CHAPTER 4

Distance Vector Routing Protocols

Objectives

Upon completion of this chapter, you should be able to answer the following questions:

- Can you identify the characteristics of distance vector routing protocols?
- What is the network discovery process of distance vector routing protocols using Routing Information Protocol (RIP)?
- What are the processes for maintaining accurate routing tables that are used by distance vector routing protocols?
- What are the conditions leading to a routing loop, and can you explain the implications for router performance?
- Which types of distance vector routing protocols are in use today?

Key Terms

This chapter uses the following key terms. You can find the definitions in the Glossary at the end of the book.

Diffusing Update Algorithm (DUAL) *page 194*

split horizon *page 204*

bounded updates *page 208*

triggered update *page 208*

count to infinity *page 212*

route poisoning *page 217*

poison reverse *page 218*

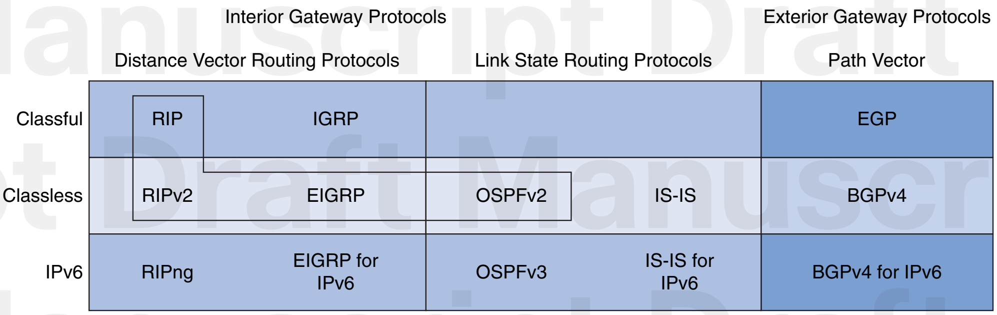
topology table *page 221*

IPX *page 222*

The dynamic routing protocol chapters of this book focus on interior gateway protocols (IGP). As discussed in Chapter 3, "Introduction to Dynamic Routing Protocols," IGPs are classified as either distance vector or link-state routing protocols.

Figure 4-1 shows a chart of the most common IP routing protocols used today. Those that are highlighted will be discussed in this book.

Figure 4-1 Dynamic Routing Protocols



	Interior Gateway Protocols				Exterior Gateway Protocols
	Distance Vector Routing Protocols		Link State Routing Protocols		Path Vector
Classful	RIP	IGRP			EGP
Classless	RIPv2	EIGRP	OSPFv2	IS-IS	BGPv4
IPv6	RIPng	EIGRP for IPv6	OSPFv3	IS-IS for IPv6	BGPv4 for IPv6

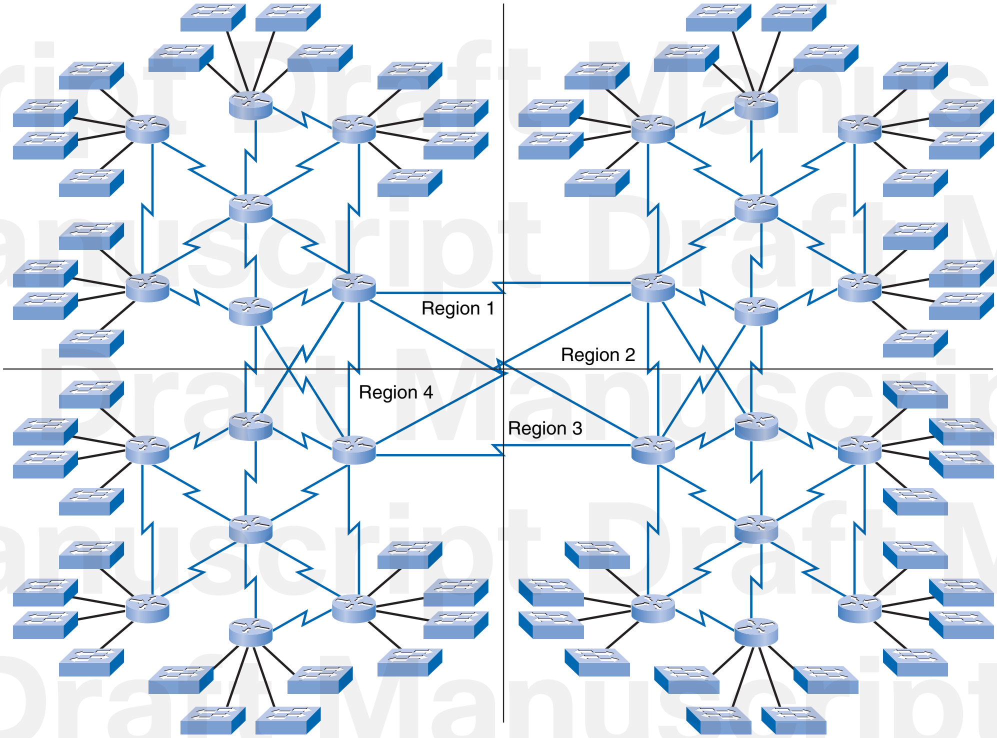
This chapter describes the characteristics, operations, and functionality of distance vector routing protocols. There are advantages and disadvantages to using any type of routing protocol. This chapter covers the operations of distance vector protocols, some of their inherent pitfalls, and the remedies to these pitfalls. Understanding the operation of distance vector routing is critical to enabling, verifying, and troubleshooting these protocols.

Introduction to Distance Vector Routing Protocols

One way to characterize routing protocols is by the type of routing algorithm they use to build and maintain their routing table. By doing this, routing protocols can be differentiated as a distance vector, link-state, or path vector routing protocol. This chapter will introduce you to the characteristics of a distance vector routing protocol. Chapter 10, "Link-State Routing Protocols," will introduce you to link-state routing protocols. Path vector routing protocols are beyond the scope of this book and are discussed in CCNP.

Figure 4-2 shows a network with a moderate number of routers and links.

Figure 4-2 Network That Would Use Dynamic Routing



Dynamic routing protocols help the network administrator overcome the time-consuming and exacting process of configuring and maintaining static routes. For example, can you imagine maintaining the static routing configurations of the 28 routers shown in Figure 4-2? What happens when a link goes down? What happens when that link goes down at 3:00 a.m.? How do you ensure that redundant paths are available 24 hours a day, 7 days a week? Dynamic routing is the most common choice for large networks like the one shown.

Distance vector routing protocols include the following:

- **RIP:** Routing Information Protocol (RIP) was originally specified in RFC 1058. It has the following key characteristics:
  - Hop count is used as the metric for path selection.
  - If the hop count for a network is greater than 15, RIP cannot supply a route to that network.
  - Routing updates are broadcast or multicast every 30 seconds, by default.

- **IGRP:** Interior Gateway Routing Protocol (IGRP) is a proprietary protocol developed by Cisco. IGRP has the following key design characteristics:
  - Bandwidth, delay, load, and reliability are used to create a composite metric.
  - Routing updates are broadcast every 90 seconds, by default.
  - IGRP is the predecessor of EIGRP and is now obsolete.
- **EIGRP:** Enhanced IGRP (EIGRP) is a Cisco-proprietary distance vector routing protocol. EIGRP has these key characteristics:
  - It can perform unequal-cost load balancing.
  - It uses ***Diffusing Update Algorithm (DUAL)*** to calculate the shortest path.
  - There are no periodic updates as with RIP and IGRP. Routing updates are sent only when there is a change in the topology.

Note

There are no RFCs for IGRP or EIGRP, because Cisco never submitted these routing protocols to the Internet Engineering Task Force (IETF) for comments.

RIP and EIGRP will be discussed in more detail in later chapters. IGRP is not discussed and is considered obsolete. IGRP will be referred to for comparison purposes only.

Distance Vector Technology

Distance vector technology is one way to characterize routing protocols based on the type of routing algorithm they use to build and maintain their routing table. The other two methods are link-state and path vector.

Meaning of Distance Vector

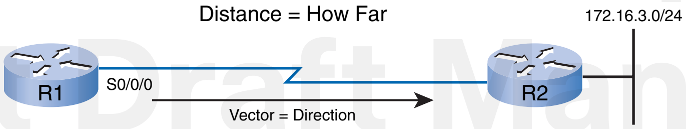
As the name implies, distance vector means that routes are advertised as vectors of distance and direction. Distance is defined in terms of a metric, such as hop count, and direction is simply the next-hop router or exit interface.

A router using a distance vector routing protocol does not have the knowledge of the entire path to a destination network. Instead the router knows only

- The direction in which or interface to which packets should be forwarded
- The distance to the destination network

For example, in Figure 4-3, R1 knows that the distance to reach network 172.16.3.0/24 is one hop and that the direction is out interface S0/0/0 toward R2.

Figure 4-3 Meaning of Distance Vector



For R1, 172.16.3.0/24 is one hop away (distance).
It can be reached through S0/0/0 (vector).

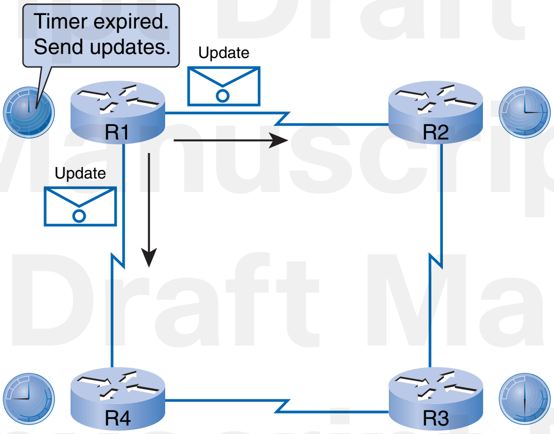
Operation of Distance Vector Routing Protocols

Some distance vector routing protocols call for the router to periodically broadcast the entire routing table to each of its neighbors. This method is inefficient because the updates not only consume bandwidth but also consume router CPU resources to process the updates.

Distance vector routing protocols share certain characteristics. Periodic updates are sent at regular intervals (30 seconds for RIP and 90 seconds for IGRP). Even if the topology has not changed in several days, periodic updates continue to be sent to all neighbors.

Figure 4-4 shows an example of a periodic update. The routing protocol for each router maintains a local timer. When that timer expires, a routing update is sent. In the figure, the timer for R1 has expired. When the local timer on each of the other routers reaches 0, they will also send their respective periodic updates. These periodic updates are entries from all or part of the routing table. This will be examined more thoroughly in Chapter 5, "RIP version 1."

Figure 4-4 Distance Vector Periodic Updates



Neighbors are routers that share a link and are configured to use the same routing protocol. The router is only aware of the network addresses of its own interfaces and the remote network addresses it can reach through its neighbors. It has no broader knowledge of the network topology. Routers using distance vector routing are not aware of the network topology.

Broadcast updates are sent to 255.255.255.255. Neighboring routers that are configured with the same routing protocol will process the updates. Other devices such as host computers will also process the update up to Layer 3 before discarding it. Some distance vector routing protocols use multicast addresses instead of broadcast addresses.

Entire routing table updates are sent, with some exceptions to be discussed later, periodically to all neighbors. Neighbors receiving these updates must process the entire update to find pertinent information and discard the rest. Some distance vector routing protocols like EIGRP do not send periodic routing table updates.

Routing Protocol Algorithms

Remember that an algorithm is a rule or process for arriving at a solution to a problem. In networking, algorithms are commonly used to determine the best route to forward traffic to a particular destination. The algorithm used by a particular routing protocol is responsible for building and maintaining the router’s routing table.

At the core of the distance vector protocol is the algorithm, which is used to calculate the best paths. Routers then send this information to neighboring routers.

An algorithm is a procedure for accomplishing a certain task, starting at a given initial state and terminating in a defined end state. Different routing protocols use different algorithms and processes to install routes in the routing table, send updates to neighbors, and make path determination decisions.

The algorithm used for the routing protocols defines the following processes:

- Mechanism for sending and receiving routing information
- Mechanism for calculating the best paths and installing routes in the routing table
- Mechanism for detecting and reacting to topology changes

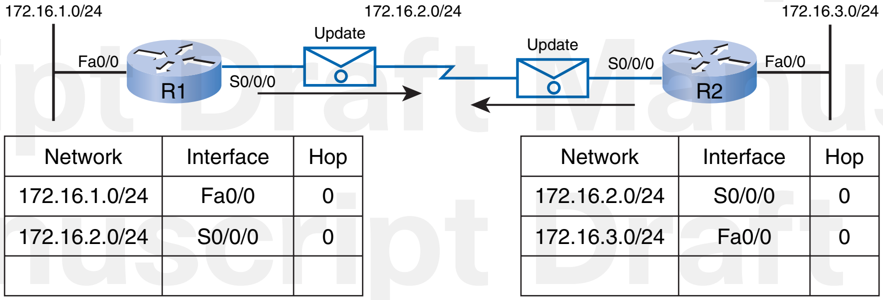
In Figure 4-5, R1 and R2 are configured with RIP. The algorithm sends and receives updates.

Both R1 and R2 then glean new information from the update. In this case, each router learns about a new network, as shown in Figure 4-6. The new networks are highlighted.

The algorithm on each router makes its calculations independently and updates the routing table with the new information.

Figure 4-7 illustrates what happens when there is a topology change. When the LAN on R2 goes down, the algorithm constructs a “triggered” update and sends it to R1. R1 then removes the network from the routing table. Triggered updates will be discussed later in this chapter.

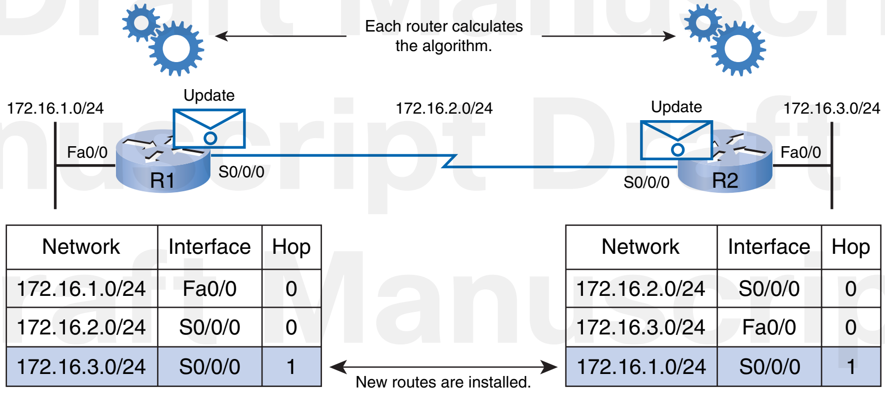
Figure 4-5 Sending and Receiving Updates



Network	Interface	Hop
172.16.1.0/24	Fa0/0	0
172.16.2.0/24	S0/0/0	0

Network	Interface	Hop
172.16.2.0/24	S0/0/0	0
172.16.3.0/24	Fa0/0	0

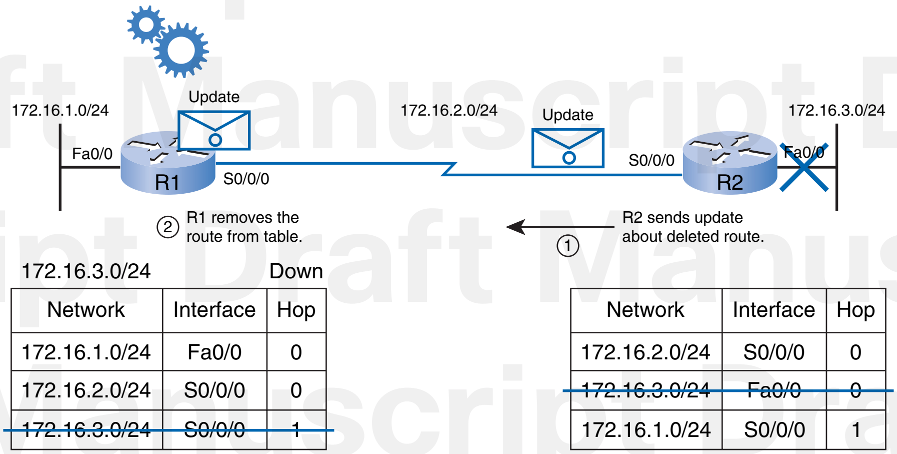
Figure 4-6 Calculating the Best Path and Installing Routes



Network	Interface	Hop
172.16.1.0/24	Fa0/0	0
172.16.2.0/24	S0/0/0	0
172.16.3.0/24	S0/0/0	1

Network	Interface	Hop
172.16.2.0/24	S0/0/0	0
172.16.3.0/24	Fa0/0	0
172.16.1.0/24	S0/0/0	1

Figure 4-7 Detecting and Reacting to Topology Changes



Network	Interface	Hop
172.16.1.0/24	Fa0/0	0
172.16.2.0/24	S0/0/0	0
~~172.16.3.0/24~~	~~S0/0/0~~	~~1~~

Network	Interface	Hop
172.16.2.0/24	S0/0/0	0
~~172.16.3.0/24~~	~~Fa0/0~~	~~0~~
172.16.1.0/24	S0/0/0	1

Routing Protocol Characteristics

There are several ways to differentiate routing protocols. The chart in Figure 4-1 shows some of the ways to characterize these routing protocols. Another way to compare routing protocols is by using other characteristics such as time to convergence and scalability.

Routing protocols can be compared based on the following characteristics:

- **Time to convergence:** Time to convergence defines how quickly the routers in the network topology share routing information and reach a state of consistent knowledge. The faster the convergence, the more preferable the protocol. Routing loops can occur when inconsistent routing tables are not updated because of slow convergence in a changing network.
- **Scalability:** Scalability defines how large a network can become based on the routing protocol that is deployed. The larger the network is, the more scalable the routing protocol needs to be.
- **Classless (use of VLSM) or classful:** Classless routing protocols include the subnet mask in the updates. This feature supports the use of variable-length subnet masking (VLSM) and better route summarization. Classful routing protocols do not include the subnet mask and cannot support VLSM.
- **Resource usage:** Resource usage includes the requirements of a routing protocol such as memory space, CPU utilization, and link bandwidth utilization. Higher resource requirements necessitate more powerful hardware to support the routing protocol operation in addition to the packet-forwarding processes.
- **Implementation and maintenance:** Implementation and maintenance describe the level of knowledge that is required for a network administrator to implement and maintain the network based on the routing protocol deployed.

Table 4-1 outlines the advantages and disadvantages of distance vector routing protocols.

Table 4-1 Advantages and Disadvantages of Distance Vector Routing Protocols

Advantages	Disadvantages
Simple implementation and maintenance. The level of knowledge required to deploy and later maintain a network with distance vector protocols is not high.	Slow convergence. The use of periodic updates can cause slower convergence. Even if some advanced techniques are used, like triggered updates which are discussed later, the overall convergence is still slower compared to link-state routing protocols.

Advantages	Disadvantages
Low resource requirements. Distance vector protocols typically do not need large amounts of memory to store the information, nor do they require a powerful CPU.	Limited scalability. Slow convergence can limit the size of the network because larger networks require more time to propagate routing information.
Depending on the network size and the IP addressing implemented, distance vector protocols typically do not require a high level of link bandwidth to send routing updates. However, this can become an issue if you deploy a distance vector protocol in a large network.	Routing loops. Routing loops can occur when inconsistent routing tables are not updated because of slow convergence in a changing network.

Comparing Routing Protocol Features

In Table 4-2, all the routing protocols discussed in the course are compared based on these characteristics. Although IGRP is no longer supported by Cisco IOS Software, it is shown here to compare it with EIGRP. Also, although the Intermediate System–to–Intermediate System (IS-IS) routing protocol is covered in the CCNP courses, it is shown here because it is a commonly used interior gateway protocol.

Table 4-2 Comparing Routing Protocol Features

	Distance Vector				Link-State	
	RIPv1	**RIPv2**	**IGRP**	**EIGRP**	**OSPF**	**IS-IS**
Speed of Convergence	Slow	Slow	Slow	Fast	Fast	Fast
Scalability—Size of Network	Small	Small	Small	Large	Large	Large
Use of VLSM	No	Yes	No	Yes	Yes	Yes
Resource Usage	Low	Low	Low	Medium	High	High
Implementation and Maintenance	Simple	Simple	Simple	Complex	Complex	Complex

Network Discovery

Network discovery is part of the process of the routing protocol algorithm that enables routers to first learn about remote networks.

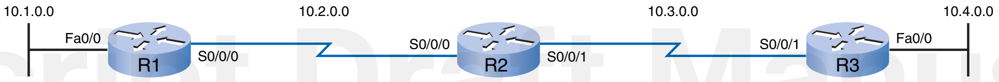
Cold Start

When a router cold-starts or powers up, it knows nothing about the network topology. It does not even know that there are devices on the other end of its links. The only information that a router has is from its own saved configuration file stored in NVRAM. After a router boots successfully, it applies the saved configuration. As described in Chapter 1, "Introduction to Routing and Packet Forwarding," and Chapter 2, "Static Routing," if the IP addressing is configured correctly and active, the router will initially discover its own directly connected networks.

After a cold start and before the exchange of routing information, the routers initially discover their own directly connected networks and subnet masks. As shown in Figure 4-8, this information is added to their routing tables:

- **R1:**
  - 10.1.0.0 available through interface FastEthernet 0/0
  - 10.2.0.0 available through interface Serial 0/0/0
- **R2:**
  - 10.2.0.0 available through interface Serial 0/0/0
  - 10.3.0.0 available through interface Serial 0/0/1
- **R3:**
  - 10.3.0.0 available through interface Serial 0/0/0
  - 10.4.0.0 available through interface FastEthernet 0/0

Figure 4-8 Network Discovery: Cold Start



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0

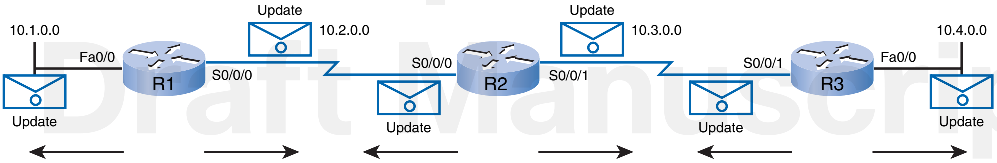
Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0

With this initial information, the routers start to exchange routing information.

Initial Exchange of Routing Information

If a routing protocol is configured, the routers begin exchanging routing updates, as shown in Figure 4-9. Initially, these updates include information only about their directly connected networks. Upon receiving an update, the router checks it for new information. Any routes that are not currently in its routing table are added.

Figure 4-9 Network Discovery: Initial Exchange of Routing Updates



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0

In Figure 4-9, Routers R1, R2, and R3 start their initial exchange. All three routers send their routing tables to their neighbors, which at this point only contain the directly connected networks.

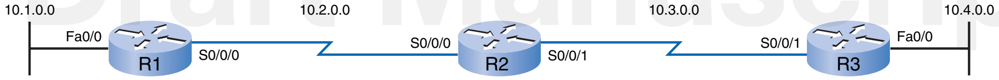
Each router processes updates in the following manner:

- **R1:**
  - Sends an update about network 10.1.0.0 out the Serial 0/0/0 interface with a metric of 1
  - Sends an update about network 10.2.0.0 out the FastEthernet 0/0 interface with a metric of 1
  - Receives an update from R2 about network 10.3.0.0 on Serial 0/0/0 with a metric of 1
  - Stores network 10.3.0.0 in the routing table with a metric of 1
- **R2:**
  - Sends an update about network 10.3.0.0 out the Serial 0/0/0 interface with a metric of 1
  - Sends an update about network 10.2.0.0 out the Serial 0/0/1 interface with a metric of 1

    - Receives an update from R1 about network 10.1.0.0 on Serial 0/0/0 with a metric of 1
    - Stores network 10.1.0.0 in the routing table with a metric of 1
    - Receives an update from R3 about network 10.4.0.0 on Serial 0/0/1 with a metric of 1
    - Stores network 10.4.0.0 in the routing table with a metric of 1
- **R3:**
    - Sends an update about network 10.4.0.0 out the Serial 0/0/1 interface with a metric of 1
    - Sends an update about network 10.4.0.0 out the FastEthernet 0/0 interface with a metric of 1
    - Receives an update from R2 about network 10.2.0.0 on Serial 0/0/1 with a metric of 1
    - Stores network 10.2.0.0 in the routing table with a metric of 1

As shown in Figure 4-10, after this first round of update exchanges, each router knows about the connected networks of their directly connected neighbors.

Figure 4-10 Network Discovery: Updated Tables After Initial Exchange



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1

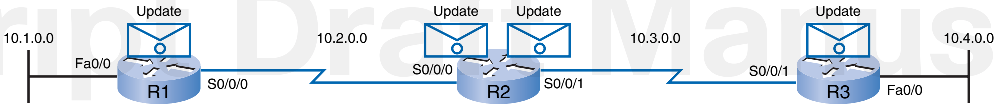
However, did you notice that R1 does not yet know about 10.4.0.0 and that R3 does not yet know about 10.1.0.0? Full knowledge and a converged network will not take place until there is another exchange of routing information.

Exchange of Routing Information

At this point, the routers have knowledge about their own directly connected networks and about the connected networks of their immediate neighbors. Continuing the journey toward convergence, the routers exchange the next round of periodic updates. Each router again checks the updates for new information.

In Figure 4-11, R1, R2, and R3 send their latest routing tables to their neighbors.

Figure 4-11 Network Discovery—Next Update



No New Information

Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/0	2

R1 and R3 now have complete routing tables.

Each router processes updates in the following manner:

- **R1:**
  - Sends an update about network 10.1.0.0 out the Serial 0/0/0 interface with a metric of 1.
  - Sends an update about networks 10.2.0.0 with a metric of 1 and 10.3.0.0 with a metric of 2 out the FastEthernet 0/0 interface.
  - Receives an update from R2 about network 10.4.0.0 on Serial 0/0/0 with a metric of 2.
  - Stores network 10.4.0.0 in the routing table with a metric of 2.
  - Same update from R2 contains information about network 10.3.0.0 on Serial 0/0/0 with a metric of 1. There is no change; therefore, the routing information remains the same.
- **R2:**
  - Sends an update about networks 10.3.0.0 with a metric of 1 and 10.4.0.0 with a metric of 2 out the Serial 0/0/0 interface.
  - Sends an update about networks 10.1.0.0 with a metric of 2 and 10.2.0.0 with a metric of 1 out the Serial 0/0/1 interface.
  - Receives an update from R1 about network 10.1.0.0 on Serial 0/0/0. There is no change; therefore, the routing information remains the same.
  - Receives an update from R3 about network 10.4.0.0 on Serial 0/0/1. There is no change; therefore, the routing information remains the same.

- **R3:**
  - Sends an update about network 10.4.0.0 out the Serial0/0/1 interface.
  - Sends an update about networks 10.2.0.0 with a metric of 2 and 10.3.0.0 with a metric of 1 out the FastEthernet 0/0 interface.
  - Receives an update from R2 about network 10.1.0.0 on Serial 0/0/1 with a metric of 2.
  - Stores network 10.1.0.0 in the routing table with a metric of 2.
  - Same update from R2 contains information about network 10.2.0.0 on Serial 0/0/1 with a metric of 1. There is no change; therefore, the routing information remains the same.

Note

Distance vector routing protocols typically implement a technique known as ***split horizon***. Split horizon prevents information from being sent out the same interface from which it was received. For example, R2 would not send an update out Serial 0/0/0 containing the network 10.1.0.0 because R2 learned about that network through Serial 0/0/0. This mechanism will be explained in more detail later in this chapter.

Convergence

The amount of time it takes for a network to converge is directly proportional to the size of that network. In Figure 4-12, a branch router in Region 4 (B2-R4) cold-starts and sends out an update with information about its four directly connected LANs.

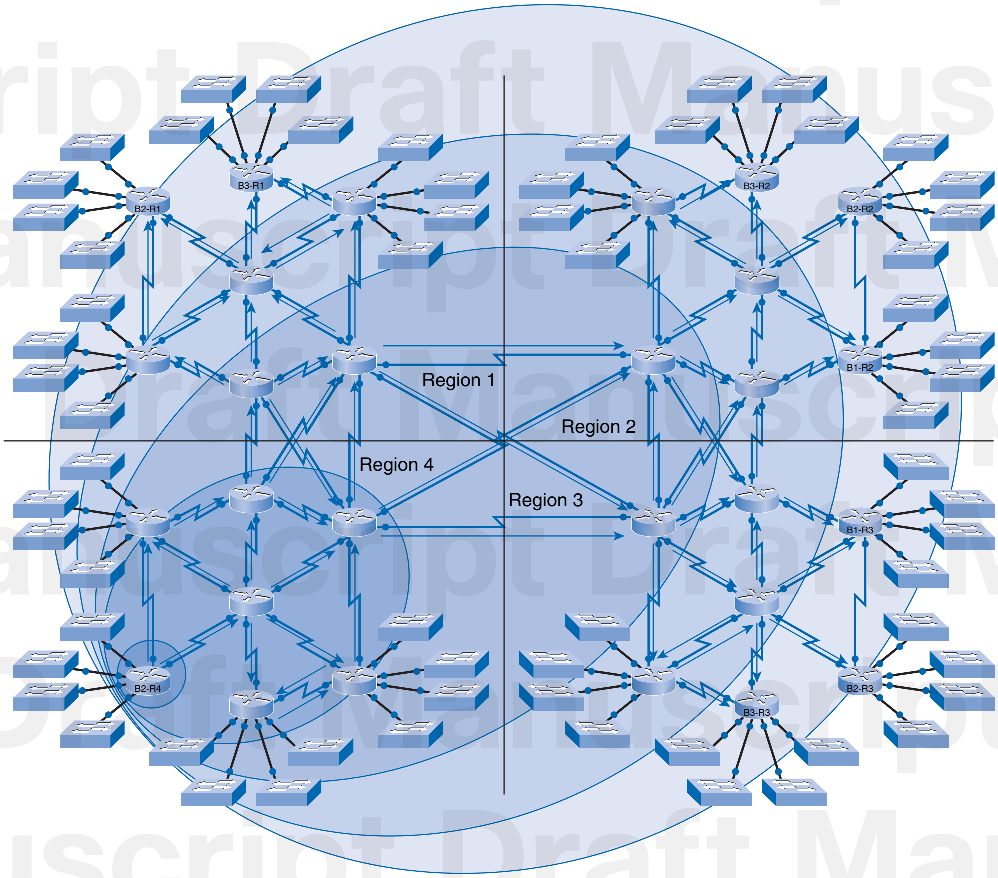
The shaded areas in the figure show the propagation of new routing information as updates are sent between neighboring routers. It takes five rounds of periodic update intervals before most of the branch routers in regions 1, 2, and 3 learn about the new routes advertised by B2-R4. Routing protocols are compared based on how fast they can propagate this information—their speed to convergence.

The speed of achieving convergence consists of

- How quickly the routers propagate a change in the topology in a routing update to their neighbors
- The speed of calculating best-path routes using the new routing information collected

A network is not completely operable until it has converged. Therefore, network administrators prefer routing protocols with shorter convergence times.

Figure 4-12 Convergence Time



Routing Table Maintenance

After the routers have initially learned about remote networks, routing protocols must maintain the routing tables so that they have the most current routing information. How the routing protocol maintains the routing table depends on the type of routing protocol (distance vector, link-state, or path vector) as well as the routing protocol itself (RIP, EIGRP, and so on).

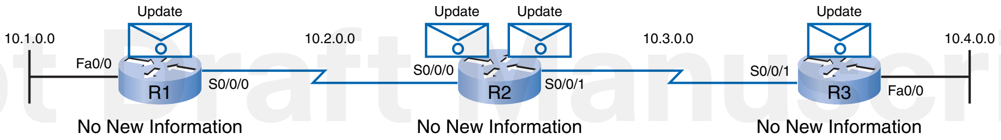
Periodic Updates

Many distance vector protocols employ periodic updates to exchange routing information with their neighbors and to maintain up-to-date routing information in the routing table. RIP and IGRP are examples of two such protocols.

Maintaining the Routing Table

In Figure 4-13, the routers are periodically sending the routing table to neighbors. Even though none of the routers have new information to share, periodic updates are sent anyway. The term *periodic updates* refers to the fact that a router sends the complete routing table to its neighbors at a predefined interval. For RIP, these updates are sent every 30 seconds as a broadcast (255.255.255.255), whether or not there has been a topology change. This 30-second interval is a route update timer that also aids in tracking the age of routing information in the routing table.

Figure 4-13 Periodic Updates



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Hop	0
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

The age of routing information in a routing table is refreshed each time an update is received. This way, information in the routing table can be maintained when there is a topology change. Changes might occur for several reasons, including

- Failure of a link
- Introduction of a new link
- Failure of a router
- Change of link parameters

RIP Timers

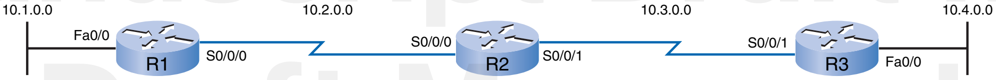
In addition to the update timer, IOS implements three additional timers for RIP:

- **Invalid:** If an update has not been received to refresh an existing route after 180 seconds (the default), the route is marked as invalid by setting the metric to 16. The route is retained in the routing table until the flush timer expires.
- **Flush:** By default, the flush timer is set for 240 seconds, which is 60 seconds longer than the invalid timer. When the flush timer expires, the route is removed from the routing table.

- **Hold-down:** This timer stabilizes routing information and helps prevent routing loops during periods when the topology is converging on new information. When a route is marked as unreachable, it must stay in holddown long enough for all routers in the topology to learn about the unreachable network. By default, the hold-down timer is set for 180 seconds. The hold-down timer is discussed in more detail later in this chapter.

Figure 4-14 shows the three-router topology we have been using to demonstrate routing protocol updates.

Figure 4-14 Three-Router Topology



Examples 4-1 and 4-2 show that the timer values can be verified with two commands: **show ip route** and **show ip protocols**.

Example 4-1 RIP Timers in the **show ip route** Command Output

```
R1# show ip route

<output omitted>

Gateway of last resort is not set

     10.0.0.0/16 is subnetted, 4 subnets
C       10.2.0.0 is directly connected, Serial0/0/0
R       10.3.0.0 [120/1] via 10.2.0.2, 00:00:04, Serial0/0/0
C       10.1.0.0 is directly connected, FastEthernet0/0
R       10.4.0.0 [120/2] via 10.2.0.2, 00:00:04, Serial0/0/0
```

Notice in the output from **show ip route** that each route learned through RIP shows the elapsed time since the last update, expressed in seconds.

Example 4-2 RIP Timers in the **show ip protocols** Command Output

```
R1# show ip protocols

Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 13 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  <output omitted>
```

```
Routing for Networks:
  10.0.0.0
Routing Information Sources:
  Gateway         Distance      Last Update
  10.3.0.1             120      00:00:27
Distance: (default is 120)
```

This information is also repeated in the **show ip protocols** command output under the heading Last Update. The **show ip protocols** command details when this router, R1, is due to send out its next round of updates. It also lists the invalid, hold-down, and flush timer default values.

Bounded Updates

Unlike other distance vector routing protocols, EIGRP does not send periodic updates. Instead, EIGRP sends ***bounded updates*** about a route when a path changes or the metric for that route changes. When a new route becomes available or when a route needs to be removed, EIGRP sends an update only about that network instead of the entire table. This information is sent only to those routers that need it.

EIGRP uses updates that are

- Nonperiodic, because they are not sent out on a regular basis
- Partial, because they are sent only when there is a change in topology that influences routing information
- Bounded, meaning that the propagation of partial updates is automatically bounded so that only those routers that need the information are updated

Note

Chapter 9, “EIGRP,” provides more detailed information on how EIGRP operates.

Triggered Updates

To speed the convergence when there is a topology change, RIP uses triggered updates. A ***triggered update*** is a routing table update that is sent immediately in response to a routing change. Triggered updates do not wait for update timers to expire. The detecting router immediately sends an update message to adjacent routers. The receiving routers, in turn, generate triggered updates that notify their neighbors of the change.

Triggered updates are sent when one of the following events occurs:

- An interface changes state (up or down)
- A route has entered (or exited) the unreachable state
- A route is installed in the routing table

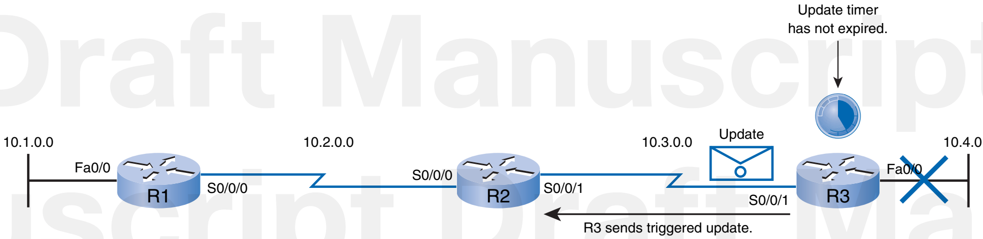
Using only triggered updates would be sufficient if there were a guarantee that the wave of updates would reach every appropriate router immediately. However, there are two problems with triggered updates:

- Packets containing the update message can be dropped.
- Packets containing the update message can be corrupted by some link in the network.

The triggered updates do not happen instantaneously. A router that has not yet received the triggered update could issue a regular update at just the wrong time, causing the bad route to be reinserted in a neighbor that had already received the triggered update.

Figure 4-15 shows how a network topology change is propagated through the network by sending a triggered update.

Figure 4-15 Triggered Updates



When network 10.4.0.0 becomes unavailable and R3 becomes aware of that, R3 sends out the information to its neighbors before the update timer expires. The information is then propagated through the network.

Random Jitter

When multiple routers transmit routing updates at the same time on multiaccess LAN segments, the update packets can collide and cause delays or consume too much bandwidth.

Note

Collisions are an issue only with hubs and not with switches.

Sending updates at the same time is known as the *synchronization* of updates. Synchronization can become a problem with distance vector routing protocols because of their usage of periodic updates. As more routers' timers become synchronized, more collisions of updates and more delays occur in the network. Initially, the updates of routers will not be synchronized. But over time, the timers across a network will become globally synchronized.

To prevent the synchronization of updates between routers, Cisco IOS uses a random variable, called RIP_JITTER, which adds a variable amount of time to the update interval for each router in the network. This random jitter, or variable amount of time, ranges from 0 to 15 percent of the specified update interval. In this way, the update interval varies randomly in a range from 25 to 30 seconds for the default 30-second interval.

Routing Loops

Routing loops can cause a severe impact on network performance. The following sections discuss the causes and solutions of routing loops with distance vector routing protocols.

Defining a Routing Loop

A routing loop is a condition in which a packet is continuously transmitted within a series of routers without ever reaching its intended destination network. A routing loop can occur when two or more routers have inaccurate routing information to a destination network.

The loop can be a result of

- Incorrectly configured static routes
- Incorrectly configured route redistribution (redistribution is a process of handing the routing information from one routing protocol to another routing protocol and is discussed in CCNP-level courses)
- Inconsistent routing tables not being updated because of slow convergence in a changing network

Distance vector routing protocols are simple in their operations. Their simplicity results in protocol drawbacks like routing loops. Routing loops are less of a problem with link-state routing protocols but can occur under certain circumstances.

Note

IP has its own mechanism to prevent the possibility of a packet traversing the network endlessly. IP has a Time to Live (TTL) field, and its value is decremented by 1 at each router. If the TTL is 0, the router drops the packet. The TTL is set by the operating system of the host that originated the packet. TTL values are typically much higher than the hop count limit of 15, with a maximum value of 255.

Implications of Routing Loops

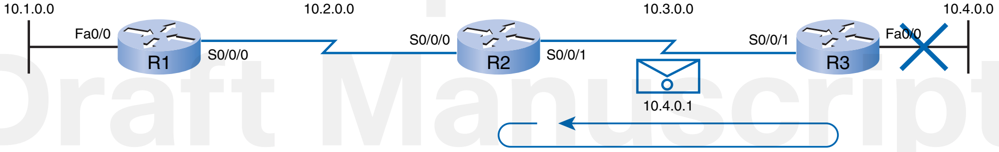
A routing loop can have a devastating effect on a network, resulting in degraded network performance or even network downtime.

A routing loop can create the following conditions:

- Link bandwidth will be used for traffic looping back and forth between the routers in a loop.
- A router's CPU will be burdened with useless packet forwarding that will negatively impact the convergence of the network.
- Routing updates might get lost or not be processed in a timely manner. These conditions would introduce additional routing loops, making the situation even worse.
- Packets might get lost in "black holes," never reaching their intended destinations.

Figure 4-16 shows a possible routing loop scenario in which mechanisms to prevent such loops do not exist.

Figure 4-16 Routing Loop



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	S0/0/1	2
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Bad route entries.

In this scenario, R2 sent R3 a route to 10.4.0.0 *before* R3 could inform R2 that the network is down. R3—not knowing the R2 doesn't have a route to 10.4.0.0—installs the new route for 10.4.0.0, pointing to R2 as the vector with a distance of 2. R2 and R3 now believe that the other router is the next hop for traffic to 10.4.0.0. The result of these bad routes is that traffic to destinations of the 10.4.0.0 network will loop between R2 and R3 until one of the routers drops the packet (the TTL expires).

As you can see, routing loops consume bandwidth and also router resources, resulting in a slow or even unresponsive network.

There are a number of mechanisms available to eliminate routing loops, primarily with distance vector routing protocols. These mechanisms include

- Defining a maximum metric to prevent count to infinity
- Hold-down timers
- Split horizon
- Route poisoning or poison reverse
- Triggered updates

Triggered updates were discussed in the previous section. The other loop-avoidance mechanisms are discussed later in this chapter.

Packet Tracer
□ Activity

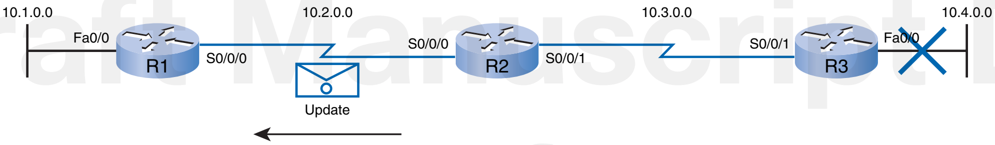
Routing Loops (4.4.1)

Use the Packet Tracer Activity to experience how a routing loop might occur with misconfigured static routes. Use file e2-441.pka on the CD-ROM that accompanies this book to perform this activity using Packet Tracer.

Count-to-Infinity Condition

Count to infinity is a condition that exists when inaccurate routing updates increase the metric value to "infinity" for a network that is no longer reachable. Figure 4-17 shows what happens to the routing tables when all three routers continue to send inaccurate updates about the downed 10.4.0.0 network to each other. The routers will continue to increment the metric until infinity for that protocol is reached. Each protocol defines infinity at a different value.

Figure 4-17 Count to Infinity



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	6

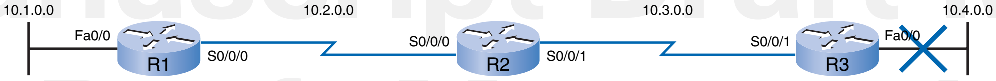
Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	7

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	S0/0/1	6
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Preventing Routing Loops by Setting a Maximum Metric Value

To eventually stop the incrementing of the metric, "infinity" is defined by setting a maximum metric value. For example, in Figure 4-18, RIP defines infinity as 16 hops—an "unreachable" metric. When the routers "count to infinity," they mark the route as unreachable.

Figure 4-18 10.4.0.0 Is Unreachable—Hop Count Is 16



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	16

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	16

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	S0/0/1	16
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Preventing Routing Loops with Hold-Down Timers

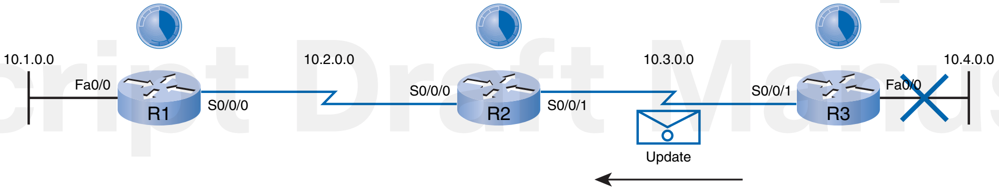
Earlier you learned that distance vector protocols employ triggered updates to speed the convergence process. Remember that in addition to triggered updates, routers using distance vector routing protocols also send periodic updates. Imagine that a particular network is unstable. The interface resets as up, then down, and then up again in rapid succession. The route is flapping. Using triggered updates, the routers might react too quickly and unknowingly create a routing loop. A routing loop could also be created by a periodic update that is sent by the routers during the instability. Hold-down timers prevent routing loops from being created by these conditions. Hold-down timers also help prevent the count-to-infinity condition.

Hold-down timers are used to prevent regular update messages from inappropriately reinstating a route that might have gone bad. Hold-down timers instruct routers to hold any changes that might affect routes for a specified period of time. If a route is identified as down or possibly down, any other information for that route containing the same status, or worse, is ignored for a predetermined amount of time (the hold-down period). This means that routers will leave a route marked as unreachable in that state for a period of time that is long enough for updates to propagate the routing tables with the most current information.

Figures 4-19 through 4-23, along with the following discussion of steps, illustrate how hold-down timers work:

1. Network 10.4.0.0 attached to R3 goes down. R3 sends a triggered update (see Figure 4-19).

Figure 4-19 Triggered Update Sent to R2



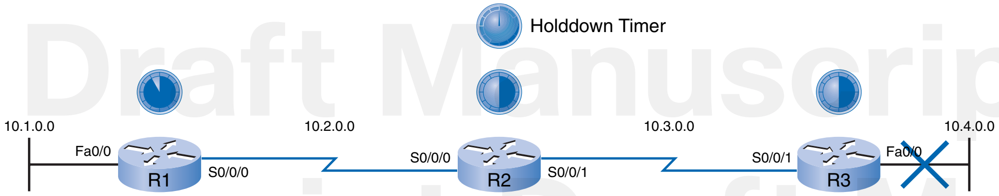
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
~~10.4.0.0~~	~~S0/0/1~~	~~0~~
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

2. R2 receives the update from R3 indicating that network 10.4.0.0 is now no longer accessible. R3 marks the network as possibly down and starts the hold-down timer (see Figure 4-20).

Figure 4-20 R2 Places 10.4.0.0 in Holddown



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

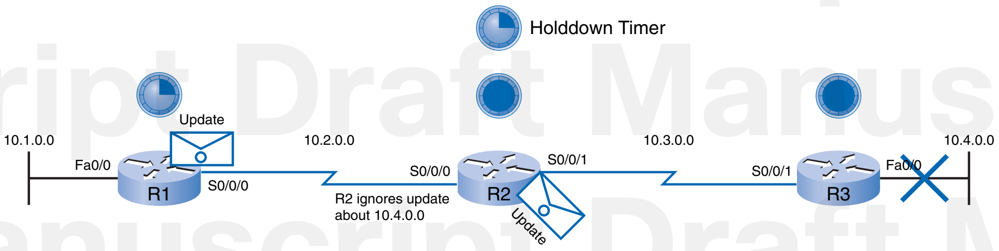
Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
~~10.4.0.0~~	~~S0/0/1~~	~~0~~
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

3. If an update with a better metric for that network is received from any neighboring router during the hold-down period, R2 will reinstate the network and the hold-down timer is removed.

4. If an update from any other neighbor is received during the hold-down period with the same or worse metric for that network, that update is ignored (see Figure 4-21). Thus, more time is allowed for the information about the change to be propagated.

Figure 4-21 R2 Ignores Update from R1



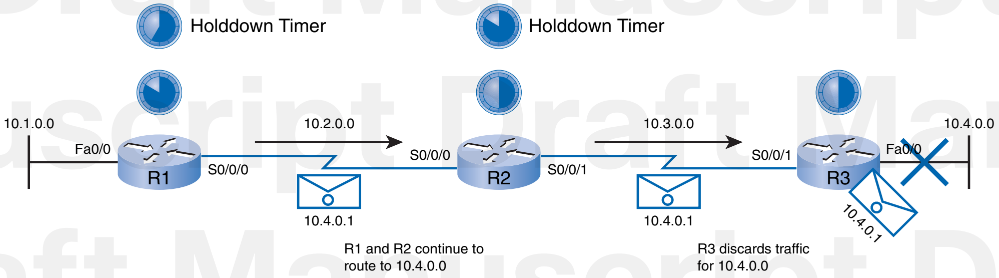
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
~~10.4.0.0~~	~~S0/0/1~~	~~0~~
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

5. R1 and R2 still forward packets to 10.4.0.0, even though it is marked as possibly down (see Figure 4-22). This allows the router to overcome any issues associated with intermittent connectivity. If the destination network is truly unavailable and the packets are forwarded, black-hole routing is created and lasts until the hold-down timer expires.

Figure 4-22 Traffic to 10.4.0.0 Is Still Routed



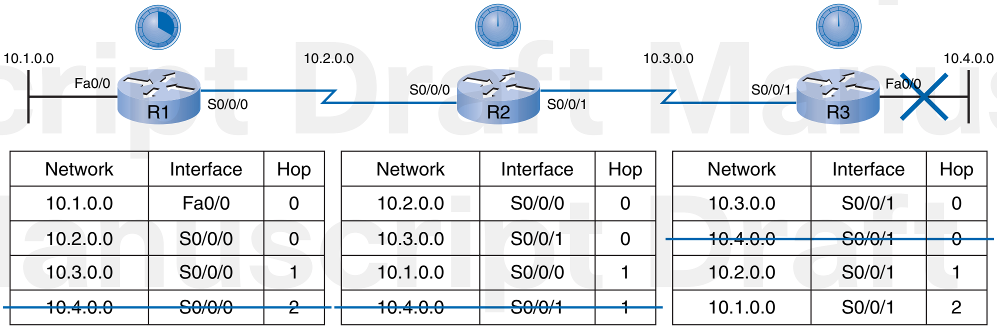
Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
~~10.4.0.0~~	~~S0/0/1~~	~~0~~
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

6. When the hold-down timers expire on R1 and R2, 10.4.0.0 is removed from the routing table. No traffic to 10.4.0.0 will be routed (see Figure 4-23).

Figure 4-23 Network Is Now Converged



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
~~10.4.0.0~~	~~S0/0/0~~	~~2~~

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
~~10.4.0.0~~	~~S0/0/1~~	~~1~~

Network	Interface	Hop
10.3.0.0	S0/0/1	0
~~10.4.0.0~~	~~S0/0/1~~	~~0~~
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Preventing Routing Loops with the Split Horizon Rule

Another method used to prevent routing loops caused by slow convergence of a distance vector routing protocol is split horizon. The split horizon rule says that a router should not advertise a network through the interface from which the update came.

Applying split horizon to the previous example of route 10.4.0.0 produces the following actions:

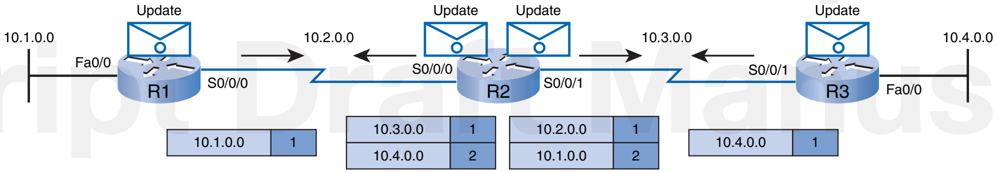
1. R3 advertises the 10.4.0.0 network to R2.
2. R2 receives the information and updates its routing table.
3. R2 then advertises the 10.4.0.0 network to R1 out S0/0/0. R2 does not advertise 10.4.0.0 to R3 out S0/0/1, because the route originated from that interface.
4. R1 receives the information and updates its routing table.
5. Because of split horizon, R1 also does not advertise the information about network 10.4.0.0 back to R2.

Complete routing updates are exchanged, with the exception of routes that violate the split horizon rule. The results look like this:

- R2 advertises networks 10.3.0.0 and 10.4.0.0 to R1.
- R2 advertises networks 10.1.0.0 and 10.2.0.0 to R3.
- R1 advertises network 10.1.0.0 to R2.
- R3 advertises network 10.4.0.0 to R2.

Figure 4-24 illustrates this example of the split horizon rule. Notice that R2 sends different routing updates to R1 and R3. Also notice that each router increments the hop count *before* sending the update.

Figure 4-24 Split Horizon Rule



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Note

Split horizon can be disabled by an administrator. Under certain conditions, this has to be done to achieve the proper routing. These conditions are discussed in later courses.

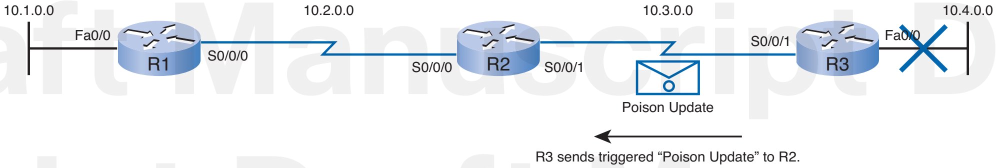
Split horizon can be combined with route poisoning or poison reverse to specifically mark a route as unreachable, as described in the sections that follow.

Route Poisoning

Route poisoning is used to mark the route as unreachable in a routing update that is sent to other routers. Unreachable is interpreted as a metric that is set to the maximum. For RIP, a poisoned route has a metric of 16.

Figure 4-25 shows route poisoning in effect.

Figure 4-25 Route Poisoning



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	16
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

The following process occurs:

1. Network 10.4.0.0 becomes unavailable because of a link failure.
2. R3 poisons the metric with a value of 16 and then sends out a triggered update stating that 10.4.0.0 is unavailable.
3. R2 processes that update. Because the metric is 16, R2 invalidates the routing entry in its routing table.
4. R2 then sends the poison update to R1, indicating that route is unavailable, again by setting the metric value to 16.
5. R1 processes the update and invalidates the routing entry for 10.4.0.0 in its routing table.

Route poisoning speeds the convergence process because the information about 10.4.0.0 spreads through the network more quickly than waiting for the hop count to reach "infinity."

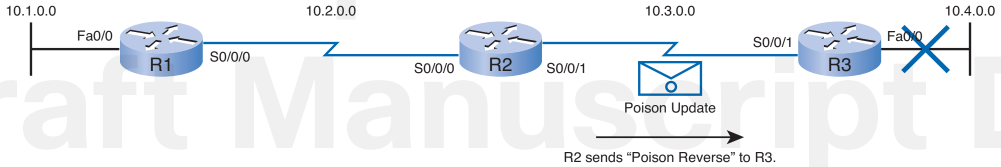
Split Horizon with Poison Reverse

Poison reverse can be combined with the split horizon technique. The method is called split horizon with poison reverse. The rule for split horizon with poison reverse states that when sending updates out a specific interface, designate any networks that were learned on that interface as unreachable.

The concept of split horizon with poison reverse is that explicitly telling a router to ignore a route is better than not telling it about the route in the first place.

Figure 4-26 shows an example of split horizon with poison reverse in effect.

Figure 4-26 Poison Reverse



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	16

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	16
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

The following process occurs:

1. Network 10.4.0.0 becomes unavailable because of a link failure.

2. R3 poisons the metric with a value of 16 and then sends out a triggered update stating that 10.4.0.0 is unavailable.
3. R2 processes that update, invalidates the routing entry in its routing table, and immediately sends a poison reverse back to R3.

Poison reverse is a specific circumstance that overrides split horizon. It occurs to ensure that R3 is not susceptible to incorrect updates about network 10.4.0.0.

Note

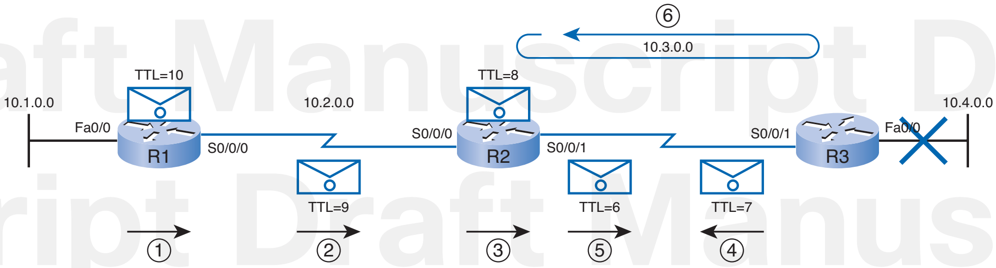
Split horizon is enabled by default. However, split horizon with poison reverse might not be the default on all IOS implementations.

Preventing Routing Loops with IP and TTL

The Time to Live (TTL) is an 8-bit field in the IP header that limits the number of hops a packet can traverse through the network before it is discarded. The purpose of the TTL field is to avoid a situation in which an undeliverable packet keeps circulating on the network endlessly. With TTL, the 8-bit field is set with a value by the source device of the packet. The TTL is decreased by 1 by every router on the route to its destination. If the TTL field reaches 0 before the packet arrives at its destination, the packet is discarded and the router sends an Internet Control Message Protocol (ICMP) error message back to the source of the IP packet.

Figure 4-27 shows a situation where the routing tables do not have accurate information about the downed 10.4.0.0 network. Even in the case of this routing loop, packets will not loop endlessly in the network. Eventually the TTL value will be decreased to 0 and the packet will be discarded by the router.

Figure 4-27 TTL in Effect



Network	Interface	Hop
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	4

Network	Interface	Hop
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.4.0.0	S0/0/0	1
10.4.0.0	S0/0/1	3

Network	Interface	Hop
10.3.0.0	S0/0/1	0
10.4.0.0	S0/0/1	2
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

The sequence of events, as depicted in Figure 4-27, is as follows:

1. R1 receives a packet with a TTL value of 10.
2. R1 decrements the TTL value to 9 and sends the packet to R2.
3. R2 decrements the TTL value to 8 and sends the packet to R3.
4. R3 decrements the TTL value to 7 and sends the packet back to R2.
5. R2 decrements the TTL value to 6 and sends the packet back to R3.
6. The packet loops between R2 and R3 until the TTL value reaches 0. Then the packet is discarded.

Distance Vector Routing Protocols Today

Later in this book, you will learn about link-state routing protocols. Although link-state routing protocols have several advantages over distance vector routing protocols, distance vector routing protocols are still in use today. In Chapter 9, you will learn that EIGRP is an "enhanced" distance vector routing protocol. These enhancements make EIGRP a viable choice for a routing protocol in many environments.

RIP and EIGRP

For distance vector routing protocols, there really are only two choices: RIP or EIGRP. The decision about which routing protocol to use in a given situation is influenced by a number of factors, including

- Size of the network
- Compatibility between models of routers
- Administrative knowledge required

Table 4-3 compares distance vector routing protocol features.

Table 4-3 Comparing Distance Vector Routing Protocol Features

	RIPv1	RIPv2	IGRP	EIGRP
Speed of Convergence	Slow	Slow	Slow	Fast
Scalability—Size of Network	Small	Small	Small	Large
Use of VLSM	No	Yes	No	Yes
Resource Usage	Low	Low	Low	Medium
Implementation and Maintenance	Simple	Simple	Simple	Complex

RIP

Over the years, RIP has evolved from a classful routing protocol (RIPv1) to a classless routing protocol (RIPv2). RIPv2 is a standardized routing protocol that works in a mixed-vendor router environment. Routers made by different companies can communicate using RIP. It is one of the easiest routing protocols to configure, making it a good choice for small networks. However, RIPv2 still has limitations. Both RIPv1 and RIPv2 have a route metric that is based only on hop count and that is limited to 15 hops.

Features of RIP include

- Supports split horizon and split horizon with poison reverse to prevents loops.
- Is capable of load-balancing up to six equal-cost paths. The default is four equal-cost paths.

RIPv2 introduced the following improvements to RIPv1:

- Includes the subnet mask in the routing updates, making it a classless routing protocol
- Has an authentication mechanism to secure routing table updates
- Supports variable-length subnet mask (VLSM)
- Uses multicast addresses instead of broadcast
- Supports manual route summarization

EIGRP

EIGRP was developed from IGRP, another distance vector protocol. EIGRP is a classless, distance vector routing protocol with features found in link-state routing protocols. However, unlike RIP or OSPF, EIGRP is a proprietary protocol developed by Cisco and runs only on Cisco routers.

EIGRP features include

- Triggered updates (EIGRP has no periodic updates).
- Use of a ***topology table*** to maintain all the routes received from neighbors (not only the best paths).
- Establishment of adjacencies with neighboring routers using the EIGRP Hello protocol.
- Support for VLSM and manual route summarization. These allow EIGRP to create hierarchically structured large networks.

Advantages of EIGRP are as follows:

- Although routes are propagated in a distance vector manner, the metric is based on minimum bandwidth and cumulative delay of the path, rather than hop count.

- Fast convergence because of Diffusing Update Algorithm (DUAL) route calculation. DUAL allows the insertion of backup routes into the EIGRP topology table, which are used in case the primary route fails. Because it is a local procedure, the switchover to the backup route is immediate and does not involve the action in any other routers.
- Bounded updates mean that EIGRP uses less bandwidth, especially in large networks with many routes.
- EIGRP supports multiple network layer protocols through Protocol Dependent Modules, which include support for IP, ***IPX***, and AppleTalk.

Summary

One way of classifying routing protocols is by the type of algorithm they use to determine the best path to a destination network. Routing protocols can be classified as distance vector, link-state, or path vector. Distance vector means that routes are advertised as vectors of distance and direction. Distance is defined in terms of a metric, such as hop count, and direction is simply the next-hop router or exit interface.

Distance vector routing protocols include

- RIPv1
- RIPv2
- IGRP
- EIGRP

Routers that use distance vector routing protocols determine the best path to remote networks based on the information they learn from their neighbors. If Router X learns of two paths to the same network, one through Router Y at seven hops and another through Router Z at ten hops, the router will choose the shorter path using Router Y as the next-hop router. Router X has no knowledge of what the network looks like beyond Routers Y and Z, and can only make its best-path decision based on the information sent to it by these two routers. Distance vector routing protocols do not have a map of the topology as do link-state routing protocols.

Network discovery is an important process of any routing protocol. Some distance vector routing protocols such as RIP go through a step-by-step process of learning and sharing routing information with their neighbors. As routes are learned from one neighbor, that information is passed on to other neighbors with an increase in the routing metric.

Routing protocols also need to maintain their routing tables to keep them current and accurate. RIP exchanges routing table information with its neighbors every 30 seconds. EIGRP, another distance vector routing protocol, does not send these periodic updates and only sends a “bounded” update when there is a change in the topology and only to those routers that need that information. EIGRP is discussed in a later chapter.

RIP also uses timers to determine when a neighboring router is no longer available, or when some of the routers might not have current routing information. This is typically because the network has not yet converged because of a recent change in the topology. Distance vector routing protocols also use triggered updates to help speed convergence time.

One disadvantage of distance vector routing protocols is the potential for routing loops. Routing loops can occur when the network is in an unconverged state. Distance vector routing protocols use hold-down timers to prevent the router from using another route to a recently down network until all the routers have had enough time to learn about this change in the topology.

Split horizon and split horizon with poison reverse are also used by routers to help prevent routing loops. The split horizon rule states that a router should never advertise a route through the interface from which it learned that route. Split horizon with poison reverse means that it is better to explicitly state that this router does not have a route to this network by poisoning the route with a metric stating that the route is unreachable.

Distance vector routing protocols are sometimes referred to as "routing by rumor," although this can be somewhat of a misnomer. Distance vector routing protocols are popular with many network administrators because they are typically easily understood and simple to implement. This does not necessarily mean that link-state routing protocols are any more complicated or difficult to configure.

Unfortunately, link-state routing protocols have received this somewhat unwarranted reputation. You will learn in later chapters that link-state routing protocols are as easy to understand and configure as distance vector routing protocols.

Activities and Labs

The activities and labs available in the companion Routing Protocols and Concepts, CCNA Exploration Labs and Study Guide (ISBN 1-58713-204-4) provide hands-on practice with the following topics introduced in this chapter:

Lab 4-1: Routing Table Interpretation Lab (4.6.1)

In this lab activity, you re-create a network based only on the output from the **show ip route** command. Then, to verify your answer, you configure the routers and compare the actual routing table to the routing table shown in the lab documentation.

Packet Tracer
□ Companion

Many of the hands-on labs include Packet Tracer Companion Activities, where you can use Packet Tracer to complete a simulation of the lab. Look for this icon in *Routing Protocols and Concepts, CCNA Exploration Labs and Study Guide* (ISBN 1-58713-204-4) for hands-on labs that have a Packet Tracer Companion.

Check Your Understanding

Complete all the review questions listed here to test your understanding of the topics and concepts in this chapter. Answers are listed in the appendix, "Check Your Understanding and Challenge Questions Answer Key."

1. Which four statements are true regarding some distance vector routing protocols?

   A. Hop counts can be used for path selection.

   B. They scale well.

   C. Routing updates are broadcast at intervals.

   D. EIGRP can do unequal-cost load balancing.

   E. RIPv1 multicasts its routing updates.

   F. RIP sends its entire routing table to directly connected neighbors (except for any routes affected by split horizon).

2. Which conditions cause some distance vector routing protocols to send routing table updates? (Choose three.)

   A. When the hold-down timer expires

   B. When a change occurs in the network topology

   C. When the update timer value expires

   D. When a triggered update is received from another router

   E. When a packet is received that is destined for an unknown network

   F. When there have been no routing table changes for 30 minutes

3. What are two characteristics of EIGRP updates?

   A. Include all EIGRP routes

   B. Include the full routing table

   C. Independent of architecture

   D. Only triggered for route topology changes

   E. Broadcast to affected neighbors

   F. Bounded only to those routers that need the update

4. What feature was added to RIP to help with synchronization errors?

   A. Hold-down timer

   B. RIP_JITTER

   C. RIP_DELAY

   D. Jitter control

5. Which two of the following are timers used for RIP?

   A. Invalid

   B. Refresh

   C. Flush

   D. Deadlink

   E. Hello

6. Which statement is true concerning the advantages of a distance vector protocol?

   A. Periodic updates speed convergence.

   B. Convergence times make routing loops impossible.

   C. Ease of implementation makes configuration simple.

   D. They work well in complex networks.

   E. Their convergence times are faster than link-state routing protocols.

7. Which mechanism can be used to avoid a count-to-infinity loop?

   A. Split horizon

   B. Route poisoning

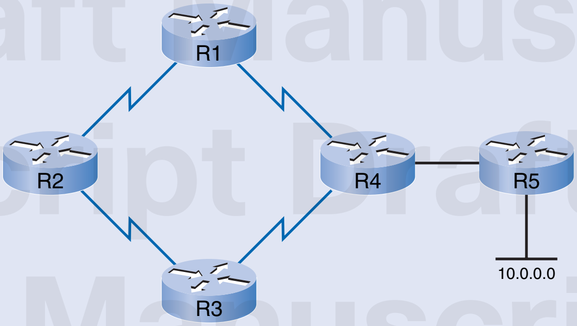
   C. Hold-down timers

   D. Triggered updates

   E. Split horizon with poison reverse

8. Refer to Figure 4-28. The network shown is running the RIP routing protocol. What mechanism will keep Router 4 from sending updates about the 10.0.0.0 network back to Router 5?

Figure 4-28 Check Your Understanding, Question #8



   A. Split horizon

   B. Poison reverse

   C. Route poisoning

   D. Hold-down timers

   E. Maximum hop count

9. What allows RIP to avoid routing loops by advertising a metric of infinity?

   A. Split horizon

   B. Route poisoning

   C. Hold-down timers

   D. Maximum hop count

   E. Time to Live (TTL) field of the IP header

10. Which field in the IP header ensures that packets will not loop endlessly on a network?

   A. CRC

   B. TOS

   C. TTL

   D. Checksome

11. Match the loop-preventing mechanism with its corresponding function.

   Loop-prevention mechanism:

   Split horizon:

   Route Poisoning:

   Hold-down timers:

   Triggered updates:

   Function:

   A. Routes learned through an interface are not advertised out that same interface.

   B. Routes learned through an interface are advertised back out the same interface as unreachable.

   C. Topology changes are immediately sent to adjacent routers.

   D. Allows time for topology changes to travel through an entire network.

Challenge Questions and Activities

These questions and activities require a deeper application of the concepts covered in this chapter. You can find the answers at the end of the chapter.

1. Briefly explain the basic operation of RIP and IGRP.

2. Explain convergence and why it is important.

3. What are the four main timers used by RIP? How many seconds are in each timer? What is the purpose of each timer?

4. What five techniques do distance vector routing protocols use to prevent routing loops?

To Learn More

Understanding the distance vector algorithm is not difficult. There are many book and online sources that show how algorithms such as the Bellman-Ford algorithm are used in networking. There are several websites devoted to explaining how these algorithms work. Seek out some of the resources and familiarize yourself with how this algorithm works.

Here are some suggested resources:

- *Interconnections, Bridges, Routers, Switches, and Internetworking Protocols*, by Radia Perlman
- *Cisco IP Routing*, by Alex Zinin
- Routing the Internet, by Christian Huitema

Check Your Understanding and Challenge Questions Answer Key

Check Your Understanding

1. A, C, D, F. Because of slow convergence, distance vector routing protocols do not scale well. RIPv2 does multicast its updates; however, RIPv1 uses broadcasts in its updates.

2. B, C, D. Most distance vector routing protocols will send a triggered update when they sense a change in the topology, such as a new link becoming active. When a triggered update is received by a router, it will immediately forward that update to other routers. Some distance vector routing protocols, such as RIP and IGRP, send periodic updates. An update timer is used to determine the interval of these routing updates. A hold-down timer expiring will not cause any new updates to be sent. The hold-down timer is used to determine how long to keep a route in the hold-down state.

3. D, F. EIGRP does not send periodic updates. EIGRP updates are only sent when there is a topology change and is only sent to those routers that need the updated information.

4. B. Cisco IOS uses the random variable RIP-JITTER, which varies the 30-second update interval from 25 to 30 seconds.

5. A, C. RIP uses several timers, including the invalid, flush, route update, and hold-down timers.

6. C. Distance vector routing protocols have the reputation of being easier to configure. Although this is true, link-state routing protocols are only slightly more difficult to configure. The ease of implementation should not usually be the basis for deciding which routing protocol to use.

7. C. Instead of propagating potentially incorrect information, the hold-down timer will cause the route to be marked as unreachable for a period of time, giving the network time to converge.

8. A. Using the split horizon rule, R4 will not send R5 an update regarding the 10.0.0.0 network because R4 received that update from R5.

9. B. Route poisoning is used to mark a route as unreachable. RIP marks a route as unreachable by advertising a metric of "infinity," or 16.

10. C. The value of the TTL (Time to Live) is set by the source. As each router receives the packet, the TTL is decreased by 1. If the TTL reaches 0, the router drops that packet.

11. Answer:

    Split horizon: A

    Route Poisoning: B

    Hold-down timers: D

    Triggered updates: C

Challenge Questions and Activities

1. RIP and IGRP are distance vector routing protocols characterized by periodic updates that are broadcast to directly connected neighbors. The entire routing table is sent in the update.

2. Convergence occurs when all routers in the network have consistent and correct information about how to reach destination networks. A network is not completely operable until it has converged; therefore, routing protocols require short convergence times.

3. Answer:

    Update timer: (30 seconds) Used to time when to send the next update

    Invalid timer: (180 seconds) Counts how long it has been since the last update for a route

    Hold-down timer: (180 seconds) The amount of time an unreachable route is in hold-down

    Flush timer: (240 seconds) Time until a route is removed from the routing table

4. Answer:

    Defining maximum metric to prevent count to infinity
    Hold-down timers
    Split horizon
    Route poisoning or poison reverse
    Triggered updates