Exam Topics in This Chapter

27 Add the RIP routing protocol to your configuration.

52 Add the IGRP routing protocol to your configuration.

52 List problems that each routing type encounters when dealing with topology changes, and describe techniques to reduce the number of these problems.

CHAPTER 7

Routing and Routing Protocols

The United States Postal Service routes a huge number of letters and packages each day. Most of us have seen pictures of huge sorting machines—even more so after the recent, horrible Anthrax attacks through the mail in the U.S. The sorting machine can run fast, sorting lots of letters—and then the letters are placed in the correct container, and onto the correct truck or plane, to reach the final destination. However, if no one programs the letter sorter to know where letters to each zip code should be sent, the sorter can't do its job. Similarly, Cisco routers can route many packets, but if the router doesn't know any routes, it can't do its job.

This chapter deals with the concepts and configuration required to fill a router's routing table. Cisco expects CCNAs to demonstrate a comfortable understanding of the logic behind the routing of packets, and the different but related logic behind routing protocols—the protocols used to discover routes. To fully appreciate the nuances of routing protocols, you need a thorough understanding of routing—the process of forwarding packets. You might want to review the section on Layer 3 in Chapter 3, "OSI Reference Model and Layered Communication," before proceeding with this chapter.

The CCNA exam covers the details of distance vector logic, which is covered in the first section of this chapter. This is the logic used by the Routing Information Protocol (RIP) and Interior Gateway Routing Protocol (IGRP), as well as IPX RIP. Along the way, alternative routing protocol algorithms (link-state and Diffusing Update Algorithm [DUAL]) are mentioned briefly.

Implementation details of RIP (Version 1 and Version 2) and IGRP are covered after that. Because EIGRP configuration is similar to IGRP, it is also covered briefly. As you'll find on the CCNA exam, knowledge and skills for routing protocol configuration and troubleshooting are topics required of CCNAs.

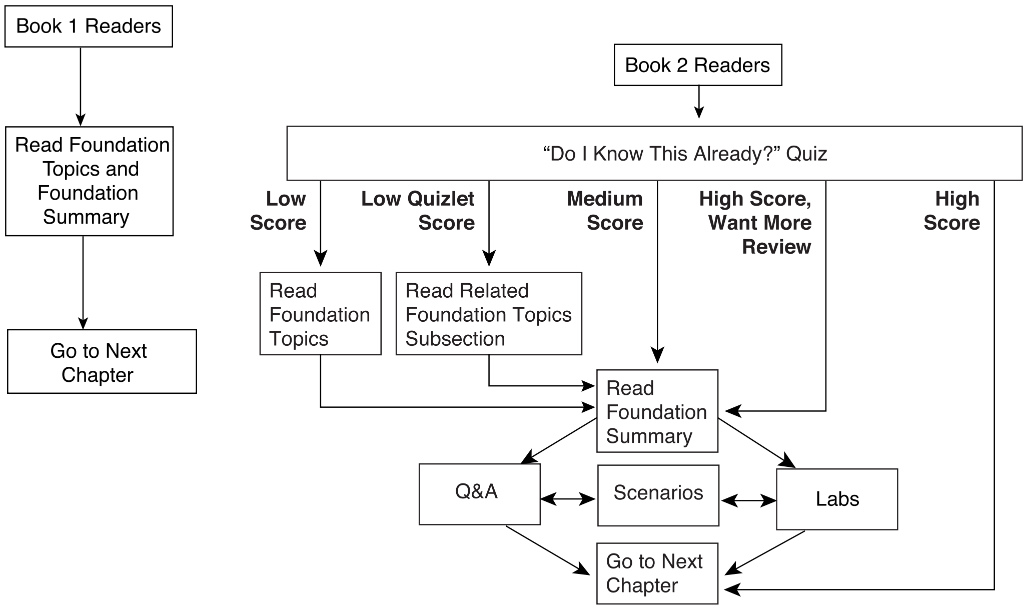
How to Best Use This Chapter

By following these steps, you can make better use of your study time:

- Keep your notes and the answers for all your work with this book in one place, for easy reference.

- Take the "Do I Know This Already?" quiz, and write down your answers. Studies show that retention is significantly increased through writing down facts and concepts, even if you never look at the information again.
- Use Figure 7-1 to guide you to the next step.

Figure 7-1 *How to Use This Chapter*



"Do I Know This Already?" Quiz

The "Do I Know This Already?" quiz helps you decide what parts of this chapter to use. If you already intend to read the entire chapter, you do not necessarily need to answer these questions now.

This 12-question quiz helps you determine how to spend your limited study time. The quiz is sectioned into two smaller six-question "quizlets" that help you select the sections of the chapter on which to focus. Figure 7-1 outlines suggestions on how to spend your time in this chapter based on your quiz score. Use Table 7-1 to record your scores.

Table 7-1 *Scoresheet for Quiz and Quizlets*

Quizlet Number	Foundation Topics Section Covering These Questions	Questions	Score
1	Distance Vector Routing Protocols	1 to 6	
2	Configuring RIP and IGRP	7 to 12	
All questions		1 to 12	

1 Define what split horizon means to the contents of a routing update. Does this apply to both the distance vector algorithm and the link-state algorithm?

____

____

____

2 Describe the purpose and meaning of route poisoning.

____

____

____

3 What term describes the underlying logic behind the OSPF routing protocol?

____

____

____

4 Describe the meaning and purpose of triggered updates.

____

____

____

5 List the interior IP routing protocols that have autosummarization enabled by default. Which of these protocols allow autosummarization to be disabled using a configuration command?

____

____

____

6 True or false: Distance vector routing protocols learn routes by transmitting routing updates.

____

____

____

7 Write down the steps you would take to migrate from RIP to IGRP in a router whose current RIP configuration includes only **router rip** followed by a **network 10.0.0.0** command.

____

____

____

8 How does the Cisco IOS Software designate a subnet in the routing table as a directly connected network? What about a route learned with IGRP or RIP?

____

____

____

9 From a router's user mode, without using debugs or privileged mode, how can you determine what routers are sending you routing updates?

____

____

____

10 If the command **router rip** followed by **network 10.0.0.0**, with no other **network** commands, is configured in a router that has an Ethernet0 interface with IP address 168.10.1.1, does RIP send updates out Ethernet0?

____

____

____

11 If the commands **router igrp 1** and **network 10.0.0.0** are configured in a router that has an Ethernet0 interface with IP address 168.10.1.1, does IGRP advertise 168.10.0.0?

____

____

____

12 Create a configuration for IGRP on a router with these interfaces and addresses: e0 using 10.1.1.1, e1 using 224.1.2.3, s0 using 10.1.2.1, and s1 using 199.1.1.1. Use process ID 5.

____

____

____

The answers to the "Do I Know This Already?" quiz are found in Appendix A, "Answers to the 'Do I Know This Already?' Quizzes and Q&A Sections." The suggested choices for your next step are as follows:

- **6 or less overall score**—Read the entire chapter. This includes the "Foundation Topics" and "Foundation Summary" sections, the "Q&A" section, and the scenarios at the end of the chapter.
- **3 or less on any quizlet**—Review the subsections of the "Foundation Topics" section, based on Table 7-1. Then move to the "Foundation Summary" section, the "Q&A" section, and the scenarios at the end of the chapter.
- **9 to 10 overall score**—Begin with the "Foundation Summary" section, and then go to the "Q&A" section and the scenarios at the end of the chapter.
- **11 or 12 overall score**—If you want more review of these topics, skip to the "Foundation Summary" section and then go to the "Q&A" section and the scenarios at the end of the chapter. Otherwise, move to the next chapter.

Foundation Topics

Distance Vector Routing Protocols

52 List problems that each routing type encounters when dealing with topology changes, and describe techniques to reduce the number of these problems.

To succeed on the CCNA exam, you need to be able to describe the logic behind distance vector routing protocols, as well as describe the operation of two IP distance vector routing protocols, RIP and IGRP. You also need to be able to configure and examine the operation of RIP and IGRP. If you understood most of the preceding chapter, you can learn enough to configure and use RIP and IGRP in about two pages of this book. However, you will not learn enough about how routing protocols do their job—and that underlying theory is covered on the exam. The first section of this chapter covers the basics of distance vector routing protocols, including terminology and operation. Then details are given. If you are looking to save a little time, consider skipping the more detailed section on how distance vector logic works.

Terminology can get in the way when you're learning about routing protocols. This book's terminology relating to routing and routing protocols is consistent with the courses in the Cisco CCNA training path, as well as with most Cisco documentation. The first term that needs to be defined is *routing protocol*. This term can be contrasted with *routed protocol*. Chapter 3 provides a silly, but I hope memorable, story (the "Ted and Ting" story) that can help distinguish between these two terms. Three definitions follow:

- A *routing protocol* fills the routing table with routing information. Examples include RIP and IGRP.
- A *routed protocol* is a protocol that has an OSI Layer 3 equivalent specification that defines logical addressing and routing. The packets defined by the network layer (Layer 3) portion of these protocols can be routed. Examples of protocols include IP and IPX.
- The term *routing type* might appear on questions remaining from the older CCNA exam, #640-407. This term refers to the type of routing protocol, such as link-state.

IP routing protocols fill the IP routing table with valid, (hopefully) loop-free routes. As you will see later, distance vector routing protocols have many features that prevent loops. Each route includes a subnet number, the interface out which to forward packets so that they are delivered to that subnet, and the IP address of the next router that should receive packets destined for that subnet (if needed). An analogy to routing is the process a stubborn man might use when taking a trip to somewhere he has never been. He might look for a road sign referring to the destination town and pointing him to the next turn. By repeating the process at each intersection, he should eventually make it to the correct town. Of course, if a routing loop occurs (in other words, he's lost!) and he stubbornly never asks for directions, he could loop forever—or at least until he runs out of gas.

Before I discuss the underlying logic, you need to consider the goals of a routing protocol. The goals described in the following list are common for any IP routing protocol, regardless of its underlying logic type:

- To dynamically learn and fill the routing table with a route to all subnets in the network.
- If more than one route to a subnet is available, to place the best route in the routing table.
- To notice when routes in the table are no longer valid, and to remove those routes from the routing table.
- If a route is removed from the routing table and another route through another neighboring router is available, to add the route to the routing table. (Many people view this goal and the preceding one as a single goal.)
- To add new routes, or to replace lost routes with the best currently available route, as quickly as possible. The time between losing the route and finding a working replacement route is called *convergence* time.
- To prevent routing loops.

Overview of Routing Protocols

Several routing protocols for TCP/IP exist. IP's long history and continued popularity have called for the specification and creation of several different competing options. So, classifying IP routing protocols based on their differences is useful—and also is a fair topic for exam questions.

For the CCNA exam, you must know the terminology and routing protocols discussed here. You must also have a deeper understanding of distance vector protocols, which are described in upcoming sections.

Routing protocols fill each router's routing table with valid, useful routes. With good routes in the routing table, a router can forward packets—which is what a router is supposed to do. One story that helps describe the process is to equate routing and routing protocols to your taking a trip, and the government workers who create road signs. You can take a trip without a map, relying on road signs to tell you which road to take at each intersection, until you get to your destination. However, if you eliminate the overhead function of the government employees who create road signs that tell you where to turn, you will not know where to go, and you will never get there. Likewise, routing protocols are overhead functions, but really useful ones.

Routing protocols can be categorized in several ways, and each requires a definition. One distinction is whether the protocol is more useful between two companies or inside a single company. Another distinction depends on how the routing protocol's internal logic behaves.

Table 7-2 lists some of the terms you need to know for the CCNA exam. After that, you will read about the basics, followed by a summary table that pretty much completes the level of understanding you need for the exam for most routing protocols. More details will follow about RIP and IGRP.

Table 7-2 *Routing Protocol Terminology*

Term	Definition
Routing protocol	A protocol whose purpose is to learn the available routes, place the best routes into the routing table, and remove routes when they are no longer valid.
Exterior routing protocol	A routing protocol designed for use between two different networks that are under the control of two different organizations. These are typically used between ISPs or between a company and an ISP. For example, a company would run BGP, an exterior routing protocol, between one of its routers and a router inside an ISP.
Interior routing protocol	A routing protocol designed for use in a network whose parts are under the control of a single organization. For example, an entire company might choose the IGRP routing protocol, which is an interior routing protocol.
Distance vector	The logic behind the behavior of some interior routing protocols, such as RIP and IGRP.
Link state	The logic behind the behavior of some interior routing protocols, such as OSPF.
Balanced hybrid	The logic behind the behavior of EIGRP, which is more like distance vector than link state but is different from these other two types of routing protocols.
Dijkstra Shortest Path First (SPF) algorithm	Magic math used by link-state protocols, such as OSPF, when the routing table is calculated.
Diffusing Update Algorithm (DUAL)	The process by which EIGRP routers collectively calculate routing tables.
Convergence	The time required for routers to react to changes in the network, removing bad routes and adding new, better routes so that the currently best routes are in all the routers' routing tables.

The CCNA exam focuses on interior routing protocols. If you are interested in pursuing CCIE or CCNP certification, understanding exterior routing protocols is important. An excellent learning tool and reference for IP routing and routing protocols is the Cisco Press book *Routing TCP/IP*, Volume I by Jeff Doyle.

OSPF and Link-State Protocols

One type of interior routing protocol is the *link-state protocol*. Link-state protocols use a topological database that is created on each router. Entries describing every router, every router's attached links, and every router's neighboring routers are included in the database so that each router can build a complete map of the network. The topology database is processed by an algorithm called the Dijkstra Shortest Path First (SPF) algorithm for choosing the best routes to add to the routing table. This detailed topology information, along with the Dijkstra algorithm, helps link-state protocols avoid loops and converge quickly.

Open Shortest Path First (OSPF) is a link-state routing protocol used for IP. Link-state protocols avoid routing loops by transmitting and keeping more detailed topology information, which allows the protocol to use calculations that prevent loops. With OSPF, the subnet mask information is also transmitted, allowing features such as VLSM and route summarization.

EIGRP and Balanced Hybrid Protocols

A second type of routing protocol is the *balanced hybrid protocol*. Balanced hybrid is a term created by Cisco to describe the inner workings of EIGRP, which uses the Diffusing Update Algorithm (DUAL) to calculate routes. A balanced hybrid protocol exchanges more topology information than a distance vector routing protocol, but it does not require full topology or the computation-intensive Dijkstra algorithm to compute loop-free routes.

Enhanced IGRP (EIGRP) is a balanced hybrid routing protocol. DUAL is the underlying algorithm. DUAL defines a method for each router to not only calculate the best current route to each subnet, but also to calculate alternative routes that could be used if the current route fails. An alternative route, using what DUAL defines as a neighboring *feasible successor route,* is guaranteed to be loop-free, so convergence can happen quickly. EIGRP also transmits the subnet mask for each routing entry. Therefore, features such as VLSM and route summarization are easily supported.

RIP, IGRP, and Distance Vector Protocols

RIP Version 1 (RIP-1) and IGRP are two distance vector routing protocols that are covered in depth on the CCNA exam. RIP and IGRP are similar in most details, with the big exception being that IGRP uses a much more robust metric. Both RIP-1 and IGRP are covered in more detail later in this chapter.

Some routing protocols are less likely to be covered on the CCNA exam, including RIP Version 2 (RIP-2). RIP-2 includes many improvements over RIP-1. Most notably, the subnet mask associated with each advertised route is included in the routing update. The mask allows routers to use features such as variable-length subnet masks (VLSMs) and route summarization—features sure to be covered on the CCNP exam.

Table 7-3 lists interior IP routing protocols and their types. A column specifying whether the routing protocol includes subnet mask information in the routing updates is listed for future reference.

Table 7-3 *Interior IP Routing Protocols and Types*

Routing Protocol	Type	Loop-Prevention Mechanisms	Mask Sent in Updates, Which Allows VLSM?
RIP-1	Distance vector	Hold-down timer, split horizon	No
RIP-2	Distance vector	Hold-down timer, split horizon	Yes
IGRP	Distance vector	Hold-down timer, split horizon	No
EIGRP	Balanced hybrid	DUAL and feasible successors	Yes
OSPF	Link-state	Dijkstra SPF algorithm and full topology knowledge	Yes

Distance Vector Routing Protocol Behavior

CCNAs deal with routing problems on a daily basis. Some of these problems are the result of the logic behind distance vector routing protocols. Understanding what distance vector routing means is to understand how a routing protocol accomplishes the following goals:

- Learning routing information
- Noticing failed routes
- Adding the current best route after one has failed
- Preventing loops

The following list summarizes the behavior of a router that uses the RIP-1 or IGRP distance vector routing protocols:

- Routers add *directly connected* subnets to their routing tables. Routers do not need to run a routing protocol to learn connected routes, but connected subnet routes are advertised to neighboring routers by the routing protocol.
- Routers broadcast or multicast *routing updates* to their neighboring routers. This is so that all neighboring routers can learn routes via the single broadcast or multicast update.
- Routers listen for *routing updates* from their neighbors so that they can learn new routes.
- Routers measure how good one route is versus another using a *metric*. The metric describes how good the route is in case a router hears of more than one route to a particular subnet. If multiple routes to the same subnet are learned, the lower-metric route is used.
- Routers include basic *topology* information in routing updates, including, at a minimum, the subnet and metric information.

- Routers send *periodic updates* and expect to receive periodic updates from neighboring routers. Failure to receive updates from a neighbor in a timely manner results in the removal of the routes previously learned from that neighbor.
- A router assumes that, for a route advertised by Router X, the *next-hop router* in that route is Router X.

Figure 7-2 demonstrates how Router A's directly connected subnets are advertised to Router B. In this case, Router A advertises two directly connected routes.

Figure 7-2 *Router A Advertising Directly Connected Routes*

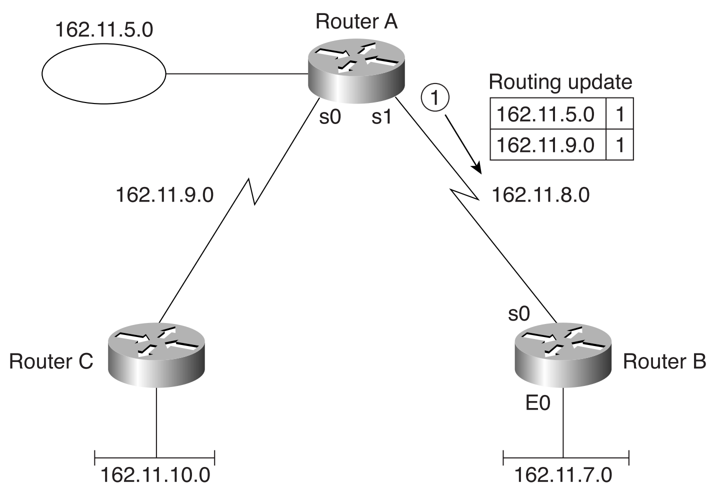


Table 7-4 shows the resulting routing table on Router B.

Table 7-4 *Routing B Routing Table After Receiving the Update Shown in Figure 7-2*

Group (Mask Is 255.255.255.0)	Outgoing Interface	Next-Hop Router	Comments
162.11.5.0	S0	162.11.8.1	This is one of two routes learned via the update in the figure.
162.11.7.0	E0	N/A	This is a directly connected route.
162.11.8.0	S0	N/A	This is a directly connected route.
162.11.9.0	S0	162.11.8.1	This is one of two routes learned via the update in the figure.

Two interesting facts about what a Cisco IOS Software-based Cisco router puts in the routing table become obvious in this example. First, just like for any other directly connected route, the two directly connected routes on Router B do not have an entry in the Next-Hop Router field, because packets to those subnets can be sent directly to hosts in those subnets. In other words, there is no need for Router B to forward packets to another router, because Router B is attached. The second interesting fact is that the Next-Hop Router entries for the routes learned from Router A show Router A's IP address as the next router. In other words, a route learned from a neighboring router goes through that router. Router B typically learns Router A's IP address for these routes simply by looking at the routing update's source IP address.

The next example gives some insight into the metric's cumulative effect. A subnet learned via an update from a neighbor is advertised, but with a higher metric. Just like a road sign in Decatur, Ga. might say "Turn here to get to Snellville, 14 miles," another road sign farther away in Atlanta might read "Turn here to get to Snellville, 22 miles." By taking the turn in Atlanta, 8 miles later, you end up in Decatur, looking at the road sign that tells you the next turn to get to Snellville, which is then 14 miles away. This example shows you exactly what happens with RIP, which uses hop count as the metric. Figure 7-3 and Table 7-5 illustrate this concept.

Figure 7-3 *Router A Advertising Routes Learned from Router C*

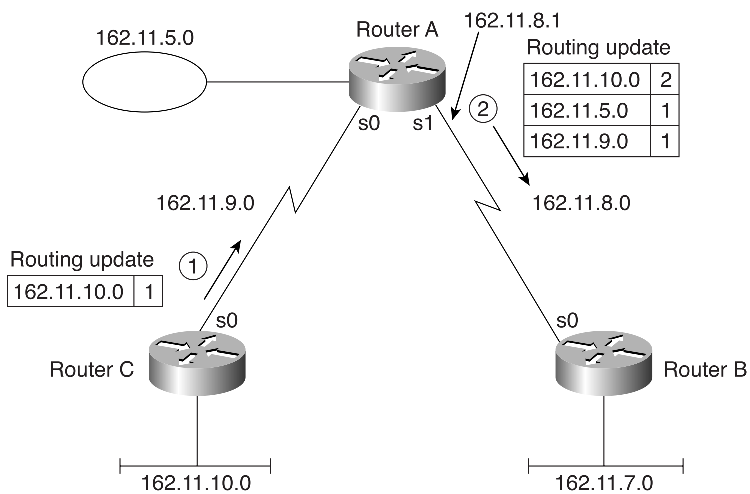


Table 7-5 *Router B Routing Table After Receiving the Update Shown in Figure 7-3*

Group	Outgoing Interface	Next-Hop Router	Metric	Comments
162.11.5.0	S0	162.11.8.1	1	This is the same route that was learned earlier.
162.11.7.0	E0	N/A	0	This is a directly connected route.

Table 7-5 *Router B Routing Table After Receiving the Update Shown in Figure 7-3 (Continued)*

Group	Outgoing Interface	Next-Hop Router	Metric	Comments
162.11.8.0	S0	N/A	0	This is a directly connected route.
162.11.9.0	S0	162.11.8.1	1	This one was also learned earlier.
162.11.10.0	S0	162.11.8.1	2	This one was learned from Router A, which learned it from Router C.

Router B believes some subnets are nearer than others, based on the metric. Router B's metric for connected routes is 0 because there is no router between B and those subnets. B's metric for routes directly connected to the router is 1 because Router A is between B and those subnets. Finally, the metric for subnet 162.11.10.0 is 2, because, from B's perspective, two routers separate it from that subnet—namely, Router A and Router C.

The origin of the term *distance vector* becomes more apparent with this example. The route to 162.11.10.0 that Router B adds to its routing table refers to Router A as the next router because Router B learned the route from Router A; Router B knows nothing about the network topology on the "other side" of Router A. So, B has a vector (send packets to Router A) and a distance (2) for the route to subnet 10, but no other details!

The next core concept of distance vector routing protocols relates to when to doubt the validity of routing information. Each router sends periodic routing updates. A routing update timer, which is equal on all routers, determines how often the updates are sent. The absence of routing updates for a preset number of routing timer intervals results in the removal of the routes previously learned from the router that has become silent.

You have read about the basic, core concepts for distance vector protocols. The next section provides a little deeper look into issues when redundancy exists in the network.

Distance Vector: Advanced Concepts

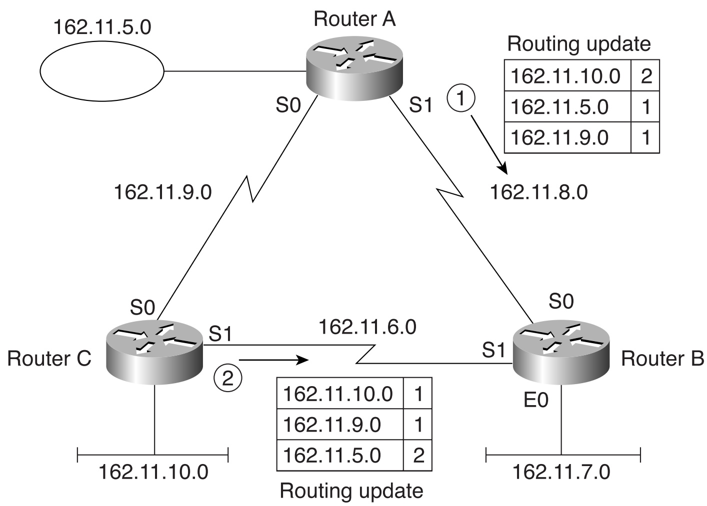
Two general categories of topics are explained in the rest of this section. First, when multiple physical paths exist in a network, multiple routes to the same network can exist. If that occurs, the metrics could tie, so which one do you use? Well, it depends, so we will take a closer look.

The second category of topics relates to loop avoidance. Just like Spanning-Tree Protocol avoids loops in a switched or bridged LAN, IP routing protocols avoid loops in the routed IP network. Distance vector protocols have several features that are needed in order to prevent loops—and these features are covered here.

Issues When Multiple Routes to the Same Subnet Exist

A router might learn one route and then learn a better route. Of course, the better route should replace the higher-metric route when this happens. Figure 7-4 outlines just such a case. Table 7-6 shows Router B's routing table when only one route to 162.11.10.0 is known, before the serial link between B and C comes up. Table 7-7 shows Router B's routing table after the link between B and C comes up, learning about another route to that same subnet.

Figure 7-4 *Routers A and C Advertising to Router B*



NOTE The routing updates in Figure 7-4 show only the information needed for the point being made in this example. Other routes that would normally be in the routing update have been omitted.

Table 7-6 *Router B Routing Table While Router B Serial 1 Is Down*

Group	Outgoing Interface	Next-Hop Router	Metric	Comments
162.11.5.0	S0	162.11.8.1	1	N/A
162.11.7.0	E0	N/A	0	N/A
162.11.8.0	S0	N/A	0	N/A
162.11.9.0	S0	162.11.8.1	1	The metric 1 route is learned from Router A.
162.11.10.0	S0	162.11.8.1	2	Currently, the best route is through Router A because the link to C is down.

Table 7-7 *Router B Routing Table After Learning a Second Valid Route to Subnet 162.11.10.0*

Group	Outgoing Interface	Next-Hop Router	Metric	Comments
162.11.5.0	S0	162.11.8.1	1	N/A
162.11.6.0	S1	N/A	0	This route was added because it is directly connected to S1, which is now up and operational.
162.11.7.0	E0	N/A	0	N/A
162.11.8.0	S0	N/A	0	N/A
162.11.9.0	S0	162.11.8.1	1	The metric 1 route was learned from Router A, but Router C also advertises a metric 1 route! Only one route is chosen—the first one that was learned.
162.11.10.0	S1	162.11.6.2	1	A better route replaces the old route. The new route has a smaller metric and points directly out S1 toward Router C.

Router B changed only one route in this case in reaction to the new routing updates from Router C. B changed its route to 162.11.10.0 because the metric for the route through Router C (metric 1) is smaller than the one from Router A (metric 2).

Router B added only one route in this example—the directly connected subnet 162.11.6.0. The route was added not because of this distance vector routing protocol; it was added by Router B because it is a directly connected subnet and because that interface is now up.

Router B did not change its route to subnet 162.11.9.0, pointing through Router A, even though another metric 1 route through Router C was learned. *In this case, the route that was already in the table is left in the table, which is a reasonable choice.* Other options in the Cisco IOS Software include adding up to six equal-cost routes in the routing table and balancing across those routes instead of using just a single route.

Avoiding Loops with Distance Vector Protocols

Start Extra Credit

Routing protocols carry out their most important functions when redundancy exists in the network. Most importantly, routing protocols ensure that the currently best routes are in the routing tables by reacting to network topology changes. Routing protocols also prevent loops.

Distance vector protocols need several mechanisms to prevent loops. Table 7-8 summarizes these issues and lists the solutions, which are explained in the upcoming text.

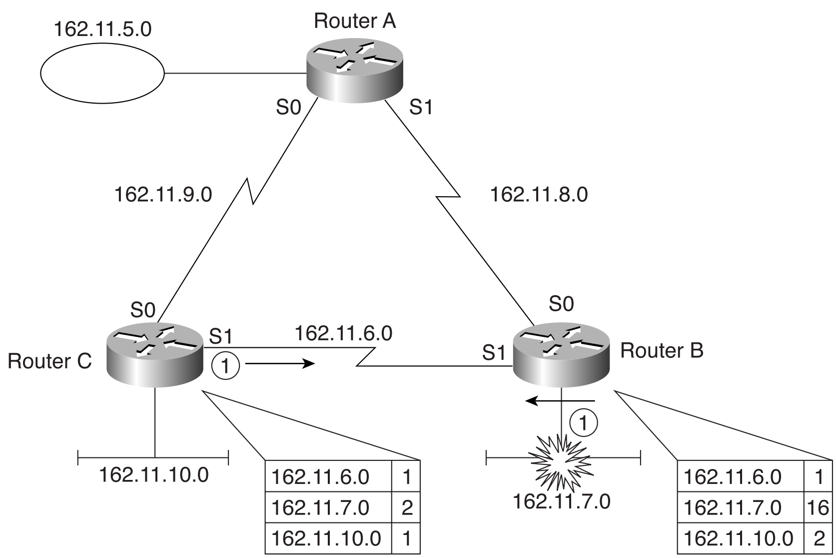
Table 7-8 *Issues Relating to Distance Vector Routing Protocols in Networks with Multiple Paths*

Issue	Solution
Multiple routes to the same subnet have equal metrics	Implementation options involve either using the first route learned or putting multiple routes to the same subnet in the routing table.
Routing loops occur due to updates passing each other over a single link	**Split horizon**—The routing protocol advertises routes out an interface only if they were not learned from updates entering that interface. **Split horizon with poison reverse**—The routing protocol uses split-horizon rules unless a route fails. In that case, the route is advertised out all interfaces, but with infinite-distance metrics.
Routing loops occur due to updates passing each other over alternative paths	**Route poisoning**—When a route to a subnet fails, the subnet is advertised with an infinite-distance metric.
Counting to infinity	**Hold-down timer**—After finding out that a route to a subnet has failed, a router waits a certain period of time before believing any other routing information about that subnet. **Triggered updates**—When a route fails, an update is sent immediately rather than waiting on the update timer to expire. Used in conjunction with route poisoning, this ensures that all routers know of failed routes before any hold-down timers can expire.

Split Horizon Routing loops can occur with distance vector routing protocols due to how quickly a router learns about a failed route. In other words, if a route fails, but Router B does not yet know it is bad, Router B advertises a route to that subnet as still being a good route. Split horizon overcomes timing issues over a single full-duplex link. Figure 7-5 shows an example of this problem.

NOTE The routing updates in Figure 7-5 show only the information needed for the point being made in this example. Other routes that would normally be in the routing update have been omitted.

Figure 7-5 *Advertisements Passing on the Serial Link for Subnet 162.11.7.0*



In Figure 7-5, the routing updates are sent periodically, and every router's timer is independent of the others! There is no requirement to make the updates flow from Routers C and B at the same time; however, in this case, Routers B and C are sending updates at the exact same time. How often does this happen? Well, statistically, more often than you might think. And if everything is working, there is no problem. It becomes a problem when Router B advertises an infinite-distance (metric) route to 162.11.7.0, because the subnet just failed. However, Router C advertises a metric 2 route to subnet 162.11.7.0, across the serial link to Router B, that passes the update from Router B. Tables 7-9 and 7-10 show the resulting routing table entries, with a reference to the metric values.

Table 7-9 *Router B Routing Table After Subnet 162.11.7.0 Fails and an Update from Router C Is Received*

Group	Outgoing Interface	Next-Hop Router	Metric	Comments
162.11.6.0	S1	N/A	0	N/A
162.11.7.0	S1	N/A	2	The old route failed, but this one is heard from Router C.
162.11.10.0	S1	162.11.6.2	1	N/A

Table 7-10 *Router C Routing Table After Subnet 162.11.7.0 Fails and an Update from Router B Is Received*

Group	Outgoing Interface	Next-Hop Router	Metric	Comments
162.11.6.0	S1	N/A	0	N/A
162.11.7.0	S1	N/A	16	The old route was metric 1 through Router B. Now B claims that the metric is infinite, so the route must have failed!
162.11.10.0	E0	N/A	1	N/A

NOTE In this chapter, the value 16 is used to represent an infinite metric. RIP uses 16 to represent infinite. IGRP uses a delay value of more than 4 billion to imply an infinite-distance route.

Now Router C has an infinite-distance route to 162.11.7.0, but Router B has a route to the same subnet, pointing through Router C. In its last update, Router C claimed to have a metric 2 route to 162.11.7.0 at the same time it was receiving an update from Router B that the route to 162.11.7.0 was no longer valid. (Infinity is shown as the value 16 in Table 7-10, which is RIP's value for infinity.) So Router B now thinks that 162.11.7.0 can be reached through Router C, and Router C thinks that 162.11.7.0 is unreachable.

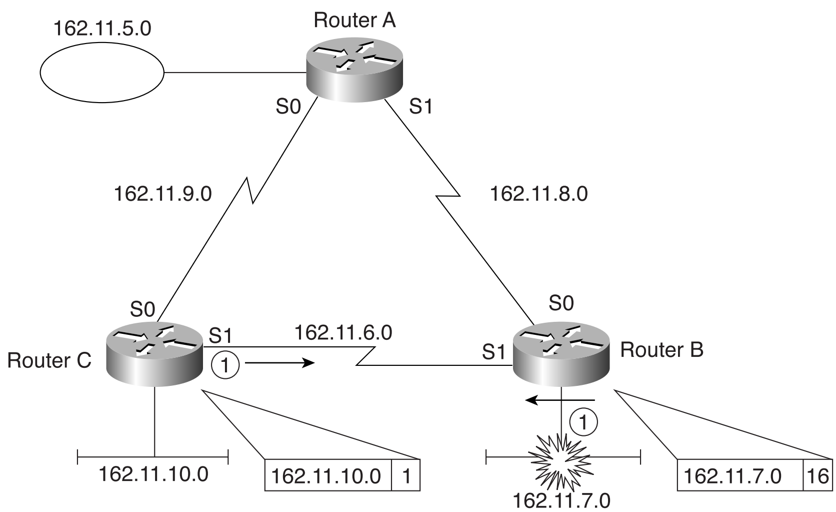
This process repeats itself with the next routing update, because it is sent the next time the routing protocol timer expires. This time, Router B advertises metric 3 and Router C advertises an infinite (bad) metric for subnet 162.11.7.0. This continues until both numbers reach infinity. Thankfully, each distance vector routing protocol implementation sets a metric value for which the number is considered infinite. For example, 16 is infinite for RIP, and 4,294,967,295 is infinite for IGRP.

In this case, *split horizon* is the solution to the problem of counting to infinity. Split horizon can be briefly summarized as follows:

> All routes with outgoing interface *x* are not included in updates sent out that same interface *x*.

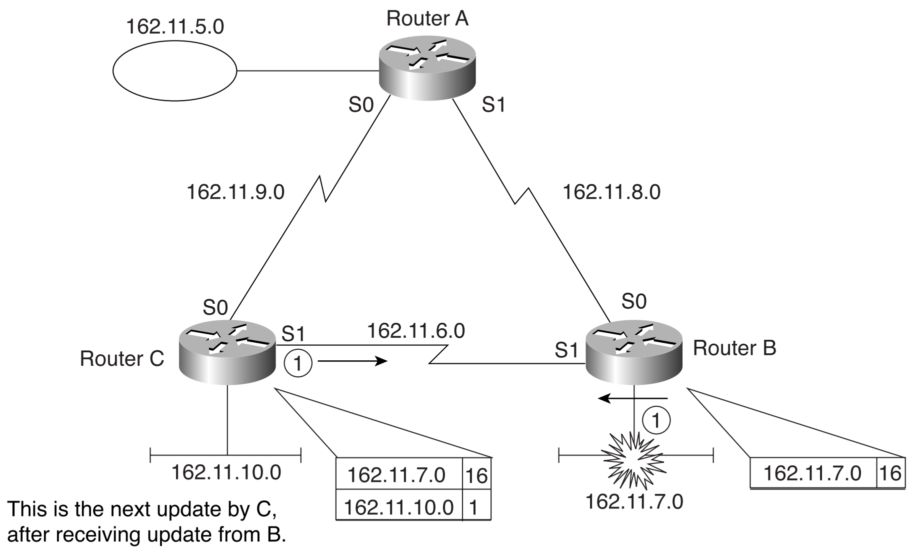
For example, in Figure 7-6, C's route to subnet 162.11.7.0 points out Serial 1, so its update sent out Serial 1 does not advertise subnet 162.11.7.0. So, when Routers B and C's updates pass each other, one update shows the route with an infinite metric, and the other says nothing about the route, so the counting-to-infinity problem goes away.

Figure 7-6 *Split Horizon Preventing Subnet 162.11.7.0 from Being Advertised by Router C*



So far, you have read about how split horizon works. However, Cisco distance vector routing protocols actually use a variant of split horizon called *split horizon with poison reverse* (or simply *poison reverse*). When the network is stable, it works just like plain old split horizon. When a route is advertised with an infinite metric, the recipient of the update advertises an infinite-metric route to that subnet out *all* interfaces—including interfaces typically prevented by split horizon. Figure 7-7 lists the pertinent contents of the routing update from Router C, using split horizon with poison reverse.

Figure 7-7 *Split Horizon Enabled with Poison Reverse*



Hold-Down Timer Split horizon defeats the counting-to-infinity problem over a single link. However, counting to infinity can occur in redundant networks (networks with multiple paths) even with split horizon enabled. The *hold-down timer* is part of a solution to the counting-to-infinity problem when networks have multiple paths to many subnets. Split horizon does not defeat the counting-to-infinity problem in all topologies. An additional solution is required that includes a hold-down timer and a routing update feature called route poisoning.

Figure 7-8 shows the version of the counting-to-infinity problem that split horizon does not solve but that holddown does. Subnet 162.11.7.0 fails again (someone should check the cabling!), and Router B advertises an infinite-metric route to 162.11.7.0. However, Router A's update timer expires at the same time (shown as number 1 in the figure), so the updates sent by Routers A and B occur at the same time. Router C hears of an infinite-distance metric to that subnet from Router B and a metric 2 route from Router A, so Router C uses the metric 2 route. Table 7-11 lists the pertinent information about Router C's routing table entry for subnet 162.11.7.0, after the updates shown in Figure 7-8.

Figure 7-8 *Counting to Infinity with a Need for Holddown*

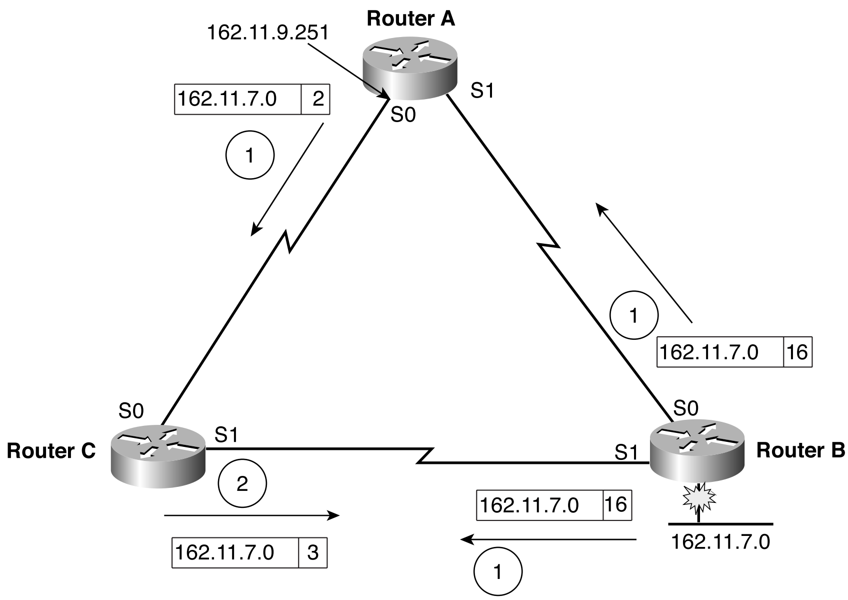


Table 7-11 *Router C's Routing Table After the Updates Shown in Figure 7-8 Are Received*

Group	Outgoing Interface	Next-Hop Router	Metric	Comments
162.11.7.0	S0	162.11.9.251	2	Used to point directly to Router B. Now the route points through Router A, metric 2.

Router C now thinks it has a valid route to 162.11.7.0, pointing back to Router A. On Router B's next update, shown in Step 2 of Figure 7-8, Router C does not advertise subnet 162.11.7.0 out S0 due to split-horizon rules. However, Router C advertises 162.11.7.0 to Router B out Serial1 with metric 3. Router C incorrectly believes that the route to 162.11.7.0 exists through Router B. Router C, in its next update, also tells Router A that it has a route to 162.11.7.0. So, counting to infinity occurs.

The solution is to enable a hold-down timer. The concept is simple—once you see the problem. Holddown is defined as follows:

> When learning that a route has failed, ignore any information about an alternative route to that subnet for a time equal to the hold-down timer.

With holddown enabled, Router C would not believe the metric 2 route learned from Router A in Step 1 of Figure 7-8. During the same time, Router B would advertise infinite-metric routes to 162.11.7.0 to Routers A and C, and then Router A would start advertising an infinite-distance route to Router C. In effect, all routers ignore good routing information about that subnet until enough time passes that everyone has heard the old, bad information.

Route Poisoning When a route fails, a routing protocol could choose to simply stop advertising that route. In fact, distance-vector protocols did just that when they were first created. However, there was no way for a router receiving updates to know the difference between the silence after a route fails and a lost routing update due to noise on the link. So route poisoning was added.

Route poisoning causes a routing protocol to advertise infinite-metric routes for a failed route. It's a lot like poison reverse—in fact, poison reverse is a subset of route poisoning. The formal definition of route poisoning is as follows:

> When a route fails, advertise an infinite-metric route for that subnet out the same interfaces on which the route previously was advertised.

In fact, in Figure 7-8, the update sent by Router B in Step 1 is an example of route poisoning. There is a difference between route poisoning and split horizon with poison reverse. Route poisoning does not break split-horizon rules. Split horizon with poison reverse is essentially route poisoning, but specifically out links on which split horizon would normally not allow routing information to flow. In either case, the result is that failed routes are advertised with infinite metrics!

Triggered (Flash) Updates When a router notices that a directly connected subnet has changed state, it waits until its update timer expires before sending another routing update. The update includes an infinite-metric route based on route poisoning and poison reverse. However, why not go ahead and send updates because routing information has changed! That is exactly what flash or triggered updates are. The router immediately sends another routing update on its other interfaces rather than waiting on the routing update timer to expire. This causes the information about the route whose status has changed to be forwarded more quickly and also starts the hold-down timers more quickly on the neighboring routers.

End Extra Credit

RIP and IGRP

To pass the CCNA exam, you need to know the particulars of how RIP and IGRP implement distance vector logic. RIP and IGRP both use distance vector logic, so they are similar in many respects. A couple of major differences exist, however; they are explained in the upcoming section. Table 7-12 outlines the features of RIP and IGRP.

Table 7-12 *RIP and IGRP Feature Comparison*

Feature	RIP (Default)	IGRP (Default)
Update timer	30 seconds	90 seconds
Metric	Hop count	Function of bandwidth and delay (the default). Can include reliability, load, and MTU.
Hold-down timer	180	280
Flash (triggered) updates	Yes	Yes
Mask sent in update	No for RIP-1; yes for RIP-2	No
Infinite-metric value	16	4,294,967,295

The IGRP metric provides a better measurement of how good a route is, as compared with RIP's metric. IGRP's metric is calculated using the bandwidth and delay settings on the interface on which the update was received. When bandwidth and delay are used, the metric is more meaningful than hop count; longer hop routes that go over faster links are considered better routes by IGRP.

RIP uses hop count as its metric. When an update is received, the metric for each subnet in the update signifies the number of routers between the router receiving the update and each subnet. Before sending an update, a router increments its metric for routes to each subnet by 1.

Finally, the issue of whether the mask is sent is particularly important if VLSMs in the same network are desired. This topic is discussed in the next section.

Configuring RIP and IGRP

27 Add the RIP routing protocol to your configuration.

28 Add the IGRP routing protocol to your configuration.

The CCNA exam covers configuration for two routing protocols—RIP and IGRP. You can configure each of them if you master the use of the **network** command. Other than that, configuration is relatively easy. You should also know the more-popular **show** and **debug** commands, which help you examine and troubleshoot routing protocols.

Of course, you can configure many optional parameters. Some optional features simply are not within the scope of the CCNA exam—and those features are not discussed in this book. I have included some coverage of the additional features that are fair game on the exam. However, you might not see many, if any, more-detailed routing protocol configuration questions on the exam. Therefore, how much you study the additional features is up to you.

Tables 7-13 and 7-14 summarize the more popular commands used for RIP and IGRP configuration and verification. Two configuration examples follow.

Table 7-13 *IP RIP and IGRP Configuration Commands*

Command	Configuration Mode
router rip	Global
router igrp *as-number*	Global
network *net-number*	Router subcommand
passive-interface [**default**] {*interface-type interface-number*}	Router subcommand
maximum-paths *number-paths*	Router subcommand
variance *multiplier*	Router subcommand
traffic-share {*balanced* | *min*}	Router subcommand

Table 7-14 *IP RIP and IGRP EXEC Commands*

Command	Description
show ip route [*ip-address* [*mask*] [**longer-prefixes**]] | [*protocol* [*process-id*]]	Shows the entire routing table, or a subset if parameters are entered.
show ip protocols	Shows routing protocol parameters and current timer values.
debug ip rip	Issues log messages for each RIP update.
debug ip igrp transactions [*ip-address*]	Issues log messages with details of the IGRP updates.
debug ip igrp events [*ip-address*]	Issues log messages for each IGRP packet.
ping [*protocol* | **tag**] {*host-name* | *system-address*}	Sends and receives ICMP echo messages to verify connectivity.
trace [*protocol*] [*destination*]	Sends a series of ICMP echoes with increasing TTL values to verify the current route to a host.

Basic RIP and IGRP Configuration

Each **network** command enables RIP or IGRP on a set of interfaces. You must understand the subtleties of the **network** command, as explained in this section. However, what “enables” really means in this case is not obvious from the Cisco IOS Software documentation. Also, the parameters for the **network** command are not intuitive to many people who are new to Cisco IOS configuration commands. Therefore, routing protocol configuration, including the **network** command, is a likely topic for tricky questions on the exam.

The **network** command "matches" one or more interfaces on a router. For each interface, the **network** command causes the router to do three things:

- The router broadcasts or multicasts routing updates out an interface.
- The router listens for incoming updates on that same interface.
- The router, when sending an update, includes the subnet off that interface in the routing update.

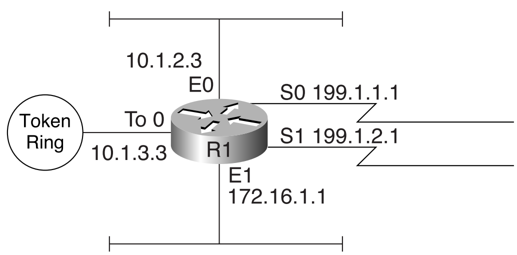
All you need to know is how to match interfaces using the **network** command. The router matches interfaces with the **network** command by asking this simple question:

> Which of my interfaces have IP addresses with the same network number referenced in this **network** subcommand?

For all interfaces that match the **network** command, the router does the three things just listed.

Examples give you a much better understanding of the **network** command, so examine Figure 7-9 and Example 7-1.

Figure 7-9 *Sample Router with Five Interfaces*



Example 7-1 *Sample Router Configuration with RIP Partially Enabled*

```
interface ethernet 0
ip address 10.1.2.3 255.255.255.0
interface ethernet 1
ip address 172.16.1.1 255.255.255.0
interface tokenring 0
ip address 10.1.3.3 255.255.255.0
interface serial 0
ip address 199.1.1.1 255.255.255.0
interface serial 1
ip address 199.1.2.1 255.255.255.0
!
router rip
network 10.0.0.0
network 199.1.1.0
```

The RIP configuration includes three commands in this case. The **router rip** global command moves the user from global configuration mode to RIP configuration mode. Then, two network commands appear, each with a different Class A, B, or C network number. So what interfaces were matched, and what did this accomplish? Well, if the goal is to enable RIP on *all* interfaces, the configuration is incomplete. Table 7-15 summarizes what this configuration accomplishes and what it does not.

Table 7-15 *What Happens with the RIP Configuration Shown in Example 7-1*

network Command	Interfaces Matched	Actions Taken
network 10.0.0.0	Token0, Ethernet0	Updates are sent out Token0 and Ethernet0. Listen for updates entering Token0 and Ethernet0. Advertise subnets 10.1.3.0 (Token0's subnet) and 10.1.2.0 (Ethernet0's subnet).
network 199.1.1.0	Serial0	Updates are sent out Serial0. Listen for updates entering Serial0. Advertise subnet 199.1.1.0 (Serial0's subnet).

For any interfaces that have IP addresses with the same network number referenced in this **network** subcommand, routing updates are broadcast and listened for, and the connected subnet is advertised. The **network** command requires a network number, not a subnet number, for the parameter. Interestingly, you can type a subnet number in the command, and the Cisco IOS Software changes the parameter to the network number in which that subnet resides.

If the goal was to configure RIP for all interfaces, a common mistake was made in this example. No **network** command matches interfaces Serial1 and Ethernet1. Example 7-2 shows the configuration process to add the additional network commands.

Example 7-2 *Completing the RIP Configuration from Example 7-1*

```
Router1#configure terminal
Router1(config)#router rip
Router1(config-router)#network 199.1.2.0
Router1(config-router)#network 172.16.0.0
Router1(config-router)#CTL-Z
Router1#
```

IGRP Configuration

You configure IGRP just like RIP, except that the **router igrp** command has an additional parameter—the AS number. All that is needed is for all routers to use the same process-id in order for IGRP to work. In Example 7-3, a complete sample IGRP configuration causes the router to advertise all connected subnets, to listen on all interfaces for IGRP updates, and to advertise on all interfaces.

Example 7-3 *Sample IGRP Configuration and* **show ip route** *Command Output*

```
interface ethernet 0
ip address 10.1.2.3 255.255.255.0
interface ethernet 1
ip address 172.16.1.1 255.255.255.0
interface tokenring 0
ip address 10.1.3.3 255.255.255.0
interface serial 0
ip address 199.1.1.1 255.255.255.0
interface serial 1
ip address 199.1.2.1 255.255.255.0
!
router igrp 1
 network 10.0.0.0
 network 199.1.1.0
 network 199.1.2.0
 network 172.16.0.0

Router1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 3 subnets
C       10.1.3.0 is directly connected, TokenRing0
C       10.1.2.0 is directly connected, Ethernet0
  I       10.1.4.0 [100/8539] via 10.1.2.14, 00:00:50, Ethernet0
     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.1.0 is directly connected, Ethernet1
I       172.16.2.0 [100/6244] via 172.16.1.44, 00:00:20, Ethernet1
C    199.1.1.0/24 is directly connected, Serial0
C    199.1.2.0/24 is directly connected, Serial1
```

IGRP configuration begins with the **router igrp 1** global configuration command. Then, four consecutive network commands match all the interfaces on the router, so that IGRP is fully enabled. In fact, the **network** commands are identical to the **network** commands in the complete RIP configuration.

IGRP Metrics

IGRP uses a composite metric. This metric is calculated as a function of bandwidth, delay, load, and reliability. By default, only bandwidth and delay are considered; the other parameters are considered only if they are enabled via configuration. Delay and bandwidth are not measured values but are set via the **delay** and **bandwidth** interface subcommands. (The same formula is used to calculate the metric for EIGRP, but with a scaling factor so that the actual metric values are larger, allowing more granularity in the metric.)

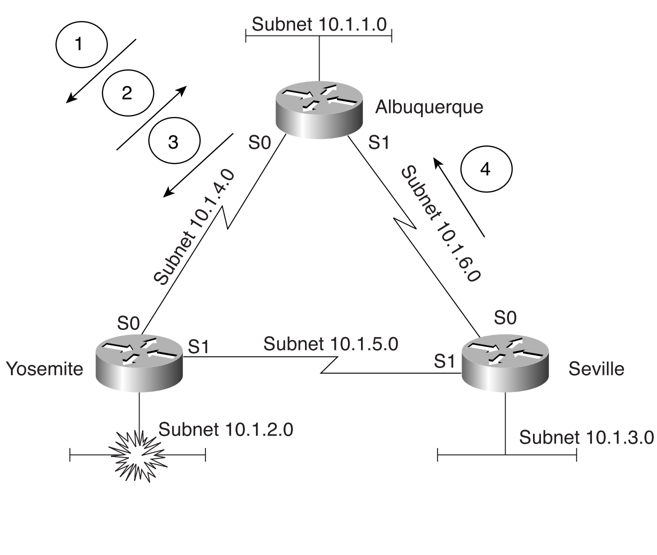
The **show ip route** command in Example 7-3 shows the IGRP metric values in brackets. For example, the route to 10.1.4.0 shows the value [100/8539] beside the subnet number. The metric 8539 is a single value, as calculated based on bandwidth and delay. The metric is calculated (by default) as the sum of the inverse of the minimum bandwidth, plus the cumulative delay on all links in the route. *In other words, the higher the bandwidth, the lower the metric; the lower the cumulative delay, the lower the metric.*

Examination of RIP and IGRP **debug** and **show** Commands

This section on basic RIP and IGRP configuration closes with one more sample network that is first configured with RIP and then is configured with IGRP. Advanced distance vector protocol concepts, such as split horizon and route poisoning, become more obvious when you look at these examples. RIP and IGRP implement split horizon and route poisoning. You can better understand them by examining the upcoming **debug** messages.

First, Figure 7-10 and Example 7-4 show a stable RIP network with split-horizon rules that affect the RIP updates. Then Ethernet 0 on Yosemite is shut down, and Yosemite advertises an infinite-distance route to 10.1.2.0 because route poisoning is in effect, as shown in Example 7-5. The numbered arrows in the figure represent routing updates. The numbers are referred to with comments inside Example 7-4.

Figure 7-10 *Sample Three-Router Network with Subnet 10.1.2.0 Failing*



Example 7-4 *RIP Configuration and* **debug***s on Albuquerque*

```
interface ethernet 0
ip addr 10.1.1.251 255.255.255.0
interface serial 0
ip addr 10.1.4.251 255.255.255.0
interface serial 1
ip addr 10.1.6.251 255.255.255.0
!
router rip
network 10.0.0.0

Albuquerque#debug ip rip
RIP: received v1 update from 10.1.6.253 on Serial1
     10.1.3.0 in 1 hops
     10.1.2.0 in 2 hops
     10.1.5.0 in 1 hops
RIP: sending v1 update to 255.255.255.255 via Serial0 (10.1.4.251)
!             (POINT NUMBER 1)
     subnet  10.1.3.0, metric 2
     subnet  10.1.1.0, metric 1
     subnet  10.1.6.0, metric 1
RIP: sending v1 update to 255.255.255.255 via Serial1 (10.1.6.251)
     subnet  10.1.2.0, metric 2
     subnet  10.1.1.0, metric 1
     subnet  10.1.4.0, metric 1
RIP: sending v1 update to 255.255.255.255 via Ethernet0 (10.1.1.251)
     subnet  10.1.3.0, metric 2
     subnet  10.1.2.0, metric 2
     subnet  10.1.6.0, metric 1
     subnet  10.1.5.0, metric 2
     subnet  10.1.4.0, metric 1
RIP: received v1 update from 10.1.4.252 on Serial0
     10.1.3.0 in 2 hops
     10.1.2.0 in 1 hops
     10.1.5.0 in 1 hops
Albuquerque#
(Yosemite E0 shutdown at this time...)

RIP: received v1 update from 10.1.4.252 on Serial0
!             (POINT NUMBER 2)
     10.1.3.0 in 2 hops
     10.1.2.0 in 16 hops (inaccessible)
     10.1.5.0 in 1 hops
RIP: sending v1 update to 255.255.255.255 via Serial0 (10.1.4.251)
!             (POINT NUMBER 3)
     subnet  10.1.3.0, metric 2
     subnet  10.1.2.0, metric 16
     subnet  10.1.1.0, metric 1
     subnet  10.1.6.0, metric 1
RIP: sending v1 update to 255.255.255.255 via Serial1 (10.1.6.251)
     subnet  10.1.2.0, metric 16
     subnet  10.1.1.0, metric 1
```

continues

Example 7-4 *RIP Configuration and* **debug***s on Albuquerque (Continued)*

```
     subnet  10.1.4.0, metric 1
RIP: sending v1 update to 255.255.255.255 via Ethernet0 (10.1.1.251)
     subnet  10.1.3.0, metric 2
     subnet  10.1.2.0, metric 16
     subnet  10.1.6.0, metric 1
     subnet  10.1.5.0, metric 2
     subnet  10.1.4.0, metric 1
RIP: received v1 update from 10.1.6.253 on Serial1
!             (POINT NUMBER 4)
     10.1.3.0 in 1 hops
     10.1.2.0 in 16 hops (inaccessible)
     10.1.5.0 in 1 hops
```

Example 7-5 *RIP Configuration on Yosemite*

```
interface ethernet 0
ip addr 10.1.2.252 255.255.255.0
interface serial 0
ip addr 10.1.4.252 255.255.255.0
interface serial 1
ip addr 10.1.5.252 255.255.255.0

router rip
network 10.0.0.0
```

First, examine the configuration on Albuquerque (Example 7-4) and Yosemite (Example 7-5). Because all interfaces on each router are part of network 10.0.0.0, RIP needs only a single **network** command on each router, so the configuration is relatively easy.

For the rest of the explanation, refer to the phrase "Point Number *X*" in Example 7-4. The following list describes what happens at each point in the process:

- **Point Number 1**—Albuquerque sends an update out Serial0, obeying split-horizon rules. Notice that 10.1.2.0, Yosemite's Ethernet subnet, is not in the update sent out Albuquerque's S0 interface.
- **Point Number 2**—This point begins right after Yosemite's E0 has been shut down, simulating a failure. Albuquerque receives an update from Yosemite, entering Albuquerque's S0 interface. The route to 10.1.2.0 has an infinite metric, which, in this case, is 16.
- **Point Number 3**—Albuquerque formerly did not mention subnet 10.1.2.0 due to split-horizon rules (point 1). The update at point 3 includes a poisoned route for 10.1.2.0 with metric 16. This is an example of split horizon with poison reverse.
- **Point Number 4**—Albuquerque receives an update in S1 from Seville. The update includes a metric 16 (infinite) route to 10.1.2.0. Seville does not suspend any split-horizon rules in order to send this route because it saw the advertisement of that route earlier, so this is a simple case of route poisoning.

Example 7-6 shows the steps needed to migrate to IGRP. It also lists some **debug** and **show** commands. Example 7-6 lists the configuration added to each of the three routers shown in Figure 7-10 to migrate to IGRP. The logic of the **network** commands works just like with RIP. The output of the **show** and **debug** commands provides some insight into the differences between RIP and IGRP.

NOTE The following configuration commands would be used on all three routers.

Example 7-6 *Migration to IGRP with Sample* **show** *and* **debug** *Commands*

```
no router rip
router igrp 5
 network 10.0.0.0

Albuquerque#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 6 subnets
I       10.1.3.0 [100/8539] via 10.1.6.253, 00:00:28, Serial1
I       10.1.2.0 [100/8539] via 10.1.4.252, 00:00:18, Serial0
C       10.1.1.0 is directly connected, Ethernet0
C       10.1.6.0 is directly connected, Serial1
I       10.1.5.0 [100/10476] via 10.1.4.252, 00:00:18, Serial0
                 [100/10476] via 10.1.6.253, 00:00:29, Serial1
C       10.1.4.0 is directly connected, Serial0
Albuquerque#debug ip igrp transactions
IGRP protocol debugging is on
Albuquerque#
07:43:40: IGRP: sending update to 255.255.255.255 via Serial0 (10.1.4.251)
07:43:40:       subnet 10.1.3.0, metric=8539
07:43:40:       subnet 10.1.1.0, metric=688
07:43:40:       subnet 10.1.6.0, metric=8476
07:43:40: IGRP: sending update to 255.255.255.255 via Serial1 (10.1.6.251)
07:43:40:       subnet 10.1.2.0, metric=8539
07:43:40:       subnet 10.1.1.0, metric=688
07:43:40:       subnet 10.1.4.0, metric=8476
07:43:40: IGRP: sending update to 255.255.255.255 via Ethernet0 (10.1.1.251)
07:43:40:       subnet 10.1.3.0, metric=8539
07:43:40:       subnet 10.1.2.0, metric=8539
07:43:40:       subnet 10.1.6.0, metric=8476
07:43:40:       subnet 10.1.5.0, metric=10476
07:43:40:       subnet 10.1.4.0, metric=8476
```

continues

Example 7-6 *Migration to IGRP with Sample* **show** *and* **debug** *Commands (Continued)*

```
07:43:59: IGRP: received update from 10.1.6.253 on Serial1
07:43:59:       subnet 10.1.3.0, metric 8539 (neighbor 688)
07:43:59:       subnet 10.1.5.0, metric 10476 (neighbor 8476)
07:44:18: IGRP: received update from 10.1.4.252 on Serial0
07:44:18:       subnet 10.1.2.0, metric 8539 (neighbor 688)
07:44:18:       subnet 10.1.5.0, metric 10476 (neighbor 8476)
Albuquerque#no debug all
All possible debugging has been turned off
Albuquerque#
Albuquerque#debug ip igrp events
IGRP event debugging is on
Albuquerque#
07:45:00: IGRP: sending update to 255.255.255.255 via Serial0 (10.1.4.251)
07:45:00: IGRP: Update contains 3 interior, 0 system, and 0 exterior routes.
07:45:00: IGRP: Total routes in update: 3
07:45:00: IGRP: sending update to 255.255.255.255 via Serial1 (10.1.6.251)
07:45:00: IGRP: Update contains 3 interior, 0 system, and 0 exterior routes.
07:45:00: IGRP: Total routes in update: 3
07:45:00: IGRP: sending update to 255.255.255.255 via Ethernet0 (10.1.1.251)
07:45:01: IGRP: Update contains 5 interior, 0 system, and 0 exterior routes.
07:45:01: IGRP: Total routes in update: 5
07:45:21: IGRP: received update from 10.1.6.253 on Serial1
07:45:21: IGRP: Update contains 2 interior, 0 system, and 0 exterior routes.
07:45:21: IGRP: Total routes in update: 2
07:45:35: IGRP: received update from 10.1.4.252 on Serial0
07:45:35: IGRP: Update contains 2 interior, 0 system, and 0 exterior routes.
07:45:35: IGRP: Total routes in update: 2
Albuquerque#no debug all
All possible debugging has been turned off
Albuquerque#show ip protocol
Routing Protocol is "igrp 5"
  Sending updates every 90 seconds, next due in 34 seconds
  Invalid after 270 seconds, hold down 280, flushed after 630
  Outgoing update filter list for all interfaces is
  Incoming update filter list for all interfaces is
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  IGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  IGRP maximum hopcount 100
  IGRP maximum metric variance 1
  Redistributing: igrp 5
  Maximum path: 4
  Routing for Networks:
    10.0.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.1.6.253           100      00:00:23
    10.1.4.252           100      00:00:08
  Distance: (default is 100)
```

You can migrate from RIP to IGRP in this case with only three configuration commands per router. As highlighted in Example 7-6, the **no router rip** command removes all RIP configuration on the router, including any **network** subcommands. The three routers each must use the same IGRP process-id (5, in this case), and because all interfaces on each of the routers are in network 10.0.0.0, only a single **network 10.0.0.0** subcommand is needed.

The **show ip route** command provides the most direct view into what a routing protocol does. First, the legend at the beginning of the **show ip route** command in Example 7-6 defines the letter codes that identify the source of the routing information—for example, **C** for connected routes, **R** for RIP, and **I** for IGRP. Each of the Class A, B, and C networks is listed, along with each of the subnets of that network. If a static mask is used within that network, the mask is shown only in the line referring to the network. Each routing entry lists the subnet number and the outgoing interface. In most cases, the next-hop router's IP address is also listed.

IGRP learns the same routes that RIP learned, but using different metrics. The output of the **show ip route** command lists six subnets, just as it did when RIP was used. Also, notice the two routes to 10.1.5.0/24—one through Yosemite and one through Seville. Both routes are included in the routing table, because the default setting for **ip maximum-paths** is 4, and because the routes have an equal metric. Looking further into the output of the **debug ip igrp transactions** command, you can see the equal-cost routes being advertised. One route is seen in the update received on Serial1; the other route in the update is received on Serial0.

The output of the **debug ip igrp transactions** command shows the details of the routing updates, whereas the **debug ip igrp events** command simply mentions that routing updates have been received.

Finally, the **show ip protocol** command lists several important details about the routing protocol. The update timer is listed, shown with the time remaining until the next routing update is to be sent. Also, the elapsed time since an update was received from each neighboring router is listed at the end of the output. This command also lists each of the neighbors from which routing information has been received. If you are in doubt as to whether updates have been received during the recent past and from what routers, the **show ip protocol** command is the place to find out.

Advanced RIP and IGRP Configuration

Start Extra Credit

The next several sections describe some of the more advanced features of these two protocols. These topics might be on the exam. However, if you are trying to focus on the most important details, taking a small risk by not covering some of the less-important details, you might want to skip these sections and move to the section, "Troubleshooting Routing and Routing Protocols."

RIP-1 and IGRP: No Subnet Masks

RIP-1 and IGRP do not transmit the subnet mask in the routing updates, as seen in the **debug** output in earlier examples. Cisco expects you to be able to explain why routing protocols that do not transmit a mask can have problems in some networks. This section explains these problems, which all originate from the same root cause.

NOTE Routers must assume the subnet mask that should be used with a subnet number listed in a routing update.

Routing protocols that do not transmit masks, such as RIP and IGRP, behave predictably:

- Updates sent out an interface in network X, when containing routes about subnets of network X, contain the subnet numbers of the subnets of network X but not the corresponding masks.
- Updates sent out an interface in network X, when containing routes about subnets of network Y, contain one route about the entire network Y, but not any routes about subnets of network Y.
- When receiving a routing update containing routes referencing subnets of network X, the receiving router assumes that the mask in use is the same mask it uses on an interface with an address in network X.
- When receiving an update about network X, if the receiving router has no interfaces in network X, it treats the route as a route to the entire Class A, B, or C network X.

Examples 7-7, 7-8, and 7-9 contain **show** and **debug** command output on Albuquerque, Yosemite, and Seville with the effects described in the preceding list. The network of Figure 7-10 is still in use, but the subnet on Seville's Ethernet has been changed from 10.1.3.0/24 to 10.1.3.192/26. Because RIP-1 does not send the mask in the update, Seville chooses *not* to address 10.1.3.192/26 onto its serial links (which use mask 255.255.255.0), because the update would be ambiguous.

Example 7-7 *Configuration and* **debug ip rip** *Output on Albuquerque*

```
interface ethernet 0
ip addr 10.1.1.251 255.255.255.0
interface serial 0
ip addr 10.1.4.251 255.255.255.0
interface serial 1
ip addr 10.1.6.251 255.255.255.0
!
router rip
network 10.0.0.0

Albuquerque#debug ip rip
RIP protocol debugging is on
```

Example 7-7 *Configuration and* **debug ip rip** *Output on Albuquerque (Continued)*

```
Albuquerque#
00:38:23: RIP: received v1 update from 10.1.4.252 on Serial0
00:38:23:      10.1.2.0 in 1 hops
00:38:23:      10.1.5.0 in 1 hops
00:38:33: RIP: sending v1 update to 255.255.255.255 via Serial0 (10.1.4.251)
00:38:33:      subnet  10.1.1.0, metric 1
00:38:33:      subnet  10.1.6.0, metric 1
00:38:33: RIP: sending v1 update to 255.255.255.255 via Serial1 (10.1.6.251)
00:38:33:      subnet  10.1.2.0, metric 2
00:38:33:      subnet  10.1.1.0, metric 1
00:38:33:      subnet  10.1.4.0, metric 1
00:38:33: RIP: sending v1 update to 255.255.255.255 via Ethernet0 (10.1.1.251)
00:38:33:      subnet  10.1.2.0, metric 2
00:38:33:      subnet  10.1.6.0, metric 1
00:38:33:      subnet  10.1.5.0, metric 2
00:38:33:      subnet  10.1.4.0, metric 1
00:38:40: RIP: received v1 update from 10.1.6.253 on Serial1
00:38:40:      10.1.2.0 in 2 hops
00:38:40:      10.1.5.0 in 1 hops
undebug all
All possible debugging has been turned off
Albuquerque#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 5 subnets
R       10.1.2.0 [120/1] via 10.1.4.252, 00:00:26, Serial0
C       10.1.1.0 is directly connected, Ethernet0
C       10.1.6.0 is directly connected, Serial1
R       10.1.5.0 [120/1] via 10.1.4.252, 00:00:27, Serial0
                 [120/1] via 10.1.6.253, 00:00:10, Serial1
C       10.1.4.0 is directly connected, Serial0
Albuquerque#
(Suspended telnet resumed to Seville....)

Seville#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set
```

continues

Example 7-7 *Configuration and* **debug ip rip** *Output on Albuquerque (Continued)*

```
     10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R       10.1.2.0/24 [120/1] via 10.1.5.252, 00:00:19, Serial1
R       10.1.1.0/24 [120/1] via 10.1.6.251, 00:00:22, Serial0
C       10.1.6.0/24 is directly connected, Serial0
C       10.1.5.0/24 is directly connected, Serial1
R       10.1.4.0/24 [120/1] via 10.1.6.251, 00:00:22, Serial0
                    [120/1] via 10.1.5.252, 00:00:19, Serial1
C       10.1.3.192/26 is directly connected, Ethernet0
Seville#
```

Example 7-8 *Configuration on Yosemite*

```
interface ethernet 0
ip addr 10.1.2.252 255.255.255.0
interface serial 0
ip addr 10.1.4.252 255.255.255.0
interface serial 1
ip address 10.1.2.252 255.255.255.0

router rip
network 10.0.0.0
```

Example 7-9 *Configuration on Seville*

```
interface ethernet 0
ip addr 10.1.3.253  255.255.255.192
interface serial 0
ip addr 10.1.6.253  255.255.255.0
interface serial 1
ip address 10.1.5.253  255.255.255.0
!
router rip
network 10.0.0.0
```

As shown in the highlighted portions of Example 7-7, subnet 10.1.3.192/26 is not advertised by Seville, as seen in its update received into Albuquerque's Serial1 interface. Essentially, *RIP does not advertise the route with a mask of 255.255.255.192 out an interface that is in the same network but that has a different mask*. If RIP on Seville had advertised the route to 10.1.3.192, Albuquerque and Yosemite would have believed there was a problem, because the subnet number is 10.1.3.192, which is not a subnet number with the mask that Albuquerque and Yosemite think is in use (255.255.255.0). So, RIP and IGRP simply do not advertise the route into the same network on an interface that uses a different mask. The use of different masks in parts of the same network is called *variable-length subnet masking* (VLSM). As shown in this example, VLSM is not supported by RIP-1 or IGRP.

RIP Version 2

RIP-2, defined by RFC 1723, adds advanced features to RIP-1. Many features are the same: Hop count is still used for the metric, it is still a distance vector protocol, and it still uses hold-down timers and route poisoning. Several features have been added. They are listed in Table 7-16.

Table 7-16 *RIP-2 Features*

Feature	Description
Transmits a subnet mask with the route	This feature allows VLSM by passing the mask along with each route so that the subnet is exactly defined.
Provides authentication	Both clear text (RFC-defined) and MD5 encryption (a Cisco-added feature) can be used to authenticate the source of a routing update.
Includes a next-hop router IP address in its routing update	A router can advertise a route but direct any listeners to a different router on that same subnet. This is done only when the other router has a better route.
Uses external route tags	RIP can pass information about routes learned from an external source and redistributed into RIP.
Provides multicast routing updates	Instead of sending updates to 255.255.255.255, the destination IP address is 224.0.0.9, an IP multicast address. This reduces the amount of processing required on non-RIP-speaking hosts on a common subnet.

RIP-2 supports the new features listed in the table, but most importantly, it supports the use of VLSM through transmitting mask information. For instance, the preceding example showed a problem using VLSM on Seville (subnet 10.1.3.192/26). RIP-2 works fine in the same network, as shown in Example 7-10. This example lists the RIP-2 configuration on each of the three routers, and Example 7-11 shows a sample RIP **debug** on Albuquerque.

Example 7-10 *RIP-2 Sample Configuration for the Routers Shown in Figure 7-10*

```
router rip
network 10.0.0.0
version 2
```

Example 7-11 *RIP-2 Routing Updates, Without Autosummarization, on Albuquerque*

```
Albuquerque#debug ip rip
RIP protocol debugging is on
Albuquerque#
```

continues

Example 7-11 *RIP-2 Routing Updates, Without Autosummarization, on Albuquerque (Continued)*

```
00:36:04: RIP: received v2 update from 10.1.4.252 on Serial0
00:36:04:      10.1.2.0/24 -> 0.0.0.0 in 1 hops
00:36:04:      10.1.5.0/24 -> 0.0.0.0 in 1 hops
00:36:04:      10.1.3.192/26 -> 0.0.0.0 in 2 hops
00:36:08: RIP: sending v2 update to 224.0.0.9 via Serial0 (10.1.4.251)
00:36:08:      10.1.1.0/24 -> 0.0.0.0, metric 1, tag 0
00:36:08:      10.1.6.0/24 -> 0.0.0.0, metric 1, tag 0
00:36:08:      10.1.3.192/26 -> 0.0.0.0, metric 2, tag 0
00:36:08: RIP: sending v2 update to 224.0.0.9 via Serial1 (10.1.6.251)
00:36:08:      10.1.2.0/24 -> 0.0.0.0, metric 2, tag 0
00:36:08:      10.1.1.0/24 -> 0.0.0.0, metric 1, tag 0
00:36:08:      10.1.4.0/24 -> 0.0.0.0, metric 1, tag 0
00:36:08: RIP: sending v2 update to 224.0.0.9 via Ethernet0 (10.1.1.251)
00:36:08:      10.1.2.0/24 -> 0.0.0.0, metric 2, tag 0
00:36:08:      10.1.6.0/24 -> 0.0.0.0, metric 1, tag 0
00:36:08:      10.1.5.0/24 -> 0.0.0.0, metric 2, tag 0
00:36:08:      10.1.4.0/24 -> 0.0.0.0, metric 1, tag 0
00:36:08:      10.1.3.192/26 -> 0.0.0.0, metric 2, tag 0
00:36:20: RIP: received v2 update from 10.1.6.253 on Serial1
00:36:20:      10.1.2.0/24 -> 0.0.0.0 in 2 hops
00:36:20:      10.1.5.0/24 -> 0.0.0.0 in 1 hops
00:36:20:      10.1.3.192/26 -> 0.0.0.0 in 1 hops
00:36:30: RIP: received v2 update from 10.1.4.252 on Serial0
00:36:30:      10.1.2.0/24 -> 0.0.0.0 in 1 hops
00:36:30:      10.1.5.0/24 -> 0.0.0.0 in 1 hops
00:36:30:      10.1.3.192/26 -> 0.0.0.0 in 2 hops

Albuquerque#no debug all
All possible debugging has been turned off

Albuquerque#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 6 subnets, 2 masks
R       10.1.2.0/24 [120/1] via 10.1.4.252, 00:00:09, Serial0
C       10.1.1.0/24 is directly connected, Ethernet0
C       10.1.6.0/24 is directly connected, Serial1
R       10.1.5.0/24 [120/1] via 10.1.4.252, 00:00:09, Serial0
                    [120/1] via 10.1.6.253, 00:00:19, Serial1
C       10.1.4.0/24 is directly connected, Serial0
R       10.1.3.192/26 [120/1] via 10.1.6.253, 00:00:19, Serial1
Albuquerque#
```

A couple of important items should be noted in the **debug** output of Example 7-11. (As always, the specific portions that are referred to are highlighted.) The updates sent by Albuquerque are sent to multicast IP address 224.0.0.9 as opposed to a broadcast address; this allows the devices that are not using RIP-2 to ignore the updates and not waste processing cycles. The **show ip route** output on Albuquerque lists the previously missing subnet, 10.1.3.192/26; this is expected, as highlighted in the **debug ip rip** messages received by Albuquerque from Seville (10.1.6.253). The subnet masks are shown in the prefix style, with /26 representing mask 255.255.255.192. Also, note the **debug** output designating **tag 0**. This means that all the external route tags have value 0, which is the default.

Migrating from RIP-1 to RIP-2 requires some planning. RIP-1 sends updates to the broadcast address, whereas RIP-2 uses a multicast. A RIP-1-only router and a RIP-2-only router will not succeed in exchanging routing information. To migrate to RIP-2, one option is to migrate all routers at the same time. This might not be a reasonable political or administrative option, however. If not, some coexistence between RIP-1 and RIP-2 is required.

The **ip rip send version** command can be used to overcome this problem. Essentially, the configuration tells the router whether to send RIP-1-style updates, RIP-2-style updates, or both for each interface. Consider the familiar Figure 7-10 network, with RIP-1 still configured on all three routers. If two of the routers are migrated—for instance, Albuquerque and Seville—they can communicate with RIP-2 easily. However, by default these two routers now send only RIP-2 updates, which Yosemite cannot understand, because it is still running RIP-1. The configurations shown in Examples 7-12, 7-13, and 7-14 overcome this problem by having Albuquerque and Seville send only RIP-1 updates to Yosemite.

Example 7-12 *Configuration on Albuquerque*

```
interface ethernet 0
ip addr 10.1.1.251 255.255.255.0
interface serial 0
ip addr 10.1.4.251 255.255.255.0
ip rip send version 1
ip rip receive version 1
interface serial 1
ip address 10.1.6.251 255.255.255.0
!
router rip
network 10.0.0.0
version 2
```

Example 7-13 *Configuration on Yosemite*

```
interface ethernet 0
ip addr 10.1.2.252 255.255.255.0
interface serial 0
ip addr 10.1.4.252 255.255.255.0
```

continues

Example 7-13 *Configuration on Yosemite (Continued)*

```
interface serial 1
ip address 10.1.5.252 255.255.255.0
!
router rip
network 10.0.0.0
```

Example 7-14 *Configuration on Seville*

```
interface ethernet 0
ip addr 10.1.2.252 255.255.255.0
interface serial 0
ip addr 10.1.4.252 255.255.255.0
interface serial 1
ip address 10.1.5.252 255.255.255.0
ip rip send version 1
ip rip receive version 1
!
router rip
network 10.0.0.0
version 2
```

The RIP-2 configuration logic works just like RIP-1. Updates are sent and received on each interface that is matched by a **network** command. But because Yosemite sends and receives only RIP-1 updates, the other two routers need the appropriate interface subcommands to tell the router to send and receive RIP-1 updates to and from Yosemite. Both Albuquerque and Seville continue to send and receive RIP-2 updates on all interfaces, so when Yosemite upgrades to RIP-2, no immediate configuration changes are required in Albuquerque and Seville.

Autosummarization and Route Aggregation

Cisco IOS Software is optimized to perform routing as fast as possible. Most of the Layer 3 routing performance improvement in the brief history of routers has been through improved algorithms. Many times those improved algorithms later have been implemented in hardware to provide even lower latency. Although these improvements have been a great benefit, it is typically true that any algorithm that searches a list runs more quickly if the list is short, as compared to searching a similar list that is long.

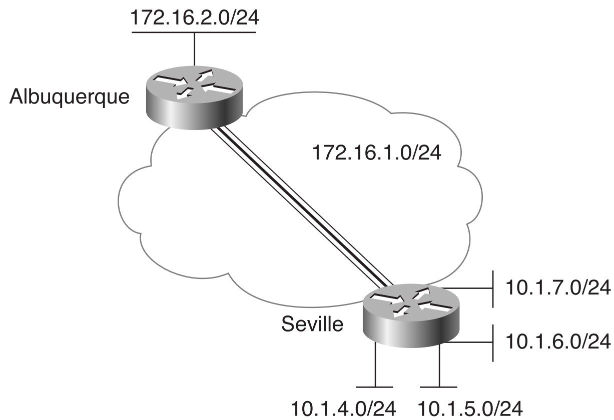
Autosummarization and route aggregation (also known as route summarization) are two Cisco IOS Software features that reduce the size of the IP routing table, thereby reducing latency per packet. Autosummarization is a routing protocol feature that operates using this rule:

> When advertised on an interface whose IP address is not in network X, routes about subnets in network X are summarized and advertised as one route. That route is for the entire Class A, B, or C network X.

RIP and IGRP perform autosummarization, and it cannot be disabled. Essentially, it must be a side effect of routing protocols that transmit the mask. For RIP-2 and EIGRP, autosummarization can be enabled or disabled.

As usual, an example makes the concept much clearer. Consider Figure 7-11, which shows two networks in use: 10.0.0.0 and 172.16.0.0. Seville has four (connected) routes to subnets of network 10.0.0.0. Example 7-15 shows the output of a **show ip route** command on Albuquerque, as well as RIP-2 **debug ip rip** output.

Figure 7-11 *Autosummarization*



Example 7-15 *Configuration on Seville*

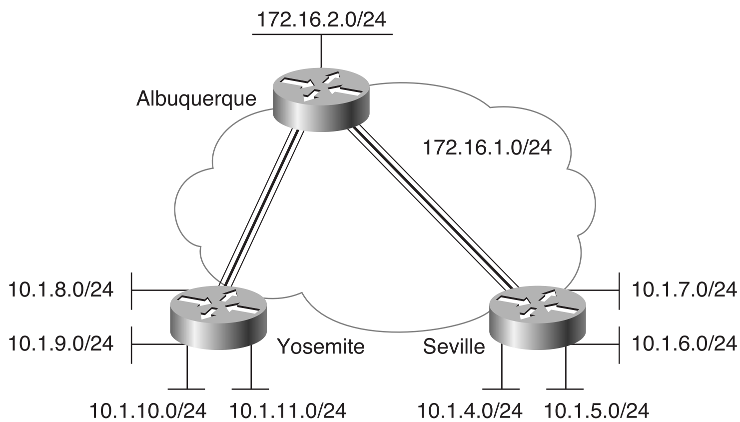
```
Albuquerque#debug ip rip
02:20:42: RIP: sending v2 update to 224.0.0.9 via Serial0.2 (172.16.1.251)
02:20:42:      172.16.2.0/24 -> 0.0.0.0, metric 1, tag 0
02:20:42: RIP: sending v2 update to 224.0.0.9 via Ethernet0 (172.16.2.251)
02:20:42:      172.16.1.0/24 -> 0.0.0.0, metric 1, tag 0
02:20:42:      10.0.0.0/8 -> 0.0.0.0, metric 2, tag 0
02:20:46: RIP: received v2 update from 172.16.1.253 on Serial0.2
02:20:46:      10.0.0.0/8 -> 0.0.0.0 in 1 hops
Albuquerque#undebug all
All possible debugging has been turned off
Albuquerque#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.1.0 is directly connected, Serial0.2
C       172.16.2.0 is directly connected, Ethernet0
R    10.0.0.0/8 [120/1] via 172.16.1.253, 00:00:09, Serial0.2
```

Notice, as highlighted in Example 7-15, that Albuquerque's received update on Serial0.2 from Seville advertises only the entire Class A network 10.0.0.0/8 because autosummarization is enabled on Seville (by default). The IP routing table lists just one route to network 10.0.0.0. This works fine, as long as network 10.0.0.0 is contiguous. Consider Figure 7-12, in which Yosemite also has subnets of network 10.0.0.0 but has no connectivity to Seville other than through Albuquerque.

Figure 7-12 *Autosummarization Pitfalls*



IP subnet design traditionally has not allowed *discontiguous networks*. A *contiguous network* is a single Class A, B, or C network for which all routes to subnets of that network pass through only other subnets of that same single network. *Discontiguous networks* refers to the concept that, in a single Class A, B, or C network, there is at least one case in which the only routes to one subnet pass through subnets of a different network. An easy analogy for residents of the United States is the term *contiguous 48,* referring to the 48 states besides Alaska and Hawaii. To drive to Alaska from the contiguous 48, for example, you must drive through another country (Canada, for the geographically impaired!), so Alaska is not contiguous with the 48 states—in other words, it is discontiguous.

Figure 7-12 breaks that rule. In this figure, there could be a PVC between Yosemite and Seville that uses a subnet of network 10.0.0.0, but that PVC might be down, causing the discontiguous network. The temporarily discontiguous network can be overcome with the use of a routing protocol that transmits masks, because the rule of discontiguous subnets can be ignored if no autosummarization, or configured summarization, is performed. Consider the routing updates and routing table on Albuquerque in Example 7-16, where RIP-2 is used so that autosummarization is disabled on all routers.

Example 7-16 *Albuquerque's Routing Table When Seville Is Not Summarizing*

```
Albuquerque#debug ip rip
RIP protocol debugging is on
Albuquerque#
02:48:58: RIP: received v2 update from 172.16.1.253 on Serial0.2
02:48:58:      10.1.7.0/24 -> 0.0.0.0 in 1 hops
02:48:58:      10.1.6.0/24 -> 0.0.0.0 in 1 hops
02:48:58:      10.1.5.0/24 -> 0.0.0.0 in 1 hops
02:48:58:      10.1.4.0/24 -> 0.0.0.0 in 1 hops
02:49:14: RIP: received v2 update from 172.16.3.252 on Serial0.1
02:49:14:      10.1.11.0/24 -> 0.0.0.0 in 1 hops
02:49:14:      10.1.10.0/24 -> 0.0.0.0 in 1 hops
02:49:14:      10.1.9.0/24 -> 0.0.0.0 in 1 hops
02:49:14:      10.1.8.0/24 -> 0.0.0.0 in 1 hops
02:49:16: RIP: sending v2 update to 224.0.0.9 via Serial0.1 (172.16.3.251)
02:49:16:      172.16.1.0/24 -> 0.0.0.0, metric 1, tag 0
02:49:16:      172.16.2.0/24 -> 0.0.0.0, metric 1, tag 0
02:49:16:      10.0.0.0/8 -> 0.0.0.0, metric 2, tag 0
02:49:16: RIP: sending v2 update to 224.0.0.9 via Serial0.2 (172.16.1.251)
02:49:16:      172.16.2.0/24 -> 0.0.0.0, metric 1, tag 0
02:49:16:      172.16.3.0/24 -> 0.0.0.0, metric 1, tag 0
02:49:16:      10.0.0.0/8 -> 0.0.0.0, metric 2, tag 0
02:49:16: RIP: sending v2 update to 224.0.0.9 via Ethernet 0 (172.16.2.251)
02:49:16:      172.16.1.0/24 -> 0.0.0.0, metric 1, tag 0
02:49:16:      172.16.3.0/24 -> 0.0.0.0, metric 1, tag 0
02:49:16:      10.0.0.0/8 -> 0.0.0.0, metric 2, tag 0
Albuquerque#no debug all
All possible debugging has been turned off
Albuquerque#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, Serial0.2
C       172.16.2.0 is directly connected, Ethernet0
C       172.16.3.0 is directly connected, Serial0.1
     10.0.0.0/24 is subnetted, 8 subnets
R       10.1.11.0 [120/1] via 172.16.3.252, 00:00:15, Serial0.1
R       10.1.10.0 [120/1] via 172.16.3.252, 00:00:15, Serial0.1
R       10.1.9.0 [120/1] via 172.16.3.252, 00:00:15, Serial0.1
R       10.1.8.0 [120/1] via 172.16.3.252, 00:00:15, Serial0.1
R       10.1.7.0 [120/1] via 172.16.1.253, 00:00:03, Serial0.2
R       10.1.6.0 [120/1] via 172.16.1.253, 00:00:03, Serial0.2
R       10.1.5.0 [120/1] via 172.16.1.253, 00:00:03, Serial0.2
R       10.1.4.0 [120/1] via 172.16.1.253, 00:00:03, Serial0.2
Albuquerque#
```

As highlighted in Example 7-16, the routing updates include the individual subnets. Therefore, Albuquerque can see routes to all subnets of network 10 and can route packets to the correct destinations in Seville and Yosemite. With autosummarization enabled, Albuquerque would think that both Seville and Yosemite had an equal-metric route to network 10.0.0.0, and some packets would be routed incorrectly.

Route summarization (also called route aggregation) works like autosummarization, except that there is no requirement to summarize into a Class A, B, or C network. Consider the network shown in Figure 7-12. Albuquerque has eight routes to subnets of network 10.0.0.0; four of those routes are learned from Seville. Consider the subnet, broadcast, and assignable addresses in each of the subnets, as shown in Table 7-17.

Table 7-17 *Route Aggregation Comparison of Subnet Numbers*

Subnet	Mask	Broadcast	Assignable Addresses
10.1.4.0	255.255.255.0	10.1.4.255	10.1.4.1 to 10.1.4.254
10.1.5.0	255.255.255.0	10.1.5.255	10.1.5.1 to 10.1.5.254
10.1.6.0	255.255.255.0	10.1.6.255	10.1.6.1 to 10.1.6.254
10.1.7.0	255.255.255.0	10.1.7.255	10.1.7.1 to 10.1.7.254

Now consider the concept of a subnet 10.1.4.0 with mask 255.255.252.0. In this case, 10.1.4.0/22 (the same subnet written differently) has a subnet broadcast address of 10.1.7.255 and assignable addresses of 10.1.4.1 to 10.1.7.254. Because 10.1.4.0/22 includes all the assignable addresses of the original four subnets, a single route to 10.1.4.0/22 is just as good as the four separate routes, assuming that the next-hop information is the same for each of the original four routes.

Route aggregation is simply a tool used to tell a routing protocol to advertise a single, larger subnet rather than individual, smaller subnets. In this case, the routing protocol advertises 10.1.4.0/22 rather than the four individual subnets. Albuquerque's routing table is then smaller. EIGRP and OSPF are the only interior IP routing protocols that support route aggregation.

Example 7-17 shows route summarization of the subnets off Seville. Still using the network shown in Figure 7-12, the routers are all migrated to EIGRP. Example 7-17 shows the EIGRP configuration on Albuquerque, the EIGRP configuration on Seville, and the resulting IP routing table on Albuquerque. (Yosemite is migrated to EIGRP as well; the configuration is not shown because the example shows only aggregation by Seville.)

Example 7-17 *Route Aggregation Example Using EIGRP*

```
On Seville:
router eigrp 9
 Network 10.0.0.0
```

Example 7-17 *Route Aggregation Example Using EIGRP (Continued)*

```
 Network 172.16.0.0
 No auto-summary
!
interface serial 0.1 point-to-point
 ip address 172.16.1.253 255.255.255.0
 frame-relay interface-dlci 901
 ip summary-address eigrp 9 10.1.4.0 255.255.252.0
```

```
On Albuquerque:
router eigrp 9
  network 172.16.0.0
  no auto-summary
Albuquerque#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, Serial0.2
C       172.16.2.0 is directly connected, Ethenet0
C       172.16.3.0 is directly connected, Serial0.1
     10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
D       10.1.11.0/24 [120/1] via 172.16.3.252, 00:00:15, Serial0.1
D       10.1.10.0/24 [120/1] via 172.16.3.252, 00:00:15, Serial0.1
D       10.1.9.0/24 [120/1] via 172.16.3.252, 00:00:15, Serial0.1
D       10.1.8.0/24 [120/1] via 172.16.3.252, 00:00:15, Serial0.1
D       10.1.4.0/22 [90/2185984] via 172.16.1.253, 00:00:58, Serial0.2
```

The **ip summary-address** interface subcommand on Seville's Serial0.1 interface is used to define the superset of the subnets that should be advertised. Notice the route in Albuquerque's routing table, which indeed shows 10.1.4.0/22 rather than the four individual subnets.

When summarizing, the superset of the original subnets can actually be smaller than the Class A, B, or C network; larger than the network; or exactly matched to a network. For instance, 192.168.4.0, 192.168.5.0, 192.168.6.0, and 192.168.7.0 can be summarized into 192.168.4.0/22, which represents four consecutive Class C networks. Summarizing when the summarized group is a set of networks is sometimes called *supernetting*.

Table 7-18 lists the features for summarizing the interior IP routing protocols.

Table 7-18 *Interior IP Routing Protocol Summarization Features*

Routing Protocol	Autosummarization Enabled?	Can Autosummarization Be Disabled?	Route Aggregation/ Summarization Allowed?
RIP-1	Yes, by default	No	No
IGRP	Yes, by default	No	No
RIP-2	Yes, by default	Yes	No
Enhanced IGRP	Yes, by default	Yes	Yes
OSPF	No, but the equivalent can be done with aggregation	N/A	Yes

Multiple Routes to the Same Subnet

What if a router learns two routes whose metrics are equal? By default, the Cisco IOS Software supports four equal-cost routes to the same IP subnet in the routing table at the same time. The traffic is balanced across the equal-metric routes on a per-destination address basis by default. You can change the number of equal-cost routes to between 1 and 6 using the **ip maximum-paths** *x* router configuration subcommand, where *x* is the maximum number of routes to any subnet.

The metric formula used for IGRP (and EIGRP) poses an interesting problem when considering equal-metric routes. IGRP can learn more than one route to the same subnet with different metrics; however, the metrics are very likely to never be exactly equal. The **variance** router subcommand is used to define how variable the metrics can be in order for routes to be considered to have equal metrics. For example, if the metric for the better of two routes is 100, and the variance is set to 2, a second route with a metric less than 200 would be considered equal-cost and would be added to the routing table.

For many years, equal-cost routes were treated equally—which seems to make sense. But what if the routes are not really equal and you use variance to add them? Well, with one more router subcommand, you can make the router either always use the truly best route or balance across the routes based on the metrics' ratios. In other words, the **traffic-share min** router IGRP subcommand tells the router to ignore all equal-metric routes in the routing table, except the route that truly has the smallest metric. So why not just add only the truly lowest metric route to the routing table? Well, if the other pretty good routes are in the table, and the best one fails, convergence time is practically instantaneous!

An alternative to using the single, truly best route, even when multiple routes are in the routing table, is to use the **traffic-share balanced** router subcommand. It tells the router to use all the routes proportionally based on the metrics for each route.

End Extra Credit

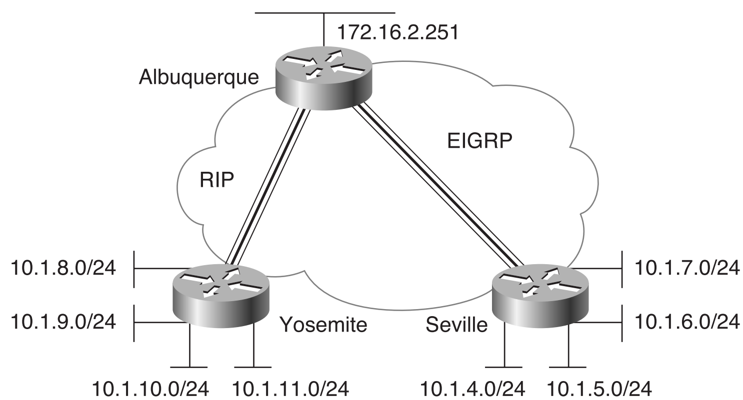
Troubleshooting Routing and Routing Protocols

Cisco would like all its certification exams—CCNA included—to prove that the test taker can build and troubleshoot live networks. Some people work with Cisco routers daily. Others' job function does not allow frequent access to routers. If the latter description applies to you, you might be trying to pass this certification so that you can move into jobs that involve routers and switches. Regardless, this section gives you some final insights into some tricky problems with routing protocols.

The **show ip route** command has a myriad of options that are helpful when you're troubleshooting a large network. The **show ip protocol** command also can provide some useful information when you're troubleshooting a routing problem. With a small network, most of the options with the **show ip route** command are unnecessary. However, knowing the options and what each can do is useful for your work with larger networks.

Example 6-21 lists the options of the **show ip route** command and gives examples of several of them. Figure 7-13 shows the network; it should look familiar from previous examples. In this case, EIGRP is used between Albuquerque and Seville, and RIP-2 is used between Albuquerque and Yosemite. There is no PVC between Yosemite and Seville. The configurations of the three routers are listed in Examples 7-18, 7-19, and 7-20. Example 7-21 lists the **show ip route** options.

Figure 7-13 *Network Environment for Use with the* **show ip route** *Options*



Example 7-18 *Albuquerque Configuration for the* **show ip route** *Options in Example 7-21*

```
Albuquerque#show running-config
Current configuration : 964 bytes
!
version 12.2
```

continues

Example 7-18 *Albuquerque Configuration for the* **show ip route** *Options in Example 7-21 (Continued)*

```
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Albuquerque
!
enable secret 5 $1$J3Fz$QaEYNIiI2aMu.3Ar.q0Xm.
enable password fred
!
ip subnet-zero
no ip domain-lookup
!
interface Serial0
 no ip address
 no ip directed-broadcast
 encapsulation frame-relay IETF
 clockrate 56000
 frame-relay lmi-type cisco
!
interface Serial0.1 point-to-point
 ip address 172.16.3.251 255.255.255.0
 no ip directed-broadcast
 frame-relay interface-dlci 902
!
interface Serial0.2 point-to-point
 ip address 172.16.1.251 255.255.255.0
 no ip directed-broadcast
 frame-relay interface-dlci 903
!
interface Serial1
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Ethernet0
 ip address 172.16.2.251 255.255.255.0
 no ip directed-broadcast
!
router eigrp 9
 passive-interface Serial0.1
 network 172.16.0.0
 no auto-summary
!
router rip
 version 2
 passive-interface Serial0.2
 network 172.16.0.0
 no auto-summary
!
ip classless
no ip http server
!
access-list 1 permit 10.0.0.0 0.255.255.255
```

Example 7-19 *Yosemite Configuration for the* **show ip route** *Options in Example 7-21*

```
Yosemite#show running-config
Current configuration : 968 bytes
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Yosemite
!
enable secret 5 $1$J3Fz$QaEYNIiI2aMu.3Ar.q0Xm.
!
ip subnet-zero
no ip domain-lookup
!
interface Serial0
 no ip address
 no ip directed-broadcast
 encapsulation frame-relay IETF
 no fair-queue
 frame-relay lmi-type cisco
!
interface Serial0.1 point-to-point
 ip address 172.16.3.252 255.255.255.0
 no ip directed-broadcast
 frame-relay interface-dlci 901
!
interface Serial1
 no ip address
 no ip directed-broadcast
 shutdown
!
!
interface Ethernet0
 ip address 10.1.8.253 255.255.255.0
!
interface Ethernet1
 ip address 10.1.9.253 255.255.255.0
!
interface Ethernet2
 ip address 10.1.10.253 255.255.255.0
!
interface Ethernet3
 ip address 10.1.11.253 255.255.255.0
!
router rip
 version 2
 network 10.0.0.0
 network 172.16.0.0
 no auto-summary
!
ip classless
no ip http server
```

Example 7-20 *Seville Configuration for the* **show ip route** *Options in Example 7-21*

```
Seville#show running-config
Current configuration : 960 bytes
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Seville
!
enable secret 5 $1$J3Fz$QaEYNIiI2aMu.3Ar.q0Xm.
!
ip subnet-zero
no ip domain-lookup
!
interface Serial0
 no ip address
 no ip directed-broadcast
 encapsulation frame-relay IETF
 no fair-queue
 frame-relay lmi-type cisco
!
interface Serial0.1 multipoint
 ip address 172.16.1.253 255.255.255.0
 no ip directed-broadcast
 ip summary-address eigrp 9 10.1.4.0 255.255.252.0
 frame-relay interface-dlci 901
!
interface Serial1
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Ethernet0
 ip address 10.1.4.253 255.255.255.0
!
interface Ethernet1
 ip address 10.1.5.253 255.255.255.0
!
interface Ethernet2
 ip address 10.1.6.253 255.255.255.0
!
interface Ethernet3
 ip address 10.1.7.253 255.255.255.0
!
router eigrp 9
 network 10.0.0.0
 network 172.16.0.0
 no auto-summary
!
ip classless
no ip http server
```

Example 7-21 show ip route: *Albuquerque*

```
Albuquerque#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, Serial0.2
C       172.16.2.0 is directly connected, Ethernet0
C       172.16.3.0 is directly connected, Serial0.1
     10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R       10.1.11.0/24 [120/1] via 172.16.3.252, 00:00:17, Serial0.1
R       10.1.10.0/24 [120/1] via 172.16.3.252, 00:00:17, Serial0.1
R       10.1.9.0/24 [120/1] via 172.16.3.252, 00:00:17, Serial0.1
R       10.1.8.0/24 [120/1] via 172.16.3.252, 00:00:17, Serial0.1
D       10.1.4.0/22 [90/2185984] via 172.16.1.253, 00:28:01, Serial0.2

Albuquerque#show ip route ?
   Hostname or A.B.C.D  Network to display information about or hostname
  bgp                  Border Gateway Protocol (BGP)
  connected            Connected
  egp                  Exterior Gateway Protocol (EGP)
  eigrp                Enhanced Interior Gateway Routing Protocol (EIGRP)
  igrp                 Interior Gateway Routing Protocol (IGRP)
  isis                 ISO IS-IS
  list                 IP Access list
  mobile               Mobile routes
  odr                  On Demand stub Routes
  ospf                 Open Shortest Path First (OSPF)
  profile              IP routing table profile
  rip                  Routing Information Protocol (RIP)
  static               Static routes
  summary              Summary of all routes
  supernets-only       Show supernet entries only
  vrf                  Display routes from a VPN Routing/Forwarding instance
  |                    Output modifiers


Albuquerque#show ip route 10.1.5.8
Routing entry for 10.1.4.0/22
  Known via "eigrp 9", distance 90, metric 2185984, type internal
  Redistributing via eigrp 9
  Last update from 172.16.1.253 on Serial0.2, 00:28:36 ago
  Routing Descriptor Blocks:
  * 172.16.1.253, from 172.16.1.253, 00:28:36 ago, via Serial0.2
      Route metric is 2185984, traffic share count is 1
      Total delay is 20630 microseconds, minimum bandwidth is 1544 Kbit
```

continues

Example 7-21 show ip route*: Albuquerque (Continued)*

```
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 1

Albuquerque#show ip route rip
     10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R       10.1.11.0/24 [120/1] via 172.16.3.252, 00:00:22, Serial0.1
R       10.1.10.0/24 [120/1] via 172.16.3.252, 00:00:22, Serial0.1
R       10.1.9.0/24 [120/1] via 172.16.3.252, 00:00:22, Serial0.1
R       10.1.8.0/24 [120/1] via 172.16.3.252, 00:00:22, Serial0.1
Albuquerque#show ip route igrp

Albuquerque#show ip route eigrp
     10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
D       10.1.4.0/22 [90/2185984] via 172.16.1.253, 00:29:42, Serial0.2

Albuquerque#show ip route connected
     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, Serial0.2
C       172.16.2.0 is directly connected, Ethernet0
C       172.16.3.0 is directly connected, Serial0.1

Albuquerque#show ip route list 1
     10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
R       10.1.11.0/24 [120/1] via 172.16.3.252, 00:00:22, Serial0.1
R       10.1.10.0/24 [120/1] via 172.16.3.252, 00:00:22, Serial0.1
R       10.1.9.0/24 [120/1] via 172.16.3.252, 00:00:22, Serial0.1
R       10.1.8.0/24 [120/1] via 172.16.3.252, 00:00:22, Serial0.1
D       10.1.4.0/22 [90/2185984] via 172.16.1.253, 00:29:58, Serial0.2

Albuquerque#show ip route summary
Route Source    Networks    Subnets     Overhead    Memory (bytes)
connected       0           3           156         420
static          0           0           0           0
rip             0           4           208         560
eigrp 9         0           1           52          140
internal        2                                   2320
Total           2           8           416         3440

Albuquerque#show ip route supernet
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route, o - ODR

Gateway of last resort is not set
```

The **show ip route** command with no options has been seen many times in this book. A review of some of the more important bits of the output is in order; most comments refer to a highlighted portion. First, the legend at the beginning of Example 7-21 defines the letter codes that identify the source of the routing information—for example, **C** for connected routes, **R** for RIP, and **I** for IGRP. Each of the Class A, B, and C networks is listed, along with each of the subnets of that network. If a static mask is used within that network, the mask is shown only in the line referring to the network (as is the case in Example 7-21, network 172.16.0.0). If the network uses VLSM, as network 10.0.0.0 appears to do because of the route summarization done by Seville, the mask information is listed on the lines referring to each of the individual subnets.

Each routing entry lists the subnet number and the outgoing interface. In most cases, the next-hop router's IP address is also listed. The outgoing interface is needed so that the router can choose the type of data link header to use to encapsulate the packet before transmission on that interface. The next-hop router's IP address is needed on interfaces for which the router needs the IP address so that it can find the associated data-link address to put in the newly built data link header. For instance, knowing the next-hop IP address of 172.16.3.252, Yosemite's IP address on the Frame Relay VC allows Albuquerque to find the corresponding DLCI in the Frame Relay map.

The numbers in brackets in the **show ip route** output for each route are interesting. The second number in brackets represents the metric value for this route. The first number defines the administrative distance.

Administrative distance is important only if multiple IP routing protocols are in use in a single router. When this is true, both routing protocols can learn routes to the same subnets. Because their metric values are different (for example, hop count or a function of bandwidth and delay), there is no way to know which routing protocol's routes are better. Therefore, Cisco supplies a method of defining which routing protocol's routes are better. The Cisco IOS Software implements this concept using something called *administrative distance*.

Administrative distance is an integer value; a value is assigned to each source of routing information. The lower the administrative distance, the better the source of routing information. IGRP's default is 100, OSPF's is 110, RIP's is 120, and EIGRP's is 90. The value 100 in brackets in the **show ip route** output signifies that the administrative distance used for IGRP routes is 100. In other words, the default value is in use. So, if RIP and IGRP are both used, and if both learn routes to the same subnets, only IGRP's routing information for those subnets is added to the routing table. If RIP learns about a subnet that IGRP does not know about, that route is added to the routing table.

Moving down Example 7-21, the **show ip route ?** command lists several options, many of which are shown in the ensuing commands in the example. You can limit the **show ip route** output to the routes learned by a particular routing protocol by referring to that routing protocol. Likewise, the output can be limited to show just connected routes.

One of the more important options for the **show ip route** command is to simply pass an IP address as the last parameter. This tells the router to perform routing table lookup, just as it would for a packet destined for that address. In Example 7-21, **show ip route 10.1.5.8** returns a set of messages, the first of which identifies the route to 10.1.4.0/22 as the route matched in the routing table. The route that is matched is listed so that you can always know the route that would be used by this router to reach a particular IP address.

Finally, another feature of **show ip route** that is useful in large networks is filtering the command's output based on an access list. Notice the command **show ip route list 1** in Example 7-21. Access list 1 is configured so that any route with information about network 10.0.0.0 is matched (permitted by the access list) and all others are denied. By referring to the access list, the **show ip route** output is filtered, showing only a portion of the routes. This is particularly useful when there are many routes in the routing table.

The many options of the **show ip route** command can be particularly useful for troubleshooting larger networks.

Foundation Summary

The Foundation Summary is a collection of tables that provides a convenient review of many key concepts in this chapter. If you are already comfortable with the topics in this chapter, this summary can help you recall a few details. If you just read this chapter, this review should help solidify some key facts. If you are doing your final preparation before the exam, these tables are a convenient way to review the day before the exam.

Table 7-19 lists some of the routing protocol terms you need to know for the CCNA exam.

Table 7-19 *Routing Protocol Terminology*

Term	Definition
Routing protocol	A protocol whose purpose is to learn the available routes, place the best routes into the routing table, and remove routes when they are no longer valid.
Exterior routing protocol	A routing protocol designed for use between two different networks that are under the control of two different organizations. These are typically used between ISPs or between a company and an ISP. For instance, a company would run BGP, an exterior routing protocol, between one of its routers and a router inside an ISP.
Interior routing protocol	A routing protocol designed for use in a network whose parts are under the control of a single organization. For example, an entire company might choose the IGRP routing protocol, which is an interior routing protocol.
Distance vector	The logic behind the behavior of some Interior routing protocols, such as RIP and IGRP.
Link state	The logic behind the behavior of some interior routing protocols, such as OSPF.
Balanced hybrid	The logic behind the behavior of EIGRP, which is more like distance vector than link state but is different from these other two types of routing protocols.
Dijkstra Shortest-Path First (SPF) algorithm	Magic math used by link-state protocols, such as OSPF, when the routing table is calculated.

continues

Table 7-19 *Routing Protocol Terminology (Continued)*

Term	Definition
DUAL	The process by which EIGRP routers collectively calculate routing tables.
Convergence	The time required for routers to react to changes in the network, removing bad routes and adding new, better routes so that the currently best routes are in all the routers' routing tables.

Table 7-20 lists interior IP routing protocols and their types. A column referring to whether the routing protocol includes subnet mask information in the routing updates is listed for future reference.

Table 7-20 *Interior IP Routing Protocols and Types*

Routing Protocol	Type	Loop-Prevention Mechanisms	Mask Sent in Updates, Which Allows VLSM?
RIP-1	Distance vector	Hold-down timer, split horizon	No
RIP-2	Distance vector	Hold-down timer, split horizon	Yes
IGRP	Distance vector	Hold-down timer, split horizon	No
EIGRP	Balanced hybrid	DUAL and feasible successors	Yes
OSPF	Link-state	Dijkstra SPF algorithm and full topology knowledge	Yes

Table 7-21 summarizes distance vector routing issues and describes solutions.

Table 7-21 *Issues Relating to Distance Vector Routing Protocols in a Network with Multiple Paths*

Issue	Solution
Multiple routes to the same subnet have equal metrics	Implementation options involve either using the first route learned or putting multiple routes to the same subnet in the routing table.

Table 7-21 *Issues Relating to Distance Vector Routing Protocols in a Network with Multiple Paths (Continued)*

Issue	Solution
Routing loops occur due to updates passing each other over a single link	**Split horizon**—The routing protocol advertises routes out an interface only if they were not learned from updates entering that interface. **Split horizon with poison reverse**—The routing protocol uses split-horizon rules unless a route fails. In that case, the route is advertised out all interfaces, but with infinite-distance metrics.
Routing loops occur due to updates passing each other over alternative paths	**Route poisoning**—When a route to a subnet fails, the subnet is advertised with an infinite-distance metric.
Counting to infinity	**Hold-down timer**—After finding out that a route to a subnet has failed, a router waits a certain period of time before believing any other routing information about that subnet. **Triggered updates**—When a route fails, an update is sent immediately rather than waiting on the update timer to expire. Used in conjunction with route poisoning, this ensures that all routers know of failed routes before any hold-down timers can expire.

Table 7-22 outlines the features of RIP and IGRP.

Table 7-22 *RIP and IGRP Feature Comparison*

Feature	RIP (Default)	IGRP (Default)
Update timer	30 seconds	90 seconds
Metric	Hop count	Function of bandwidth and delay (the default). Can include reliability, load, and MTU.
Hold-down timer	180	280
Flash (triggered) updates	Yes	Yes
Mask sent in update	No for RIP-1; yes for RIP-2	No
Infinite-metric value	16	4,294,967,295

Tables 7-23 and 7-24 summarize the more-popular commands used for RIP and IGRP configuration and verification.

Table 7-23 *IP RIP and IGRP Configuration Commands*

Command	Configuration Mode
router rip	Global
router igrp *as-number*	Global
network *net-number*	Router subcommand
passive-interface [**default**] {*interface-type interface-number*}	Router subcommand
maximum-paths *number-paths*	Router subcommand
variance *multiplier*	Router subcommand
traffic-share {*balanced* | *min*}	Router subcommand

Table 7-24 *IP RIP and IGRP EXEC Commands*

Command	Function
show ip route [*ip-address* [*mask*] [**longer-prefixes**]] | [*protocol* [*process-id*]]	Shows the entire routing table, or a subset if parameters are entered.
show ip protocols	Shows routing protocol parameters and current timer values.
debug ip rip	Issues log messages for each RIP update.
debug ip igrp transactions [*ip-address*]	Issues log messages with details of the IGRP updates.
debug ip igrp events [*ip-address*]	Issues log messages for each IGRP packet.
ping [*protocol* | **tag**] {*host-name* | *system-address*}	Sends and receives ICMP echo messages to verify connectivity.
trace [*protocol*] [*destination*]	Sends a series of ICMP echoes with increasing TTL values to verify the current route to a host.

Table 7-25 summarizes RIP-2 features.

Table 7-25 *RIP-2 Features*

Feature	Description
Transmits a subnet mask with the route	This feature allows VLSM by passing the mask along with each route so that the subnet is exactly defined.
Provides authentication	Both clear text (RFC-defined) and MD5 encryption (a Cisco-added feature) can be used to authenticate the source of a routing update.
Includes a next-hop router IP address in its routing update	A router can advertise a route but direct any listeners to a different router on that same subnet. This is done only when the other router has a better route.
Uses external route tags	RIP can pass information about routes learned from an external source and redistributed into RIP.
Provides multicast routing updates	Instead of sending updates to 255.255.255.255, the destination IP address is 224.0.0.9, an IP multicast address. This reduces the amount of processing required on non-RIP-speaking hosts on a common subnet.

Table 7-26 lists the features for summarization of the interior IP routing protocols.

Table 7-26 *Interior IP Routing Protocol Summarization Features*

Routing Protocol	Autosummarization Enabled?	Can Autosummarization Be Disabled?	Route Aggregation/ Summarization Allowed?
RIP-1	Yes, by default	No	No
IGRP	Yes, by default	No	No
RIP-2	Yes, by default	Yes	No
Enhanced IGRP	Yes, by default	Yes	Yes
OSPF	No, but the equivalent can be done with aggregation	N/A	Yes

Q&A

As mentioned in Chapter 1, "All About the Cisco Certified Network Associate Certification," the questions and scenarios in this book are more difficult than what you should experience on the exam. The questions do not attempt to cover more breadth or depth than the exam, but they are designed to make sure that you know the answer. Rather than allowing you to derive the answer from clues hidden in the question, the questions challenge your understanding and recall of the subject. Questions from the "Do I Know This Already?" quiz at the beginning of this chapter are repeated here to ensure that you have mastered this chapter's topics. Hopefully these questions will help limit the number of exam questions on which you narrow your choices to two options and then guess.

The answers to these questions can be found in Appendix A.

1. What type of routing protocol algorithm uses a hold-down timer? What is its purpose?

2. Define what split horizon means to the contents of a routing update. Does this apply to both the distance vector algorithm and the link-state algorithm?

3. Write down the steps you would take to migrate from RIP to IGRP in a router whose current RIP configuration includes only **router rip** followed by a **network 10.0.0.0** command.

4. How does the Cisco IOS Software designate a subnet in the routing table as a directly connected network? What about a route learned with IGRP or RIP?

5 Create a configuration for IGRP on a router with these interfaces and addresses: e0 using 10.1.1.1, e1 using 224.1.2.3, s0 using 10.1.2.1, and s1 using 199.1.1.1. Use process ID 5

____

____

____

6 Create a configuration for IGRP on a router with these interfaces and addresses: to0 using 200.1.1.1, e0 using 128.1.3.2, s0 using 192.0.1.1, and s1 using 223.254.254.1.

____

____

____

7 From a router's user mode, without using debugs or privileged mode, how can you determine what routers are sending you routing updates?

____

____

____

8 If the command **router rip** followed by **network 10.0.0.0**, with no other **network** commands, is configured in a router that has an Ethernet0 interface with IP address 168.10.1.1, does RIP send updates out Ethernet0?

____

____

____

9 If the commands **router igrp 1** and **network 10.0.0.0** are configured in a router that has an Ethernet0 interface with IP address 168.10.1.1, does IGRP advertise 168.10.0.0?

____

____

____

10 If the commands **router igrp 1** and **network 10.0.0.0** are configured in a router that has an Ethernet0 interface with IP address 168.10.1.1, mask 255.255.255.0, does this router have a route to 168.10.1.0?

____

____

____

11 Must IGRP metrics for multiple routes to the same subnet be exactly equal for the multiple routes to be added to the routing table? If not, how close in value do the metrics have to be?

12 When you're using RIP, what configuration command controls the number of equal-cost routes that can be added to the routing table at the same time? What is the maximum number of equal-cost routes to the same destination that can be included in the IP routing table at once?

13 When you're using IGRP, what configuration command controls the number of equal-cost routes that can be added to the routing table at the same time? What is the maximum number of equal-cost routes to the same destination that can be included in the IP routing table at once?

14 What feature supported by RIP-2 allows it to support variable-length subnet masks (VLSM)?

15 Name three features of RIP-2 that are not features of RIP-1.

16 What configuration commands are different between a router configured for RIP-1 and a router configured for only the support of RIP-2?

17 List the interior IP routing protocols that have autosummarization enabled by default. Which of these protocols allow autosummarization to be disabled using a configuration command?

18 Which interior IP routing protocols support route aggregation?

19 Which command lists all IP routes learned via RIP?

20 Which command or commands list all IP routes in network 172.16.0.0?

21 Assume that several subnets of network 172.16.0.0 exist in a router's routing table. What must be true about those routes so that the output of the **show ip route** command lists mask information only on the line that lists network 172.16.0.0 but doesn't show mask information on each route for each subnet?

22 True or false: Distance vector routing protocols learn routes by transmitting routing updates.

23 Assume that a router is configured to allow only one route in the routing table to each destination network. If more than one route to a particular subnet is learned, and if each route has the same metric value, which route is placed in the routing table if the routing protocol uses distance vector logic?

24 Describe the purpose and meaning of route poisoning.

25 Describe the meaning and purpose of triggered updates.

26 What term describes the underlying logic behind the OSPF routing protocol?

Scenarios

Scenario 7-1: IP Configuration 1

Your job is to deploy a new network. The network engineering group has provided a list of addresses and a network diagram, as shown in Figure 7-14 and Table 7-27.

Figure 7-14 *Scenario 7-1 Network Diagram*

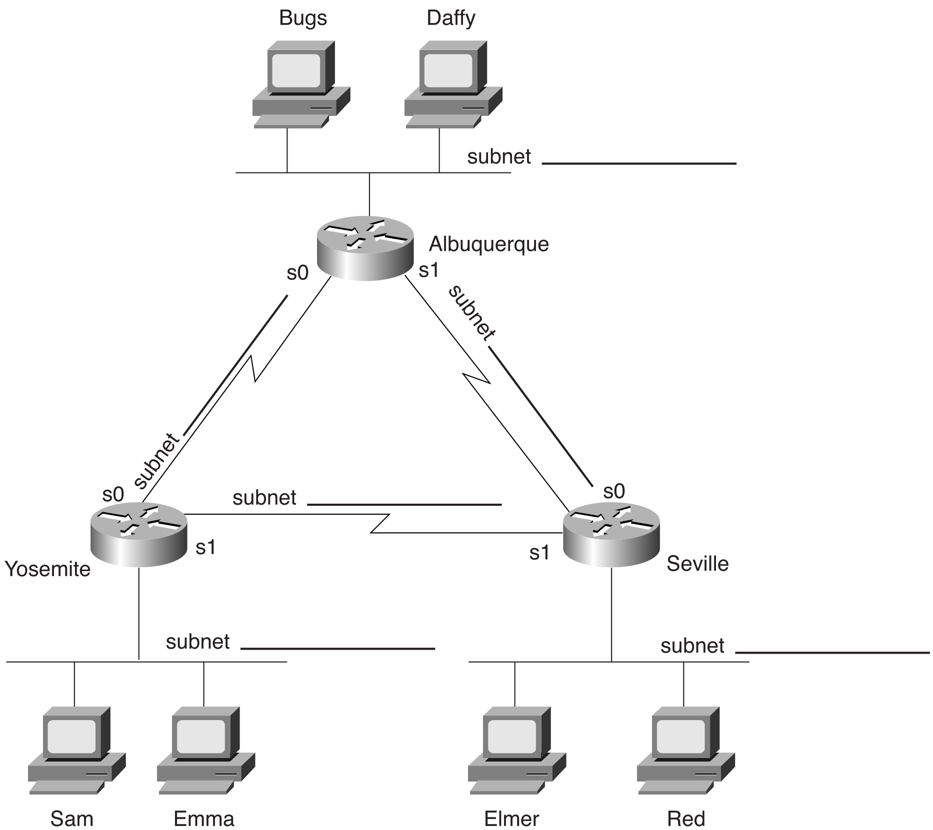


Table 7-27 *Scenario 7-1 IP Addresses*

Location of Subnet Geographically	Subnet Mask	Subnet Number	Subnet Broadcast
Ethernet off router in Albuquerque	255.255.255.0	148.14.1.0	
Ethernet off router in Yosemite	255.255.255.0	148.14.2.0	
Ethernet off router in Seville	255.255.255.0	148.14.3.0	
Serial between Albuquerque and Yosemite	255.255.255.0	148.14.4.0	
Serial between Albuquerque and Seville	255.255.255.0	148.14.5.0	
Serial between Seville and Yosemite	255.255.255.0	148.14.6.0	

Assuming the details established in Figure 7-14 and Table 7-27 for Scenario 7-1, complete or answer the following:

1. Create the configurations to enable IP as described in Table 7-27. Choose IP addresses as appropriate.
2. Describe the contents of the routing table on Seville after the routers are installed and all interfaces are up but no routing protocols or static routes have been configured.
3. Configure static routes for each router so that any host in any subnet can communicate with other hosts in this network.
4. Configure IGRP to replace the static routes in task 3.
5. Calculate the subnet broadcast address for each subnet.

Scenario 7-2: IP Configuration 2

Your job is to deploy a new network. The network engineering group has provided a list of addresses and a network diagram, with Frame Relay global DLCIs, as shown in Figure 7-15 and Table 7-28.

Figure 7-15 *Scenario 7-2 Network Diagram*

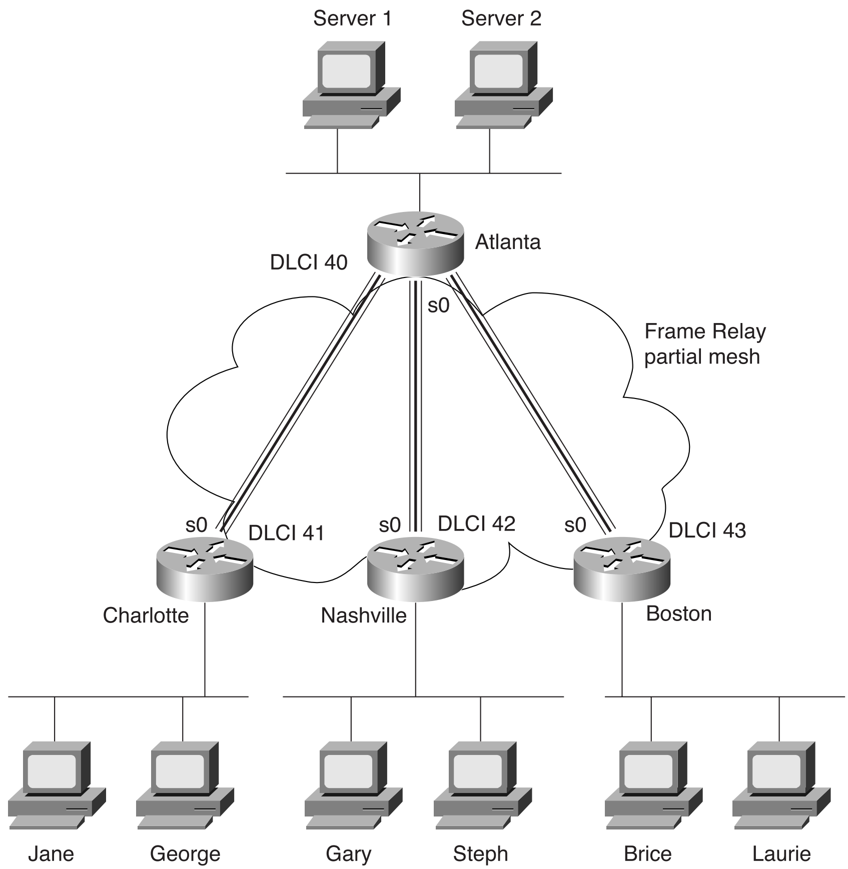


Table 7-28 *Scenario 7-2 IP Addresses*

Location of Subnet Geographically	Subnet Mask	Subnet Number	Subnet Broadcast
Ethernet off router in Atlanta	255.255.255.0	10.1.1.0	
Ethernet off router in Charlotte	255.255.255.0	10.1.2.0	

continues

Table 7-28 *Scenario 7-2 IP Addresses (Continued)*

Location of Subnet Geographically	Subnet Mask	Subnet Number	Subnet Broadcast
Ethernet off router in Nashville	255.255.255.0	10.1.3.0	
Ethernet off router in Boston	255.255.255.0	10.1.4.0	
VC between Atlanta and Charlotte	255.255.255.0	10.2.1.0	
VC between Atlanta and Nashville	255.255.255.0	10.2.2.0	
VC between Atlanta and Boston	255.255.255.0	10.2.3.0	

Assuming the details established in Figure 7-15 and Table 7-28 for Scenario 7-2, complete or answer the following:

1 Create the configurations to enable IP as described in Table 7-28. Do not enable a routing protocol.

2 Configure RIP.

3 Calculate the subnet broadcast address for each subnet.

4 Describe the contents of the RIP update sent from Boston to Atlanta. Also describe the contents of the RIP update sent from Atlanta to Charlotte.

Scenario 7-3: IP Addressing and Subnet Derivation

Complete the tasks and answer the questions following the upcoming figures and examples. Figure 7-16 shows the network diagram for Scenario 7-3, and Examples 7-22, 7-23, and 7-24 contain **show** command output from the three routers. Use Table 7-29 to record the subnet numbers and broadcast addresses as directed in the upcoming tasks.

Figure 7-16 *Scenario 7-3 Network Diagram*

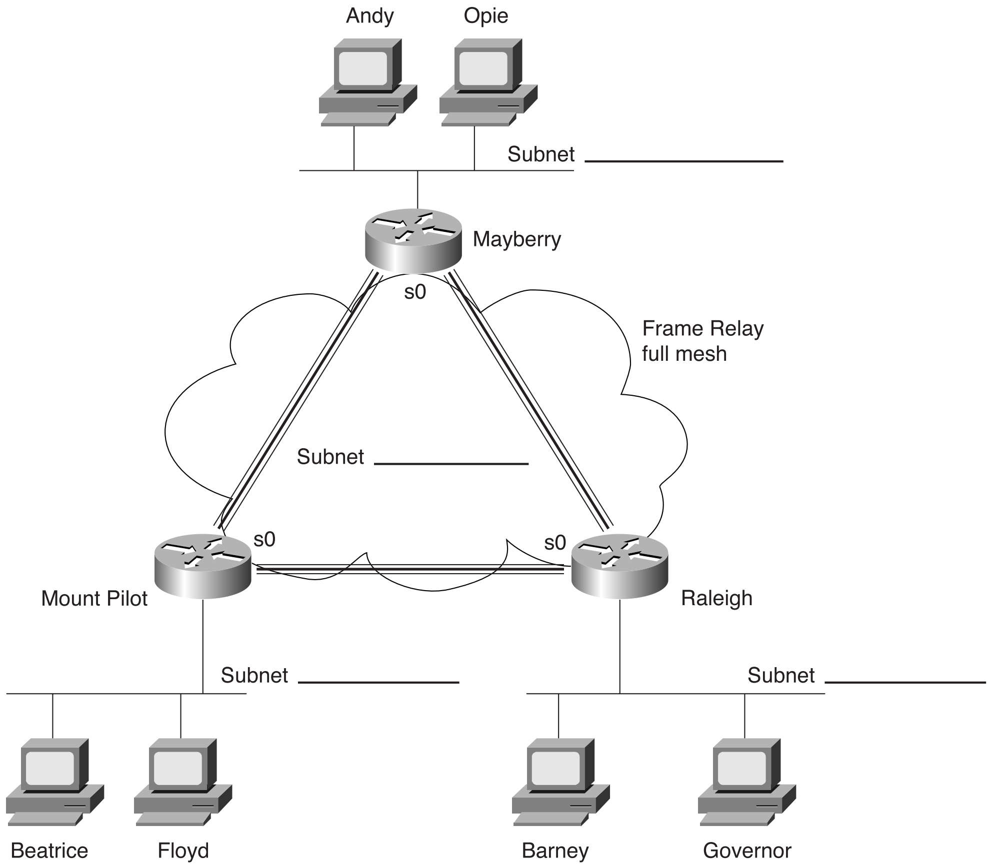


Table 7-29 *Subnets and Broadcast Addresses*

Location of Subnet Geographically	Subnet Mask	Subnet Number	Subnet Broadcast
Ethernet off router in Mayberry	255.255.255.0		
Ethernet off router in Mount Pilot	255.255.255.0		

continues

Table 7-29 *Subnets and Broadcast Addresses (Continued)*

Location of Subnet Geographically	Subnet Mask	Subnet Number	Subnet Broadcast
Ethernet off router in Raleigh	255.255.255.0		
VC between Mayberry and Mount Pilot	255.255.255.0		
VC between Mayberry and Raleigh	255.255.255.0		
VC between Mount Pilot and Raleigh	255.255.255.0		

Example 7-22 *Scenario 7-3:* **show** *Commands on Router Mayberry*

```
Mayberry#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     170.1.0.0/24 is subnetted, 4 subnets
C       170.1.1.0 is directly connected, Serial0
I       170.1.103.0 [100/8539] via 170.1.1.3, 00:00:50, Serial0
I       170.1.102.0 [100/8539] via 170.1.1.2, 00:00:32, Serial0
C       170.1.101.0 is directly connected, Ethernet0

Mayberry#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
Serial0                170.1.1.1       YES NVRAM  up                    up
Serial1                10.1.6.251      YES NVRAM  administratively down down
Ethernet0              170.1.101.1     YES NVRAM  up                    up

Mayberry#debug ip igrp transactions
IGRP protocol debugging is on
Mayberry#debug ip igrp events
IGRP event debugging is on
Mayberry#
IGRP: received update from 170.1.1.3 on Serial0
      subnet 170.1.1.0, metric 10476 (neighbor 8476)
      subnet 170.1.103.0, metric 8539 (neighbor 688)
      subnet 170.1.102.0, metric 10539 (neighbor 8539)
      subnet 170.1.101.0, metric 10539 (neighbor 8539)
IGRP: Update contains 4 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update: 4
```

Example 7-22 *Scenario 7-3:* **show** *Commands on Router Mayberry (Continued)*

```
IGRP: received update from 170.1.1.2 on Serial0
      subnet 170.1.1.0, metric 10476 (neighbor 8476)
      subnet 170.1.103.0, metric 10539 (neighbor 8539)
      subnet 170.1.102.0, metric 8539 (neighbor 688)
      subnet 170.1.101.0, metric 10539 (neighbor 8539)
IGRP: Update contains 4 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update: 4
IGRP: sending update to 255.255.255.255 via Serial0 (170.1.1.1)
      subnet 170.1.1.0, metric=8476
      subnet 170.1.103.0, metric=8539
      subnet 170.1.102.0, metric=8539
      subnet 170.1.101.0, metric=688
IGRP: Update contains 4 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update: 4
IGRP: sending update to 255.255.255.255 via Ethernet0 (170.1.101.1)
      subnet 170.1.1.0, metric=8476
      subnet 170.1.103.0, metric=8539
      subnet 170.1.102.0, metric=8539
IGRP: Update contains 3 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update:
```

Example 7-23 *Scenario 7-3:* **show** *Commands on Router Mount Pilot*

```
MountPilot#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

DLCI = 47, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0

  input pkts 38            output pkts 37           in bytes 3758
  out bytes 3514           dropped pkts 0           in FECN pkts 0
  in BECN pkts 0           out FECN pkts 0          out BECN pkts 0
  in DE pkts 0             out DE pkts 0
  out bcast pkts 36         out bcast bytes 3436
  pvc create time 00:17:39, last time pvc status changed 00:17:39

DLCI = 49, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0

  input pkts 31            output pkts 31           in bytes 3054
  out bytes 3076           dropped pkts 0           in FECN pkts 0
  in BECN pkts 0           out FECN pkts 0          out BECN pkts 0
  in DE pkts 0             out DE pkts 0
  out bcast pkts 31         out bcast bytes 3076
  pvc create time 00:17:40, last time pvc status changed 00:16:40

MountPilot#show frame-relay map

Serial0 (up): ip 170.1.1.1 dlci 47(0x2F,0x8F0), dynamic,
              broadcast,, status defined, active
Serial0 (up): ip 170.1.1.3 dlci 49(0x31,0xC10), dynamic,
              broadcast,, status defined, active
```

continues

Example 7-23 *Scenario 7-3:* **show** *Commands on Router Mount Pilot*

```
MontPilot#debug ip igrp packet
IGRP: sending update to 255.255.255.255 via Serial0 (170.1.1.2)
      subnet 170.1.1.0, metric=8476
      subnet 170.1.103.0, metric=8539
      subnet 170.1.102.0, metric=688
      subnet 170.1.101.0, metric=8539
IGRP: Update contains 4 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update: 4
IGRP: sending update to 255.255.255.255 via Ethernet0 (170.1.102.2)
      subnet 170.1.1.0, metric=8476
      subnet 170.1.103.0, metric=8539
      subnet 170.1.101.0, metric=8539
IGRP: Update contains 3 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update: 3
IGRP: received update from 170.1.1.1 on Serial0
      subnet 170.1.1.0, metric 10476 (neighbor 8476)
      subnet 170.1.103.0, metric 10539 (neighbor 8539)
      subnet 170.1.102.0, metric 10539 (neighbor 8539)
      subnet 170.1.101.0, metric 8539 (neighbor 688)
IGRP: Update contains 4 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update: 4
IGRP: received update from 170.1.1.3 on Serial0
      subnet 170.1.1.0, metric 10476 (neighbor 8476)
      subnet 170.1.103.0, metric 8539 (neighbor 688)
      subnet 170.1.102.0, metric 10539 (neighbor 8539)
      subnet 170.1.101.0, metric 10539 (neighbor 8539)
IGRP: Update contains 4 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update: 4

%FR-5-DLCICHANGE: Interface Serial0 - DLCI 47 state changed to DELETED
MountPilot#

IGRP: received update from 170.1.1.3 on Serial0
      subnet 170.1.1.0, metric 10476 (neighbor 8476)
      subnet 170.1.103.0, metric 8539 (neighbor 688)
      subnet 170.1.102.0, metric 10539 (neighbor 8539)
      subnet 170.1.101.0, metric 10539 (neighbor 8539)
IGRP: Update contains 4 interior, 0 system, and 0 exterior routes.
IGRP: Total routes in update: 4
```

Example 7-24 *Scenario 7-3:* **show** *Commands on Router Raleigh*

```
Raleigh#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set
```

Example 7-24 *Scenario 7-3:* **show** *Commands on Router Raleigh (Continued)*

```
     170.1.0.0/24 is subnetted, 4 subnets
C       170.1.1.0 is directly connected, Serial0
C       170.1.103.0 is directly connected, Ethernet0
I       170.1.102.0 [100/8539] via 170.1.1.2, 00:00:09, Serial0
I       170.1.101.0 [100/8539] via 170.1.1.1, 00:00:42, Serial0

Raleigh#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
Serial0                170.1.1.3       YES NVRAM  up                    up
Serial1                180.1.1.253     YES NVRAM  administratively down down
Ethernet0             170.1.103.3     YES NVRAM  up                    up

Raleigh#show ip protocol
Routing Protocol is "igrp 4"
  Sending updates every 90 seconds, next due in 56 seconds
  Invalid after 270 seconds, hold down 280, flushed after 630
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  IGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  IGRP maximum hopcount 100
  IGRP maximum metric variance 1
  Redistributing: igrp 4
  Maximum path: 4
  Routing for Networks:
    170.1.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    170.1.1.2            100      00:00:20
    170.1.1.1            100      00:00:53
  Distance: (default is 100)

Raleigh#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

DLCI = 47, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0

  input pkts 36            output pkts 35           in bytes 3674
  out bytes 3436           dropped pkts 0           in FECN pkts 0
  in BECN pkts 0           out FECN pkts 0          out BECN pkts 0
  in DE pkts 0             out DE pkts 0
  out bcast pkts 34         out bcast bytes 3358
  pvc create time 00:22:07, last time pvc status changed 00:21:58

DLCI = 48, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0

  input pkts 35            output pkts 35           in bytes 3444
  out bytes 3422           dropped pkts 0           in FECN pkts 0
  in BECN pkts 0           out FECN pkts 0          out BECN pkts 0
  in DE pkts 0             out DE pkts 0
  out bcast pkts 34         out bcast bytes 3358
  pvc create time 00:22:08, last time pvc status changed 00:21:58
```

Assuming the details established in Figure 7-16, Table 7-29, and Examples 7-22, 7-23, and 7-24 for Scenario 7-3, complete or answer the following:

1. Examining the **show** commands on the various routers, complete Table 7-29 with the subnet numbers and broadcast addresses used in this network.
2. Describe the contents of the IGRP update from Raleigh, sent out its virtual circuit to Mount Pilot. How many routes in Raleigh's IGRP update are sent to Mount Pilot? How many routes are in Raleigh's routing table? Is the number different? Why? (Hint: Look at the IGRP debug output in Example 7-23 and the IP routing table in Example 7-24.)
3. If the VC between Mount Pilot and Mayberry fails and routing protocol convergence completes, will Mayberry have a route to 170.1.1.0/24? Why or why not?Answers to Scenarios

Answers to Scenario 7-1: IP Configuration 1

Refer to the network illustrated in Figure 7-14 and Table 7-27 to establish the Scenario 7-1 design details and the context of the answers to the five tasks for this scenario.

Answers to Task 1 for Scenario 7-1

Task 1 for Scenario 7-1 asks for completed configurations. They are shown in Examples 7-25, 7-26, and 7-27. You could have chosen different IP addresses, but your choices must have the same first three octets as those shown in Example 7-25.

Example 7-25 *Albuquerque Configuration for Scenario 7-1*

```
hostname Albuquerque
!
enable secret 5 $1$ZvR/$Gpk5a5K5vTVpotd3KUygA1
!
interface Serial0
 ip address 148.14.4.1 255.255.255.0
!
interface Serial1
 ip address 148.14.5.1 255.255.255.0
!
interface ethernet0
 ip address 148.14.1.1 255.255.255.0
```

Example 7-26 *Yosemite Configuration for Scenario 7-1*

```
hostname Yosemite
enable secret 5 $1$ZvR/$Gpk5a5K5vTVpotd3KUygA1
!
interface Serial0
 ip address 148.14.4.2 255.255.255.0
```

Example 7-26 *Yosemite Configuration for Scenario 7-1 (Continued)*

```
!
interface Serial1
 ip address 148.14.6.2 255.255.255.0
!
interface ethernet0
 ip address 148.14.2.2 255.255.255.0
```

Example 7-27 *Seville Configuration for Scenario 7-1*

```
hostname Seville
enable secret 5 $1$ZvR/$Gpk5a5K5vTVpotd3KUygA1
!
interface Serial0
 ip address 148.14.5.3 255.255.255.0
!
interface Serial1
 ip address 148.14.6.3 255.255.255.0
!
interface ethernet0
 ip address 148.14.3.3 255.255.255.0
```

Answers to Task 2 for Scenario 7-1

Task 2 for Scenario 7-1 asks for a description of the IP routing table on Seville. This is shown in Table 7-30. This table exists before static and dynamic routes are added.

Table 7-30 *Routing Table in Seville*

Group	Outgoing Interface	Next-Hop Router
148.14.3.0	e0	
148.14.5.0	s0	
148.14.6.0	s1	

The Next-Hop Router field is always the IP address of another router, or it is null if the route describes a directly connected network.

Answers to Task 3 for Scenario 7-1

Task 3 for Scenario 7-1 asks for static route configuration. The routes to allow users on LANs to reach each other are shown in upcoming examples. However, routes to the subnets on serial links are not shown in these examples for the sake of brevity. The users should not need to send packets to IP addresses on the serial links' subnets, but rather to other hosts on the LANs. Examples 7-28, 7-29, and 7-30 show the configurations on the three routers.

Example 7-28 *Albuquerque Configuration for Scenario 7-1*

```
ip route 148.14.2.0 255.255.255.0 148.14.4.2
ip route 148.14.3.0 255.255.255.0 serial1
```

Example 7-29 *Yosemite Configuration for Scenario 7-1*

```
ip route 148.14.1.0 255.255.255.0 148.14.4.1
ip route 148.14.3.0 255.255.255.0 serial1
```

Example 7-30 *Seville Configuration for Scenario 7-1*

```
ip route 148.14.1.0 255.255.255.0 148.14.5.1
ip route 148.14.2.0 255.255.255.0 serial1
```

Both valid styles of static route configuration are shown. In any topological case, the style of static route command using the next router's IP address is valid. If the route points to a subnet that is on the other side of a point-to-point serial link, the static route command can simply refer to the outgoing serial interface.

Answers to Task 4 for Scenario 7-1

Task 4 for Scenario 7-1 asks for IGRP configuration. The same configuration is used on each router. It is shown in Example 7-31. The IGRP process-id must be the same on each router. If an IGRP update is received but lists a different process-id, the update is ignored.

Example 7-31 *IGRP Configuration for Scenario 7-1*

```
router igrp 1
network 148.14.0.0
```

Answers to Task 5 for Scenario 7-1

Task 5 for Scenario 7-1 asks for the broadcast addresses for each subnet. Table 7-31 shows these.

Table 7-31 *Completed Scenario 7-1 IP Addresses*

Location of Subnet Geographically	Subnet Mask	Subnet Number	Subnet Broadcast
Ethernet off router in Albuquerque	255.255.255.0	148.14.1.0	148.14.1.255
Ethernet off router in Yosemite	255.255.255.0	148.14.2.0	148.14.2.255
Ethernet off router in Seville	255.255.255.0	148.14.3.0	148.14.3.255
Serial between Albuquerque and Yosemite	255.255.255.0	148.14.4.0	148.14.4.255
Serial between Albuquerque and Seville	255.255.255.0	148.14.5.0	148.14.5.255
Serial between Seville and Yosemite	255.255.255.0	148.14.6.0	148.14.6.255

Answers to Scenario 7-2: IP Configuration 2

Refer to the network illustrated in Figure 7-15 and Table 7-28 to establish the Scenario 7-2 design details and the context of the answers to the four tasks for this scenario.

Answers to Task 1 for Scenario 7-2

Task 1 for Scenario 7-2 asks for completed configurations. They are shown in Examples 7-32 through 7-35.

Example 7-32 *Atlanta Configuration for Scenario 7-2*

```
hostname Atlanta
no ip domain-lookup
!
interface serial0
encapsulation frame-relay
interface serial 0.1
ip address 10.2.1.1 255.255.255.0
frame-relay interface-dlci 41
!
interface serial 0.2
ip address 10.2.2.1 255.255.255.0
frame-relay interface-dlci 42
!
interface serial 0.3
ip address 10.2.3.1 255.255.255.0
frame-relay interface-dlci 43
!
interface ethernet 0
ip address 10.1.1.1 255.255.255.0
```

Example 7-33 *Charlotte Configuration for Scenario 7-2*

```
hostname Charlotte
no ip domain-lookup
!
interface serial0
encapsulation frame-relay
interface serial 0.1
ip address 10.2.1.2 255.255.255.0
frame-relay interface-dlci 40
!
interface ethernet 0
ip address 10.1.2.2 255.255.255.0
```

Example 7-34 *Nashville Configuration for Scenario 7-2*

```
hostname nashville
no ip domain-lookup
!
interface serial0
encapsulation frame-relay
```

continues

Example 7-34 *Nashville Configuration for Scenario 7-2 (Continued)*

```
interface serial 0.1
ip address 10.2.2.3 255.255.255.0
frame-relay interface-dlci 40
!
interface ethernet 0
ip address 10.1.3.3 255.255.255.0
```

Example 7-35 *Boston Configuration for Scenario 7-2*

```
hostname boston
no ip domain-lookup
!
interface serial0
encapsulation frame-relay
interface serial 0.1
ip address 10.2.3.4 255.255.255.0
frame-relay interface-dlci 40
!
interface ethernet 0
ip address 10.1.4.4 255.255.255.0
```

Answers to Task 2 for Scenario 7-2

Task 2 for Scenario 7-2 asks for RIP configuration. The same configuration is used on each router. It is shown in Example 7-36.

Example 7-36 *RIP Configuration for Scenario 7-2*

```
router rip
network 10.0.0.0
```

Answers to Task 3 for Scenario 7-2

Task 3 for Scenario 7-2 asks for the broadcast addresses for each subnet. Table 7-32 shows these.

Table 7-32 *Completed Scenario 7-2 IP Addresses*

Location of Subnet Geographically	Subnet Mask	Subnet Number	Subnet Broadcast
Ethernet off router in Atlanta	255.255.255.0	10.1.1.0	10.1.1.255
Ethernet off router in Charlotte	255.255.255.0	10.1.2.0	10.1.2.255
Ethernet off router in Nashville	255.255.255.0	10.1.3.0	10.1.3.255
Ethernet off router in Boston	255.255.255.0	10.1.4.0	10.1.4.255
VC between Atlanta and Charlotte	255.255.255.0	10.2.1.0	10.2.1.255
VC between Atlanta and Nashville	255.255.255.0	10.2.2.0	10.2.2.255
VC between Atlanta and Boston	255.255.255.0	10.2.3.0	10.2.3.255

Answers to Task 4 for Scenario 7-2

Task 4 for Scenario 7-2 requires that you consider the effects of split horizon. Split-horizon logic considers subinterfaces to be separate interfaces, in spite of the fact that several subinterfaces share the same physical interface. Boston advertises 10.1.4.0 in its RIP update only out its subinterface 1. All other routes in Boston's routing table are learned through RIP updates from Atlanta via updates entering that same subinterface. Therefore, Boston does not advertise those routes in updates it sends on that same subinterface.

The RIP updates from Atlanta to Charlotte out Atlanta's subinterface 1 advertise all subnets not learned from RIP updates entering that same subinterface. All subnets except 10.1.2.0 (learned from Charlotte) and 10.2.1.0 (subinterface 1's subnet) are listed in Atlanta's RIP update to Charlotte. Subnet 10.1.4.0, learned from Boston, is indeed included in updates to Charlotte. Split horizon considers subinterfaces to be separate interfaces.

Answers to Scenario 7-3: IP Addressing and Subnet Derivation

Refer to the network illustrated in Figure 7-16 and Examples 7-22, 7-23, and 7-24 to establish the Scenario 7-3 design details and the context of the answers to the three tasks for this scenario.

Answers to Task 1 for Scenario 7-3

Task 1 for Scenario 7-3 asks you to complete a table with the subnet numbers and broadcast addresses used in this scenario's network after examining the **show** commands on the various routers in Examples 7-22, 7-23, and 7-24. Table 7-33 lists the subnet numbers and broadcast addresses requested in this task.

Table 7-33 *Completed Subnets and Broadcast Addresses*

Location of Subnet Geographically	Subnet Mask	Subnet Number	Subnet Broadcast
Ethernet off router in Mayberry	255.255.255.0	170.1.101.0	170.1.101.255
Ethernet off router in Mount Pilot	255.255.255.0	170.1.102.0	170.1.102.255
Ethernet off router in Raleigh	255.255.255.0	170.1.103.0	170.1.103.255
VC between Mayberry and Mount Pilot	255.255.255.0	170.1.1.0	170.1.1.255
VC between Mayberry and Raleigh	255.255.255.0	170.1.1.0	170.1.1.255
VC between Mount Pilot and Raleigh	255.255.255.0	170.1.1.0	170.1.1.255

Notice that the same subnet is used for all three virtual circuits. A full mesh of virtual circuits is used, and a single subnet was chosen rather than one subnet per virtual circuit.

Answers to Task 2 for Scenario 7-3

Task 2 for Scenario 7-3 asks you to describe the contents of the IGRP update from Raleigh, sent out its virtual circuit to Mount Pilot. Notice that there are four routes in the routing table and four routes in the routing update. Split horizon is disabled on serial interfaces using Frame Relay as configured without subinterfaces. Split horizon is disabled by the Cisco IOS Software if Frame Relay multipoint subinterfaces are used as well. Therefore, all four routes in the IP routing table are advertised in routing updates sent out Serial0.

Answers to Task 3 for Scenario 7-3

Mayberry still has a route to 170.1.1.0/24, which is the subnet covering all the Frame Relay interfaces in this scenario. Because only one VC went down and the other VC is still up, it is reasonable to expect that the physical interface is still up. No subinterfaces are configured in this scenario, so Mayberry still has a connected route for each interface that’s currently up, including 170.1.1.0/24 on Serial0.