0 0.0.0.255
RouterC(config)# router rip
RouterC(config-router)# offset-list MYLIST out 4 s0
```

7.x Interoperating between RIPv1 and RIPv2

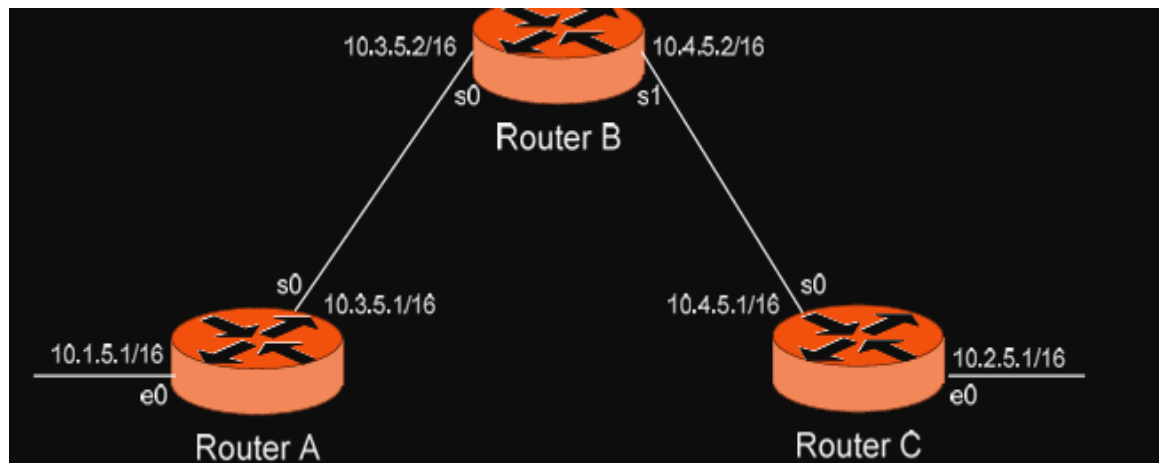


Figure 7.10 Interoperation Between RIPv1 and v2

In *Figure 7.10* we can see the start of Interoperation in between Recall that, with some configuration, RIPv1 and RIPv2 can interoperate.

By default:

- RIPv1 routers will send only Version 1 packets
- RIPv1 routers will receive both Version 1 and 2 updates
- RIPv2 routers will both send and receive only Version 2 updates

If Router A is running RIPv1, and Router B is running RIPv2, some additional configuration is necessary. ^[3]

Either we must configure Router A to send Version 2 updates:

```
RouterA(config)#interface s0
RouterA(config-if)# ip rip send version 2
```

Or configure Router B to accept Version 1 updates.

```
RouterB(config)# interface s0
RouterB(config-if)#ip rip receive version 1
```

Notice that this is configured on an interface. Essentially, we're configuring the version of RIP on a per-interface basis.

We can also have an interface send or receive both versions simultaneously:

```
RouterB(config)# interface s0
RouterB(config-if)# ip rip receive version 1 2
```

We can further for RIPv2 to send broadcast updates, instead of multicasts:

```
RouterB(config)# interface s0
RouterB(config)# ip rip v2-broadcast
```

7.xi Triggering RIP Updates

On point-to-point interfaces, we can actually force RIP to only send routing updates if there is a change:

```
RouterB(config)# interface s0.150 point-to-point
RouterB(config-if)# ip rip triggered
```

Again, this is only applicable to point-to-point links. We cannot configure RIP triggered updates on an Ethernet network.

Troubleshooting RIP [\[5\]](#)

Various troubleshooting commands exist for RIP.

To view the IP routing table:

```
Router# show ip route
<eliminated irrelevant header>
Gateway of last resort is not set
```

```
C 172.16.0.0 is directly connected, Ethernet0
C 172.17.0.0 is directly connected, Serial0
C 172.18.0.0 [120/1] via 172.17.1.2, 00:00:15, Serial0
C 192.168.123.0 [120/1] via 172.16.1.1, 00:00:00, Ethernet0
```

To view a specific route within the IP routing table:

```
Router# show ip route 172.18.0.0
Routing entry for 172.18.0.0/16
Known via "rip", distance 120, metric 1
Last update from 172.17.1.2 on Serial 0, 00:00:15
```

To debug RIP in real time:

```
Router# debug ip rip
```

7.xii Troubleshooting RIP

To view information specific to the RIP protocol:

```
Router# show ip protocols
```

```
Routing Protocol is "rip"
```

```
Sending updates every 30 seconds, next due in 20 seconds
Invalid after 180 seconds, hold down 180, flushed after 240
Outgoing update filter list for all interfaces is not set
Incoming update filter list for all interfaces is not set
Incoming routes will have 4 added to metric if on list 1
```

```
Redistributing: connected, static, rip
```

```
Default version control: send version 1, receive any version
```

```
Interface table      Send      Recv    Triggered RIP
```

```
Key-chain Ethernet0    1          1        2
Serial0                1 2        1        2
```

```
Automatic network summarization is in effect
```

```
Maximum path: 4
```

```
Routing for Networks:
```

```
.      172.16.0.0
.      172.17.0.0
```

```
Routing Information Sources:
```

```
Gateway      Distance      Last Update
172.17.1.2    120           00:00:17
```

```
Distance: (default is 120)
```

This command provides us with information on RIP timers, on the RIP versions configured on each interface, and the specific networks RIP is advertising^[5]

To view all routes in the RIP database, and not just the entries added to the routing table:

```
Router# show ip rip database
```

```
7.0.0.0/8    auto-summary
```

```
7.0.0.0/8
```

```
[5] via 172.16.1.1, 00:00:06, Ethernet0
```

```
172.16.0.0/16    directly connected, Ethernet0
```

```
172.17.0.0/16    directly connected, Serial0
```

CHAPTER 8

SCENARIO

Scenario-1

University Campus Network Simulation

Background:

We can consider a university campus with a number of departments located in several buildings, each with its own subnet. Campus IT services face difficulties in routing between these departments so that students and staff may access shared resources across all the departments like course databases, research servers, and online learning portals, as well as a network connection to the central data centre of the university.

RIP can be implemented by the IT department to manage the interdepartmental traffic. The justification for employing the protocol is because of simplicity and the fact that the network does not need something that would demand a very complex routing design, though it demands effective and reliable packet delivery between the various departments. ^[7]

Details:

1. **Network Segmentation:** Each department should have its subnet, and the routers in every building are networked together to constitute a mesh using RIP. This will enable each department to communicate with any other department without any hassle.
2. **Routing Updates and Hop Counting:** RIP updates should flood every 30 seconds across the network. At this time, the routers share the best path to reach other buildings - of course, by far the shortest hop count. This frequent update cycle will have helped ensure that the routers can maintain current knowledge of the alternatives.
3. **Failure Recovery:** If one link fails (such as a connection between two academic buildings fails), RIP will detect the link change, update the routing table, and, if an alternate path is available, re-route the traffic while at the same time the IT team repairs the link.
4. **Traffic Management:** Configuring the administrative distance of RIP would allow the IT team to favour some of the routes, like paths to the critical resources that are within the data centre. Since, in addition, only hop count is considered and not any

others, RIP may fail to consistently optimize the path for the best quickest paths, with congested links, a condition acceptable with the relatively lightly trafficked campus. [7]

Expected Outcome:

RIP allows the campus to provide reliable routing for interdepartmental access without requiring a lot of network administration. However, while the campus network scale will eventually be too large for the scalability of RIP-and potentially subject to route loops-the IT team realizes that they will likely have to move to protocols like OSPF if the campus does grow this large. This use of RIP serves the campus's current needs adequately, allowing departments to communicate seamlessly while keeping network administration reasonable.

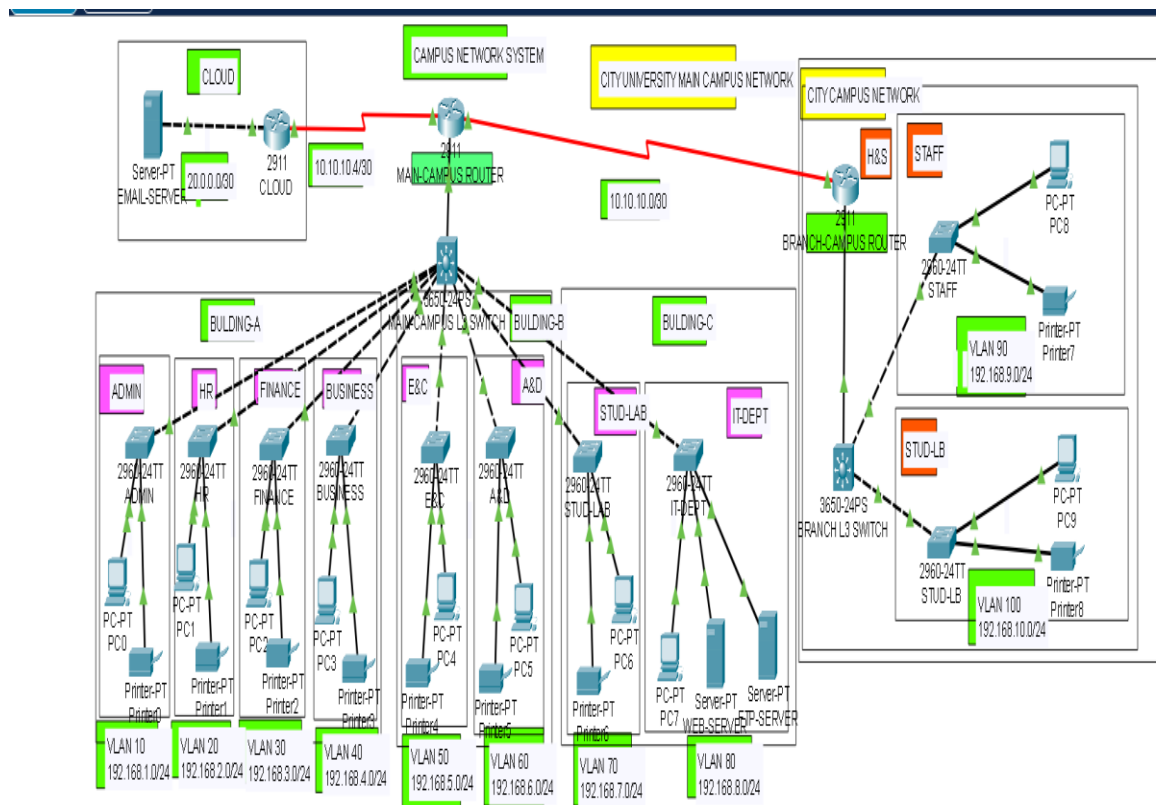


Figure 8.1 Sample Campus Network Simulation

In *Figure 8.1* RIP allows the campus to provide reliable routing for interdepartmental access without requiring a lot of network administration. This use of RIP serves the campus's current needs adequately, allowing departments to communicate seamlessly while keeping network administration reasonable.

Scenario-2

SRM Hostel SANNASI Network with RIP Version 1

Background:

SRM Hostel SANNASI has multiple floors-spanning with multiple rooms, along with recreational areas all connected with a LAN based connection. Each room has all different routers with their own Wi-Fi settings. Therefore, the network should be easy to manage routing so it will provide better access to shared resources and services. Since there's no need for subnet masking, this network was developed with RIP Version 1 for this simplicity for these requirements.

Solution Used:

RIP-1 is executed on the routers that join every floor's LAN. This provides elementary inter-floor communication based upon frequent updates to each of the routing tables of the routers. The use of the hop count metric of RIP-1 ensures good routing, and this small network does not require any IT experience in order to be kept up to date with routing table updates via broadcast.

Implementation Details:

1. Subnet Setup:

There is a different subnet and router on each floor where they use the implementation of RIP-1 for transmitting basic routing information so a full view of the network can be developed.

2. Routing Table Refreshes: The routers periodically transmit their tables every 30 seconds to learn any other routes.

3. Metric Value Hop Count: Routes are based on the fewest hops and ensure efficient data flow between floors.

4. Dynamic Adjustment and Failure Recovery: RIP-1 can automatically adjust to network changes or outages by removing unreachable routes and routing traffic accordingly.

5. Constraints: RIP-1 being classful in nature has many constraints. Its broadcasting would add a lot of traffic. Increasing the network size requires that RIP-2 or OSPF be implemented.

Outcome:

RIP-1 allows SRM Hostel SANNASI to have a multi-story, multi-floor network. Upgrades are done automatically while routing is quite straightforward. But if the network was to grow or one feels the need for enhanced capability, then classless may be implemented later on.

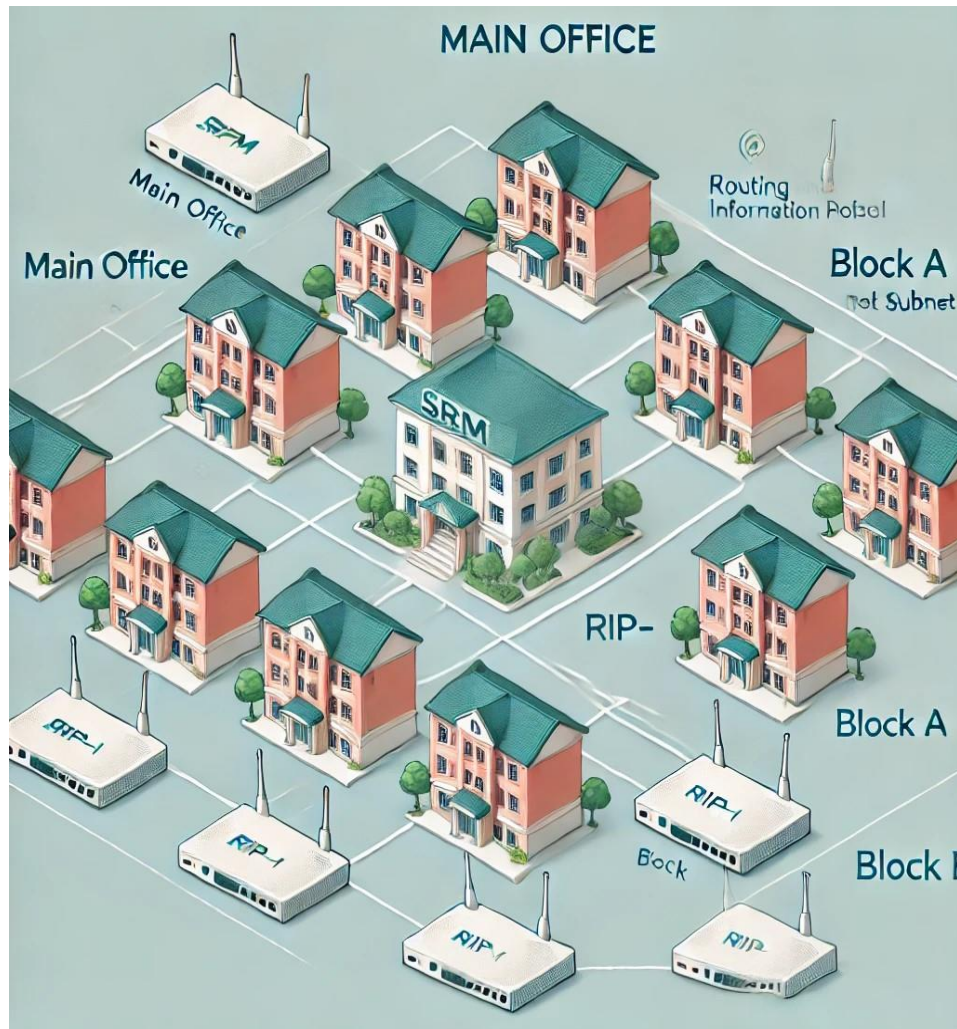


Figure 8.2 Sample Hostel Network Layout

In *Figure 8.2* displays the network layout of Hostel network configuration with RIP implemented. It illustrates updated routing tables, network paths established by RIP, and the stabilized configuration once all routing information has converged.

CONCLUSION

This project delved into the intricacies of the Routing Information Protocol (RIP); a fundamental distance-vector routing protocol used in network environments. We explored its two primary versions, RIPv1 and RIPv2, highlighting their key features, limitations, and advancements.

RIP, while simple to configure, is inherently limited by its hop count metric and the potential for routing loops. However, its classless variant, RIPv2, introduced significant improvements, including support for VLSM and authentication.

Through a comprehensive analysis of RIP's operation, we gained insights into its periodic updates, routing table exchanges, and convergence mechanisms. We also examined its role in network topology, its interaction with other protocols, and its impact on network performance.

By understanding the strengths and weaknesses of RIP, network administrators can make informed decisions about its deployment in appropriate network scenarios. While RIP may not be suitable for large-scale, complex networks, it remains a valuable tool for smaller networks and specific use cases.

REFERENCES

1. RFC 1058
“Routing Information Protocol
Editor: Charles Hedrick, Network Working Group, 1988.
The original Request for Comments (RFC) document for RIP,
describing the protocol’s specifications and early implementation
2. RFC 2453
RIP Version 2
Editors: Gary Malkin, Network Working Group, 1998.
This document details the improvements made in RIPv2, such as
support for subnet masks and multicasting.
3. "Data Communication and Networking",5th ed.,2010
Behrouz A. Forouzan
4. " Data Communication and Networks" 2016
Bhushan Trivedi
5. "CCNA Study Guide",7th ed.,2011
Todd Lammle
6. “Routing Protocols”
Risala Tasin Khan ,Associate Professor IIT, JU
7. “Campus-Network-System-Using-Cisco-Packet-Tracer”
<https://github.com/Anas436/Campus-Network-System-Using-Cisco-Packet-Tracer> ,
Md. Anas Mondol

AADIT VINAYAK RA2211029010012



Networking
cisco Academy

Certificate of Course Completion

AADIT VINAYAK

has successfully achieved student level credential for completing the Networking Basics course.

The student was able to proficiently:

- Explain important concepts in network communication, network types, components, and connections.
- Explain the importance of standards and protocols in network communications.
- Explain how communication occurs on Ethernet networks.
- Explain the features of an IP address and IPv4 addresses are used in network communication.
- Explain features of IPv6 addressing.
- Explain how routers connect networks together.
- Use various tools to test and troubleshoot network connectivity.
- Configure an integrated wireless router and wireless client to connect securely to the internet.



Verified

Networking Basics

October 17, 2024



Scan to Verify

Laura Quintana

Laura Quintana
Vice President and General Manager
Cisco Networking Academy



VEDANT PANDEY RA2211029010013



 Networking
CISCO Academy

Certificate of Course Completion

VEDANT PANDEY

has successfully achieved student level credential for completing the Networking Basics course.

The student was able to proficiently:

- Explain important concepts in network communication, network types, components, and connections.
- Explain the importance of standards and protocols in network communications.
- Explain how communication occurs on Ethernet networks.
- Explain the features of an IP address and IPv4 addresses are used in network communication.
- Explain features of IPv6 addressing.
- Explain how routers connect networks together.
- Use various tools to test and troubleshoot network connectivity.
- Configure an integrated wireless router and wireless client to connect securely to the internet.



Scan to Verify

September 22, 2024


Laura Quintana
Vice President and General Manager
Cisco Networking Academy

CHIRANJEEV KUMAR RA2211029010019



 Networking
cisco Academy

Certificate of Course Completion

CHIRANJEEV KUMAR(RA2211029010019)

has successfully achieved student level credential for completing the Networking Basics course.

The student was able to proficiently:

- Explain important concepts in network communication, network types, components, and connections.
- Explain the importance of standards and protocols in network communications.
- Explain how communication occurs on Ethernet networks.
- Explain the features of an IP address and IPv4 addresses are used in network communication.
- Explain features of IPv6 addressing.
- Explain how routers connect networks together.
- Use various tools to test and troubleshoot network connectivity.
- Configure an integrated wireless router and wireless client to connect securely to the internet.



October 04, 2024

Scan to Verify


Laura Quintana
Vice President and General Manager
Cisco Networking Academy