CCNA™

CISCO® CERTIFIED NETWORK ASSOCIATE

11

Configuring Advanced Routing Protocols

CERTIFICATION OBJECTIVES

11.01 OSPF

11.02 EIGRP

✓ Two-Minute Drill

Q&A Self Test

In Chapter 10, you were introduced to the configuration of two distance vector routing protocols: IP RIP and IGRP. This chapter focuses on two advanced routing protocols: OSPF and EIGRP. OSPF is a link state protocol, and EIGRP is a hybrid protocol. This chapter covers only basic operation and configuration of these protocols. A more thorough discussion is covered in Cisco's BSCI CCNP and CCDP exam.

OSPF

The Open Shortest Path First (OSPF) protocol is a link state protocol that handles routing for IP traffic. Its newest implementation, version 2, which is explained in RFC 2328, is an open standard, like RIP. Chapter 9 offered a brief introduction to link state protocols. As you will see in this section, OSPF draws heavily on the concepts described in that chapter, but it also has some features of its own. Besides covering the characteristics of OSPF, you'll be presented with enough information to undertake a very basic routing configuration using OSPF.

Characteristics of OSPF

OSPF was created in the mid-1980s in order to overcome many of the deficiencies and scalability problems that RIP had in large enterprise networks. Because it is based on an open standard, OSPF is very popular in many corporate networks today and has many advantages, including these:

- It will run on most routers, since it is based on an open standard.
- It uses the SPF algorithm, developed by Dijkstra, to provide a loop-free topology.
- It provides fast convergence with triggered, incremental updates via Link State Advertisements (LSAs).
- It is a classless protocol and allows for a hierarchical design with VLSM and route summarization.

Given its advantages, OSPF does have its share of disadvantages:

- It requires more memory to hold the adjacency (list of OSPF neighbors), topology (a link state database containing all of the routers and their routes), and routing tables.
- It requires extra CPU processing to run the SPF algorithm, which is especially true when you first turn on your routers and they are initially building the adjacency and topology tables.

- For large networks, it requires careful design to break up the network into an appropriate hierarchical design by separating routers into different *areas*.
- It is complex to configure and more difficult to troubleshoot.

Knowing the advantages and disadvantages of any routing protocol is useful when it comes to picking a protocol. Typically, OSPF is used in large enterprise networks that have either a mixed routing vendor environment or a policy that requires an open standard for a routing protocol, which gives a company flexibility when it needs to replace any of its existing routers.

exam watch

Remember the advantages and disadvantages of OSPF, listed in the preceding bullets. Also, classless protocols include the subnet mask value along with the route when advertising routing information: distance vector protocols do not include the subnet mask in their routing updates.

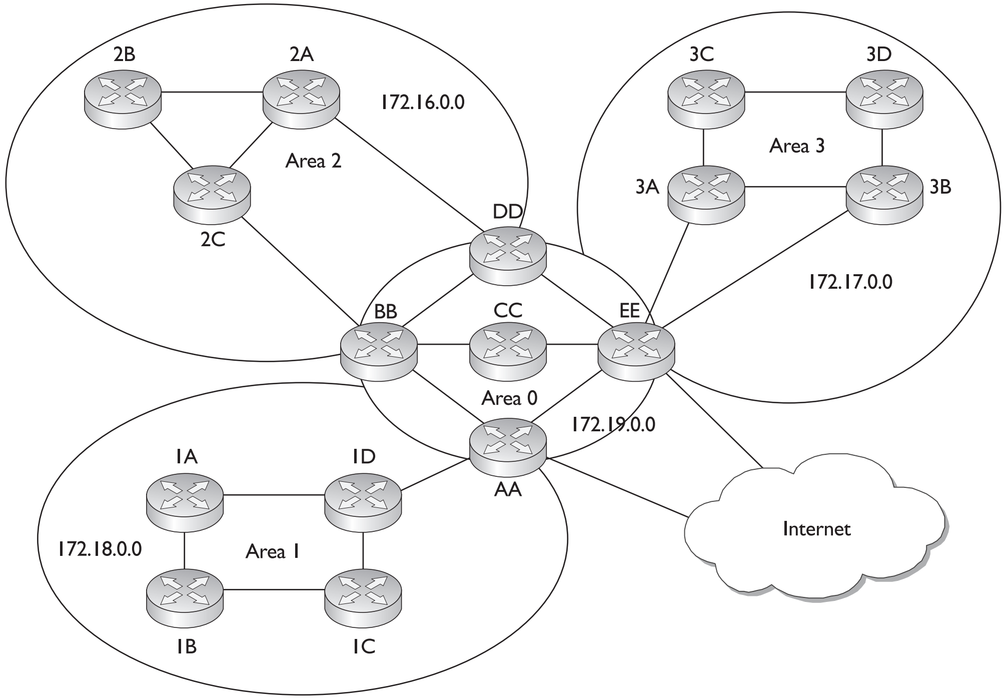
Hierarchical Design: Areas

To provide scalability to very large networks, OSPF supports two important concepts: autonomous systems and areas. Autonomous systems were discussed in Chapter 9. Within an AS, *areas* are used to provide hierarchical routing. Basically, areas are used to control when and how much routing information is shared across your network. In flat network designs, such as IP RIP, if a change occurs on one router, perhaps a flapping route problem, it affects *every* router in the entire network. With a correctly designed hierarchical network, these changes can be contained within a single area.

OSPF implements a two-layer hierarchy: the backbone (area 0) and areas off of the backbone (areas 1–65,535), as is shown in Figure 11-1. This network includes a backbone and three areas off of the backbone. Through a correct IP addressing design, you should be able to summarize routing information between areas. By summarizing your routing information, perhaps one summarized route for each area, you are reducing the amount of information that routers need to know about. For instance, each area in Figure 11-1 is assigned a separate Class B network number. Through summarization on the border routers between areas, other areas would not need to see all the Class B subnets—only the summarized network numbers.

For instance, Area 2 doesn't need to see all of the subnets of Area 3's 172.18.0.0 network number, since there are only two paths out of Area 2 to the backbone. Area 2, however, needs to see all of its internal subnets to create optimized routing tables to reach internal networks. Therefore, each area should contain specific routes only for

FIGURE 11-1 OSPF hierarchical design



its own areas and summarized routes to reach other areas. By performing this summarization, the routers have a smaller topology database (they know only about links in their own area and the summarized routes) and their routing tables are smaller (they know only about their own area's routes and the summarized routes). Through a correct hierarchical design, you can scale OSPF to very large sizes.

exam watch

OSPF supports a two-layer hierarchy: the backbone (area 0) and areas connected to the backbone.

Note that the CCNA exam focuses on single-area designs, and throughout the rest of the sections, the material covers only single-area concepts. The BSCI exam for the CCNP and CCDP certifications, however, spends a lot of time on both single- *and* multi-area designs. Designing a multi-area OSPF network can become very complicated and requires a lot of networking knowledge and skill.

Metric Structure

Unlike RIP, which uses hop count as a metric, OSPF uses cost. Cost is actually the inverse of the bandwidth of a link: the faster the speed of the connection, the lower the cost. The most preferred path is the one with the lowest cost. By using cost as a metric, OSPF will choose more intelligent paths than RIP.

Remember that on synchronous serial links, no matter what the clock rate of the physical link is, the bandwidth always defaults to 1544 Kbps. You'll want to code this correctly with the **`bandwidth`** *Interface Subconfiguration* mode command. This is important if you have multiple synchronous serial paths to a destination, especially if they have different clock rates. OSPF supports load balancing of up to six equal-cost paths to a single destination. However, if you don't configure the bandwidth metric correctly on your serial interfaces, your router might accidentally include paths with different clock rates, which can cause load-balancing issues.

For example, if you have one serial connection clocked at 1,544 Kbps and another clocked at 256 Kbps and you don't change the bandwidth values, OSPF will see *both* connections as 1,544 Kbps and attempt to use both when reaching a single destination. This can create throughput problems when the router is performing load balancing—half of the traffic will go down one link and half down the other, creating congestion problems.

exam watch

OSPF uses cost as a metric, which is the inverse of the bandwidth of a link.

Router Identities

Each router in an OSPF network needs a unique ID. The ID is used to provide a unique identity to the OSPF router. This is included in any OSPF messages the router generates. The router ID is chosen according to one of the two following criteria:

- The highest IP address on its loopback interfaces (this is a logical interface on a router)
- The highest IP address on its active interfaces

exam watch

Remember how a router acquires its router ID for OSPF.

If you have an IP address on an active loopback interface, the router will use the highest IP address from the bunch for its router ID. The router ID is used by the router to announce itself to the other OSPF routers in the network. This ID must be unique. If you have no loopback interfaces configured, then the router will use the highest IP address from one of its physical interfaces. If there is no active interface, the OSPF process will not start and therefore you will not have any OSPF routes in your routing table. It is highly recommended that you use a loopback interface because it is always up and thus the router can obtain a router ID.

Finding Neighbors

OSPF learns about its neighbors and builds its adjacency and topology tables by sharing LSAs. There are different types of LSAs. When learning about the neighbors that a router is connected to, as well as keeping tabs on known neighbors, OSPF routers will generate hello LSAs every 10 seconds. When a neighbor is discovered and an adjacency is formed with the neighbor, a router expects to see hello messages from the neighbor. If a neighbor is not seen within the dead interval time, which defaults to 40 seconds, the neighbor is declared dead. When this occurs, the router will advertise this information, via an LSA message, to other neighboring OSPF routers.

Whereas RIP accepts routing updates from just about any other RIP router, OSPF has some rules concerning if and how routing information should be shared. First, before a router will accept any routing information from another OSPF router, they have to build an *adjacency* with each other on their connected interfaces. When this adjacency is built, the two routers (on the connected interfaces) are called *neighbors*, which indicates a special relationship between the two. In order for two routers to become neighbors, the following must match on each router:

- The area number and its type
- The hello and dead interval timers
- The OSPF password (optional), if it is configured
- The area stub flag (used to contain OSPF messages and routing information, this is beyond the scope of this book)

If these items do not match, then the routers will not form an adjacency and will ignore each other’s routing information.

Let’s assume that you turned on all your routers simultaneously on a segment. In this case, the OSPF routers will go through three states called the *exchange process*:

1. ***Down* state** The new router has not exchanged any OSPF information with any other router.
2. ***Init* state** A destination router has received a new router's hello and adds it to its neighbor list (assuming that certain values match). Note that communication is only unidirectional at this point.

3. ***Two-Way* state** The new router receives a unidirectional reply to its initial hello packet and adds the destination router to its neighbor database.

Once the routers have entered a *two-way* state, they are considered neighbors. At this point, an election process takes place to elect the designated router (DR) and the backup designated router (BDR).

Designated and Backup Designated Routers

An OSPF router will not form adjacencies to just any router. Instead, a client/server design is implemented in OSPF. For each network multi-access segment, there is a DR and a BDR as well as other routers. As an example, if you have ten VLANs in your switched area, you'll have ten DRs and ten BDRs. The one exception of a segment not having these two routers is on a WAN point-to-point link.

When an OSPF router comes up, it forms adjacencies with the DR and the BDR on each multi-access segment that it is connected to. Any exchange of routing information is between these DR/BDR routers and the other OSPF neighbors on a segment (and vice versa). An OSPF router talks to a DR using the IP multicast address of 224.0.0.6. The DR and the BDR talk to all routers using the 224.0.0.5 multicast IP address.

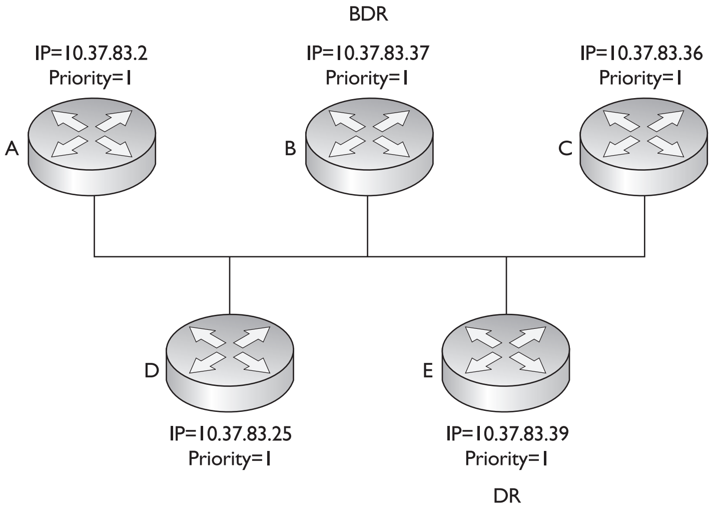
The OSPF router with the highest priority becomes the DR for the segment. If there is a tie, the router with the highest router ID will become the DR. By default, all routers have a priority of *1* (priorities can range 0–255). If the DR fails, the BDR is promoted to DR and another router is elected as the BDR. Figure 11-2 shows an example of the election process, where router E is elected as the DR and router B, the BDR.

exam watch

OSPF routers use Link State Advertisements (LSAs) to communicate with each other. One type of LSA is a hello, which is used to form neighbor relationships and as a keep-alive function. Hellos are generated every ten seconds. When sharing link information (directly connected routes), links are sent to the DR (224.0.0.6) and the DR disseminates this to everyone (224.0.0.5) else on the segment. The router with the highest priority (or highest router ID) becomes the DR. This process is true for multi-access segments, but not point-to-point links, where DRs are not necessary.

FIGURE 11-2

DR and BDR election process



Sharing Routing Information

After electing the DR/BDR pair, the routers continue to generate hellos to maintain communication. This is considered an *exstart* state, in which the OSPF routers are ready to share link state information. The process the routers go through is called an *exchange protocol*:

1. ***Exstart* state** The DR and BDR form adjacencies with the other OSPF routers on the segment, and then within each adjacency, the router with the highest router ID becomes the master and starts the exchange process first (shares its link state information)—note that the DR is not necessarily the master for the exchange process. The remaining router in the adjacency will be the slave.
2. ***Exchange* state** The master starts sharing link state information first, with the slave. These are called DBDs (database description packets), also referred to as DDPs. The DBDs contain the link-state type, the ID of the advertising router, the cost of the advertised link, and the sequence number of the link. The slave responds back with an LSACK—an acknowledgment to the DBD from the master. The slave then compares the DBD's information with its own.
3. ***Loading* state** If the master has more up-to-date information than the slave, the slave will respond to the master's original DBD with an LSR (Link State Request). The master will then send a LSU (Link State Update) with the detailed information of the links to the slave. The slave will then incorporate this into its local link state database. Again, the slave will generate an LSACK

to the master to acknowledge the fact that it received the LSU. If a slave has more up-to-date information, it will repeat the "exchange" and "loading" states.

4. ***Full* state** Once the master and the slave are synchronized, they are considered to be in a full state.

To summarize these four steps, OSPF routers share a type of LSA message in order to disclose information about available routes. Basically, an LSA update message contains a link and a state, as well as other information. A *link* is the router interface on which the update was generated (a connected route). The *state* is a description of this interface, including the IP address configured on it as well as the relationship this router has with its neighboring router. However, OSPF routers will not share this information with just any OSPF router.

OSPF uses incremental updates after entering a full state. This means that whenever changes take place, only the change is shared with the DR, which will then share this information with other routers on the segment. Figure 11-3 shows an example of this. In this example, Network Z, connected to router C, goes down. Router C sends a multicast to the DR and the BDR (with a destination multicast address of 224.0.0.6), telling them about this change. Once the DR and the BDR incorporate the change internally, the DR then tells the other routes on the segment (via a multicast message sent to 224.0.0.5, which is all OSPF routers) about the change concerning Network Z. Any router receiving the update will then share this update to the DRs of other segments that they are connected to.

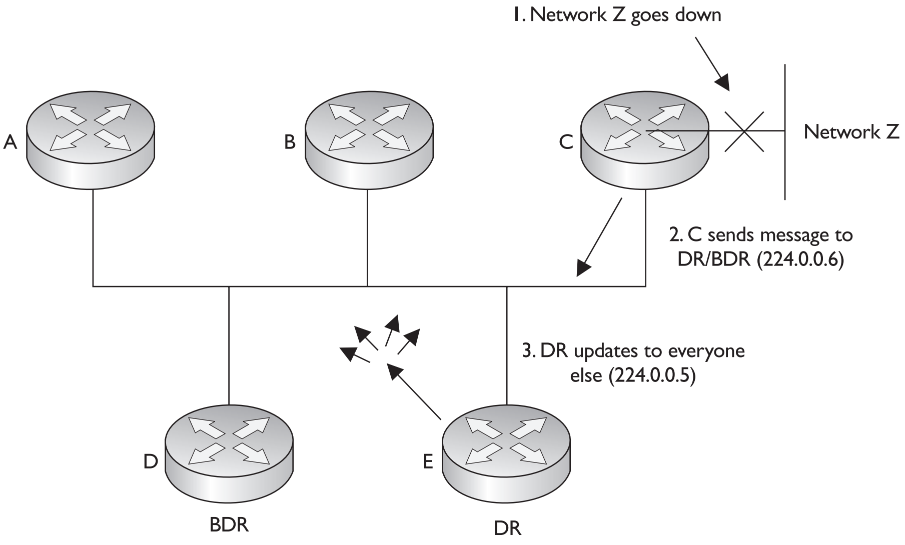
Note that the communications between OSPF routers is connection-oriented, even though multicasts are used. For example, if a router tells a DR about a change, the DR acknowledges this new piece of information. Likewise, when the DR shares this information with the other routers on the segment, the DR expects acknowledgments back from each of these neighbors. Remember that when an OSPF router exchanges

exam watch

OSPF routers share information about their *connected* ***routes with the DR, which includes the link-state type, the ID of the advertising router, the cost of the advertised link, and the sequence number of the link. This is different from distance vector protocols. Distance vector protocols share their entire routing table with their neighbors with the exception of routes learned from the same interface of the neighbor (split horizon) and the connected route of the interface where the neighbor resides.***

FIGURE 11-3

LSA update process


1. Network Z goes down
Network Z
A
B
C
2. C sends message to DR/BDR (224.0.0.6)
3. DR updates to everyone else (224.0.0.5)
D
E
BDR
DR

updates with another, the process requires an acknowledgment: this ensures that router or routers have received the update.

The exception to the incremental update process is that the DR floods its database every 30 minutes to ensure that all of the routers on the segment have the most up-to-date link state information. It does this with a destination address of 224.0.0.5 (all OSPF routers on the segment).

When building the routing table using link state information, an OSPF router can keep up to six paths to a destination in its routing table. The only restriction is that the paths must have the same cost.

exam watch

A two-way state indicates that two OSPF routers are neighbors. A full state indicates the completion of sharing of links between routers.

Configuring OSPF

Configuring OSPF is slightly *different* from configuring RIP or IGRP. When configuring OSPF, use the following syntax:

```
Router(config)# router ospf process_ID
Router(config-router)# network IP_address wildcard_mask
                         area area_#
```

The *process_ID* is locally significant and is used to differentiate between different OSPF processes running on the router. Your router might be a boundary router between two OSPF autonomous systems, and to differentiate them on your router, you'll give them unique process IDs. Note that these numbers do *not* need to match between different routers and that they have nothing to do with autonomous system numbers.

When specifying what interfaces go into an area for OSPF, use the **network** command. As you can see in the preceding example, the syntax of this command is different than for RIP's and IGRP's configuration, where you specify only a class address. OSPF is classless. With this command, you can be very specific about what interface belongs to a particular area. The syntax of this command is to list an IP address followed by a *wildcard mask.* This is *different* from a subnet mask. A wildcard mask tells the router the interesting component of the address—in other words, what part of the address it should match on. This mask is also used with access lists, which are discussed in Chapter 13.

A wildcard mask is 32 bits in length. A 0 in a bit position means there must be a match, and a 1 in a bit position means the router doesn't care. Actually, a wildcard mask is an *inverted* subnet mask, with the 1's and 0's switched. Using a wildcard mask, you can be very specific about which interfaces belong to which areas. The last part of the command tells the router which area these addresses on the router belong to.

Let's look at some code examples to see how the wildcard mask works. I'll use the router shown in Figure 11-4 as an illustration.

```
Router(config)# router ospf 1
Router(config-router)# network 10.1.1.1  0.0.0.0  area 0
Router(config-router)# network 10.1.2.1  0.0.0.0  area 0
Router(config-router)# network 172.16.1.1  0.0.0.0  area 0
Router(config-router)# network 172.16.2.1  0.0.0.0  area 0
```

In this example, the interfaces with addresses of 10.1.1.1, 10.1.2.1, 172.16.1.1, and 172.16.1.1 all are associated with area 0. A wildcard mask of 0.0.0.0 says that there must be an exact match against the address in order to place it into area 0. Here's another example:

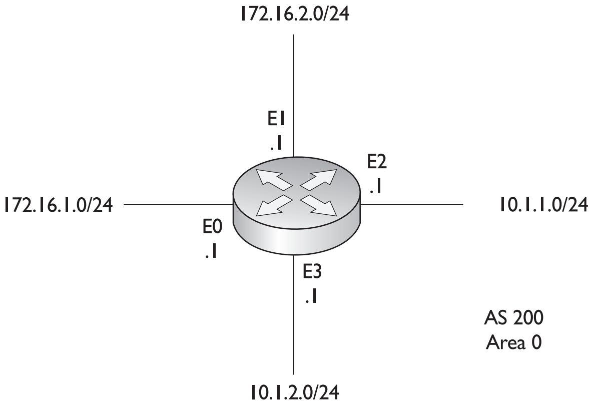
```
Router(config)# router ospf 1
Router(config-router)# network 10.0.0.0 0.255.255.255 area 0
Router(config-router)# network 172.16.0.0 0.0.255.255 area 0
```

In this example, interfaces beginning with 10 or 172.16 are to be associated with area 0. Or, if all the interfaces on your router belonged to the same area, you could use this configuration:

```
Router(config)# router ospf 1
Router(config-router)# network 0.0.0.0 255.255.255.255 area 0
```

FIGURE 11-4

OSPF network configuration example



In this example, all interfaces are placed in area 0. As you can see, OSPF is very flexible in allowing you to specify which interface or interfaces will participate in OSPF and which area or areas they will belong to.

CertCam

11.01. The CD contains a multimedia demonstration of configuring OSPF on a router.

exam watch

When configuring the OSPF routing process, you must specify a process ID. Unlike in RIP or IGRP, the `network` statement allows you to specify an IP address and a wildcard mask, which is an inverted subnet mask. You must also specify the area that this address or addresses will belong to: `network` `network_# wildcard_mask` `area` `area_#`.

Loopback Interfaces

A *loopback interface* is a logical, virtual interface on a router. By default, the router doesn't have any loopback interfaces, but they can easily be created. All IOS platforms support loopback interfaces, and you can create as many of these interfaces as you need. These interfaces are treated as physical interfaces on a router: you can assign addressing information to them, include their network numbers in routing updates, and even terminate IP connections on them, like telnet. Here are some reasons you might want to create a loopback interface:

- To assign a router ID to an OSPF router
- To use for testing purposes, since this interface is always up
- To terminate special connections, such as GRE tunnels or IPSec connections, since this interface is always up

To create a loopback interface, use the following command:

```
Router(config)# interface loopback port_#
Router(config-if)# ip address IP_address subnet_mask
```

As you can see, creating a loopback interface is easy. You can specify port numbers from 0 to 2147483647. The number you use is only locally significant. Once you enter the loopback interface, you can execute almost any interface command on it; for instance, you can assign it an IP address with the **`ip address`** command.

> exam watch
>
> ***A loopback interface is a logical interface that always remains up. Use the `interface loopback` command to create it.***

CertCam

11.02. The CD contains a multimedia demonstration of creating a loopback interface on a router.

Changing Metrics

You have two ways to affect the cost metric that OSPF uses in picking the best-cost routes for the routing table. First, remember that the cost metric is the inverse of the accumulated bandwidth values of routers' interfaces. The default measurement that Cisco uses in calculating the cost metric is: cost = 10^8/(interface bandwidth). You can also affect the value of the cost by changing the 10^8 value with the **`auto-cost reference-bandwidth`** command. Table 11-1 contains some costs for different interface types:

> exam watch
>
> ***Remember the OSPF interface costs in Table 11-1; especially for serial connections.***

To change the cost of an interface, use the following configuration:

```
Router(config)# interface type [slot_#/]port_#
Router(config-if)# ip ospf cost cost_value
```

Notice that the cost is assigned within an interface. This value can range from 1 to 65,535. Note that each vendor might use a different calculation to come up with a cost value. It is very important that the costs for a link match for every router on a

TABLE 11-1 OSPF Costs for Different Interfaces

Bandwidth Value	Interface Type
1785	56 Kbps serial line
1652	64 Kbps serial line
64	T1
25	4Mb Token Ring
10	Ethernet
6	16Mb Token Ring
1	Fast Ethernet and FDDI

given segment. Mismatched cost values on a segment can cause routers to continually run the SPF algorithm, greatly affecting the routers' performance.

Normally, you won't be changing the default cost values on an interface. However, since OSPF uses the inverse of bandwidth as a metric, and serial interfaces default to a bandwidth of 1,544 Kbps, you will definitely want to match the bandwidth metric on the serial interface to its real clock rate. To configure the bandwidth on your router's interfaces, use the following command:

```
Router(config) interface type [slot_#/]port_#
Router(config-if)# bandwidth speed_in_Kbps
```

As an example, if the clock rate were 64,000, you would use the following command to correctly configure the bandwidth: **`bandwidth 64`**. Note that the speed is in Kbps. For example, let's assume you configured the bandwidth with this: **`bandwidth 64000`**. By doing this, the router would assume the bandwidth metric of the interface is 64 Mbps, not Kbps.

CertCam

11.03. The CD contains a multimedia demonstration of changing OSPF metrics on a router.

exam watch

The `bandwidth` command should be used on synchronous serial interfaces to match the bandwidth metric to the clocked rate of the interface. Synchronous serial interfaces default to a bandwidth metric of 1,544 Kbps.

Troubleshooting OSPF

Once you have configured OSPF, you have a variety of commands available to view and troubleshoot your configuration and operation of OSPF:

- **show ip protocols**
- **show ip route**
- **show ip ospf interface**
- **show ip ospf neighbor**
- **debug ip ospf adj**
- **debug ip ospf events**
- **debug ip ospf packet**

The following sections cover these commands.

The show ip protocols Command

The **show ip protocols** command displays all of the IP routing protocols that you have configured and are running on your router. Here's an example of this command with OSPF:

```
Router# show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
 Incoming update filter list for all interfaces is not set
  Router ID 192.168.100.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    0.0.0.0 255.255.255.255 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.1.100        110      00:00:24
    192.168.100.1        110      00:00:24
  Distance: (default is 110)
```

In this example, the router's ID is 192.168.100.1. All interfaces are participating in OSPF (0.0.0.0 255.255.255.255) and are in area 0. There are two OSPF routers in this network: 192.168.1.100 (another router) and 192.168.100.1 (this router). Notice that the default administrative distance is 110.

11.04. The CD contains a multimedia demonstration of using the `show ip protocols` command on an OSPF router.

The `show ip route` Command

Your router keeps a list of the best paths to destinations in a routing table. To view the routing table, use the **`show ip route`** command:

```
Router# 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, * - candidate default,
       U - per-user static route, o - ODR,
       T - traffic engineered route

Gateway of last resort is not set
     10.0.0.0/24 is subnetted, 1 subnets
O       10.0.1.0 [110/65] via 192.168.1.100, 00:04:18, Serial0
C    192.168.1.0/24 is directly connected, Serial0
C    192.168.100.0/24 is directly connected, Ethernet0
```

OSPF routes show up as an `O` in the output of the `show ip route` command.

In this example, there is one OSPF route (O): 10.0.1.0. This route has an administrative distance of 110, has a cost of 65, and can be reached via neighbor 192.168.1.100.

11.05. The CD contains a multimedia demonstration of using the `show ip route` command on an OSPF router.

The `show ip ospf interface` Command

On an interface-by-interface basis, your OSPF router keeps track of what area an interface belongs to and what neighbors, if any, are connected to the interface. To view this, use the **`show ip ospf interface`** command:

```
Router# show ip ospf interface
Ethernet 1 is up, line protocol is up
Internet Address 172.16.255.1/24, Area 0
Process ID 100, Router ID 172.16.255.1, Network Type BROADCAST, Cost: 10
```

```
Transmit Delay is 1 sec, State DROTHER, Priority 1
Designated Router id 172.16.255.11, Interface address 172.16.255.11
Backup Designated router id 172.16.255.10, Interface addr 172.16.255.10
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 0:00:03
Neighbor Count is 3, Adjacent neighbor count is 2
  Adjacent with neighbor 172.16.255.10  (Backup Designated Router)
  Adjacent with neighbor 172.16.255.11  (Designated Router)
```

In this example, the router ID is 172.16.255.1. Its state is DROTHER, which means that it is *not* the DR or BDR. Actually, the DR is 172.16.255.11 and the BDR is 172.16.255.10. There are a total of three neighbors, with two adjacencies—remember that adjacencies are built only between routers and the DR and BDR.

CertCam

11.06. The CD contains a multimedia demonstration of using the `show ip ospf interface` command on an OSPF router.

exam watch

The `show ip ospf interface` command displays your router's ID, the ID of the DR and BDR, the hello timer (10 seconds), the dead interval (40 seconds), the number of neighbors, and the number of adjacencies.

The `show ip ospf neighbor` Command

To see all of your router's OSPF neighbors, use the **`show ip ospf neighbor`** command:

```
Router# show ip ospf neighbor
   ID           Pri   State         Dead Time     Address         Interface
172.16.255.11    1    FULL/DR        0:00:31    172.16.255.11     Ethernet0
172.16.255.10    1    FULL/BDR       0:00:33    172.16.255.10     Ethernet0
172.16.255.9     1    2WAY/DROTHER   0:00:35    172.16.255.9      Ethernet0
172.16.254.2     1    FULL/DR        0:00:39    172.16.254.2      Serial0.1
```

In this example, there are three routers connected to `Ethernet0`: 172.16.255.11 is a DR, 172.16.255.10 is a BDR, and 172.16.255.9 is another OSPF router (DROTHER). Notice that for the DR and the BDR, the state is *full*, which is to be expected, since this router and

exam watch

The `show ip ospf neighbor` command lists all of the router's OSPF neighbors, their OSPF states, their router IDs, and which interface the neighbors are connected to.

the DR/BDR share routing information with each other. The DROTHER router is in a *two-way* state, which indicates that the router is a neighbor, but this router and the DROTHER router will not share routing information directly with each other.

11.07. The CD contains a multimedia demonstration of using the `show ip ospf neighbor` command on an OSPF router.

The `debug ip ospf adj` Command

For more detailed troubleshooting, you can use **`debug`** commands. If you want to view the adjacency process that a router builds to other routers, use the **`debug ip ospf adj`** command:

```
Router# debug ip ospf adj
172.16.255.11 on Ethernet0, state 2WAY
OSPF: end of Wait on interface Ethernet0
OSPF: DR/BDR election on Ethernet0
OSPF: Elect BDR 172.16.255.10
OSPF: Elect DR 172.16.255.11
       DR: 172.16.255.11 (Id) BDR: 172.16.255.10 (Id)
OSPF: Send DBD to 172.16.255.11 on Ethernet0
       seq 0x10DB opt 0x2 flag 0x7 len 32
OSPF: Build router LSA for area 0, router ID 172.16.255.11
```

In this example, you can see the election process for the DR and BDR and the sharing of links (DBDs) with the DR.

11.08. The CD contains a multimedia demonstration of using the `debug ip ospf adj` command on an OSPF router.

The `debug ip ospf events` Command

If you want to view OSPF events on your router, use the **`debug ip ospf events`** command:

```
Router# debug ip ospf events
4d02h: OSPF: Rcv hello from 192.168.1.100 area 0 from Serial0
192.168.1.100
4d02h: OSPF: End of hello processing
```

In this example, the router received a hello packet from 192.168.1.00, which is connected to `Serial0`. Other kinds of information that you might see are:

- Hello intervals that do not match for routers on a segment

- Dead intervals that do not match for routers on a segment
- Mismatched subnet masks for OSPF routers on a segment

CertCam

11.09. The CD contains a multimedia demonstration of using the `debug ip ospf events` command on an OSPF router.

The `debug ip ospf packet` Command

If you want to view OSPF packet contents of LSAs, use the **`debug ip ospf packet`** command:

```
Router# debug ip ospf packet
4d02h: OSPF: rcv. v:2 t:1 l:48 rid:192.168.1.100
      aid:0.0.0.0 chk:15E4 aut:0 auk: from Serial0
```

exam watch

Be familiar with the terms in Table 11-2.

Table 11-2 explains the values shown in this command.

CertCam

11.10. The CD contains a multimedia demonstration of using the `debug ip ospf packet` command on an OSPF router.

TABLE 11-2 Field Values of the **`debug ip ospf packet`** Command

Field Value	Explanation
`Aid:`	OSPF Area ID number
`Auk:`	OSPF authentication key used for neighbor authentication
`Aut:`	Type of OSPF authentication (0-none, 1-simple password, 2-MD5 hashing)
`Keyid:`	MD5 key value if this authentication mechanism is enabled
`L:`	Length of the packet
`Rid:`	OSPF router ID
`Seq:`	Sequence number
`T:`	OSPF packet type (1-hello, 2-data description, 3-link state request, 4-link state update, 5-link state acknowledgment
`V:`	OSPF version number

EXERCISE 11-1

ON THE CD

Configuring OSPF

These last few sections dealt with the configuring OSPF on a router. This exercise will help you reinforce this material for setting up and troubleshooting OSPF. You'll perform this lab using Boson's NetSim™ simulator. This exercise has you set OSPF on the two routers (2600 and 2500). You can find a picture of the network diagram for Boson's NetSim™ simulator in the Introduction of this book. After starting up the simulator, click on the *LabNavigator* button. Next, double-click on *Exercise 11-1* and click on the *Load Lab* button. This will load the lab configuration based on Chapter 5's and 7's exercises.

1. On the 2600, verify that the `fa0/0` and `s0` interfaces are up. If not, bring them up. Examine the IP addresses configured on the 2600 and look at its routing table.

   At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2600*. On the 2600, use the **`show interfaces`** command to verify your configuration. If `fa0/0` and `s0` are not up, go into the interfaces (`fa0/0` and `s0`) and enable them: **`configure terminal`**, **`interface`** *`type port`*, **`no shutdown`**, **`end`**, **`show interfaces`**. Use the **`show ip route`** command. You should have two connected networks: 192.168.1.0 connected to `fa0/0` and 192.168.2.0 connected to `s0`.

2. On the 2500, verify that the `e0` and `s0` interfaces are up. If not, bring them up. Examine the IP addresses configured on the 2500 and look at its routing table.

   At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2500*. On the 2500, verify that the `e0` and `s0` interfaces are up. If not, bring them up: **`configure terminal`**, **`interface`** *`type port`*, **`no shutdown`**, **`end`**, **`show interfaces`**. Use the **`show interfaces`** command to verify that the IP addresses you configured on Chapter 5 are still there. Use the **`show ip route`** command. You should have two connected networks: 192.168.3.0 connected to e0 and 192.168.2.0 connected to s0.

3. Test connectivity between Host1 and the 2600. Test connectivity between Host3 and the 2500. Test connectivity between Host3 and Host1.

At the top of the simulator in the menu bar, click on the *eStations* icon and choose *Host1*. From Host1, ping the 2600: **`ping 192.168.1.1`**. The ping should be successful. At the top of the simulator in the menu bar, click on the *eStations* icon and choose *Host3*. From Host3, ping the 2500 router: **`ping 192.168.3.1`**. The ping should be successful. From Host3, ping Host 1: **`ping 192.168.1.10`**. The ping should fail: there is no route from the 2500 to this destination (look at the 2500's routing table: it doesn't list 192.168.1.0/24).

4. Enable OSPF on the 2600 and 2500 routers, using a process ID of 1, and put all interfaces in area 0.

   At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2600*. On the 2600 router, configure the following: **`configure terminal`**, **`router ospf 1`**, **`network 0.0.0.0 255.255.255.255 area 0`**, **`end`**. At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2600*. On the 2500 router, configure the following: **`configure terminal`**, **`router ospf 1`**, **`network 0.0.0.0 255.255.255.255 area 0`**, **`end`**.

5. On the 2600 and 2500, verify the operation of OSPF. Is either router a DR or BDR on the WAN link?

   At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2600*. Use the **`show ip protocols`** command to make sure that OSPF is configured—check for the neighboring router's update. Use the **`show ip route`** command and look for the remote LAN network number as a RIP (O) entry in the routing table. Use the **`show ip ospf neighbor`** command to view your neighboring router. Neither should be a DR or BDR on the serial link, since point-to-point connections don't use DRs and BDRs. At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2500*. Use the same above commands, **`show ip protocols`**, **`show ip route`**, and **`show ip ospf neighbor`**, to verify the operation of OSPF.

6. On Host1, test connectivity to Host3.

   At the top of the simulator in the menu bar, click on the *eStations* icon and choose *Host1*. On Host1, execute this: **`ping 192.168.3.2`**. The ping should be successful.

EXERCISE 11-2

ON THE CD

Troubleshooting OSPF

The last exercise dealt with configuring OSPF on the 2600 and 2500 routers. This exercise will help you introduce you to an already configured network, but with some configuration issues which are preventing OSPF connectivity. You'll perform this lab using Boson's NetSim™ simulator. You can find a picture of the network diagram for Boson's NetSim™ simulator in the Introduction of this book. After starting up the simulator, click on the *LabNavigator* button. Next, double-click on *Exercise 11-2* and click on the *Load Lab* button. This will load the lab configuration based on Chapter 5's and 7's exercises (with problems, of course.

Lets' start with your problem: Host1 cannot ping Host3. Your task is to figure out what the problems are and fix them: there are three problems. In this example, OSPF has been preconfigured on the routers. I would recommend that you try this troubleshooting process on your own first; and if you have problems, come back to the steps and solutions provided below.

1. Test connectivity from Host1 to Host3 with ping as well as from Host1 to its default gateway.

   At the top of the simulator in the menu bar, click on the *eStations* icon and choose *Host1*. On Host1, ping Host3: **ping 192.168.3.2**. Note that the ping fails. Ping the default gateway address: **ping 192.168.1.1**. The ping should fail, indicating that at least layer-3 is functioning between Host1 and the 2600. Examine the IP configuration on Host1 by executing: **winipcfg**. Make sure the IP addressing information is correct: IP address of 192.168.1.10, subnet mask of 255.255.255.0, and default gateway address of 192.168.1.1. Notice that the IP address is 192.168.100.10. Change this address to 192.168.1.10. Click on the *OK* button to save your changes and close **winipcfg**. Try pinging the 2600 again: **ping 192.168.1.1**. The ping should succeed. At the top of the simulator in the menu bar, click on the *eStations* icon and choose *Host1*. On Host1, ping Host3: **ping 192.168.3.2**. Note that the ping still fails.

2. Test connectivity from Host3 to its default gateway.

   At the top of the simulator in the menu bar, click on the *eStations* icon and choose *Host3*. Examine the IP configuration on Host3 by executing: **winipcfg**. Make sure the IP addressing information is correct: IP address of 192.168.3.2, subnet mask of 255.255.255.0, and default gateway address of 192.168.3.1. Click on the *Cancel* button to close **winipcfg**. Ping the default gateway address: **ping 192.168.3.1**. The ping should be fail, indicating that there is a problem between Host3 and the 2500. In this example, layer-2 is functioning correctly; therefore, it must be a problem with the 2500.

3. Check the interface statuses and IP configuration on the 2500 and verify connectivity to the 2600. Also verify OSPF's configuration.

   At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2500*. Check the status of the interfaces: **`show interfaces`**. Notice that the `e0` is has the wrong IP address (192.168.30.1) and is disabled. Go into `e0`, fix the IP address, and enable it: **`configure terminal`**, **`interface e0`**, **`ip address 192.168.3.1`**, **`no shutdown`**, **`end`**. Verify the status of the `e0 interface:` **`show interface e0`**. Try pinging Host3: **`ping 192.168.3.2`**. The ping should succeed. Try pinging the 2600's `serial0 interface:` **`ping 192.168.2.1`**. The ping succeeds. Examine the 2500's OSPF configuration: **`show ip protocol`**. You should see OSPF as the routing protocol and networks 192.168.2.0 and 192.168.3.0 included. From this output, it looks like OSPF is configured correctly on the 2500. Save the configuration on the router: **`copy running-config startup config`**.

4. Test connectivity from the 2500 to Host1. Examine the routing table.

   From the 2500 router, test the connection to Host1: **`ping 192.168.1.10`**. The ping should fail. This indicates a layer-3 problem between the 2500 and Host1. Examine the routing table: **`show ip route`**. Notice that there are only two connected routes (192.168.2.0/24 and 192.168.1.0/24), but no OSPF routes.

5. Access the 2600 router and examine OSPF's configuration. Fix the problem.

   At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2600*. Examine the routing table: **`show ip protocol`**. What networks are advertised by the 2600? You should see 192.168.100.0 and 192.168.2.0. Obviously, `fa0/0`'s interface isn't included since 192.168.1.0 is not configured. Fix this configuration problem: **`configure terminal`**, **`router ospf 1`**, **`no network 192.168.100.0 0.0.0.255 area 0`**, **`network 192.168.1.0 0.0.0.255 area 0`**, **`end`**. Test connectivity to Host3: **`ping 192.168.3.2`**. The ping should be successful. Save the configuration on the router: **`copy running-config startup config`**.

6. Examine the routing table on the 2500. Test connectivity from the 2500 to Host1.

   At the top of the simulator in the menu bar, click on the *eRouters* icon and choose *2500*. Examine the routing table: **`show ip route`**. Notice that there are only two connected routes (192.168.2.0/24 and 192.168.1.0/24) *and* one OSPF route (192.168.1.0/24). From the 2500 router, test the connection to Host1: **`ping 192.168.1.10`**. The ping should succeed.

7. Now test connectivity between Host1 and Host3.

   At the top of the simulator in the menu bar, click on the *eStations* icon and choose *Host1*. Test connectivity to Host3: **`ping 192.168.3.2`**. The ping should be successful.

In the next section, you will learn about EIGRP and how to configure it.

EIGRP

The Enhanced Interior Gateway Routing Protocol (EIGRP) is a Cisco-proprietary routing protocol for IP. It's actually based on IGRP, with many enhancements built into it. Because it has its roots in IGRP, the configuration is similar; however, it has many link state characteristics that were added to it to allow EIGRP to scale to enterprise network sizes. These characteristics include:

- Fast convergence
- Loop-free topology
- VLSM and route summarization
- Multicast and incremental updates
- Routes for multiple routed protocols

The following sections cover some of the characteristics of EIGRP, its operation, and its configuration.

Characteristics of EIGRP

Here is a brief comparison of EIGRP and IGRP:

- Both offer load balancing across six paths (equal or unequal).
- They have similar metric structures.
- EIGRP has faster convergence (triggered updates and saving a neighbor's routing table locally).
- EIGRP has less network overhead, since it uses incremental updates.

EIGRP and IGRP use the same metric structure. Both can use bandwidth, delay, reliability, and MTU when computing a best metric path to a destination. By default, only *bandwidth* and *delay* are used in the metric computation.

One interesting point about these protocols is that if you have some routers in your network running IGRP and others running EIGRP, and both sets have the same autonomous system number, routing information will *automatically* be shared between the two. The routers have to perform a conversion concerning the metrics. Even though both protocols use the same metric components, they store them in different size values: EIGRP uses a 32-bit metric, while IGRP uses a 24-bit metric. When integrating the two protocols together, EIGRP routes are divided by 256 to fit a 24-bit metric structure when passed to IGRP and IGRP routes are multiplied by 256 to fit a 32-bit metric structure when passed to EIGRP.

EIGRP uses the Diffusing Update Algorithm (DUAL) to update the routing table. This algorithm can enable very fast convergence by storing a neighbor's routing information in a local topology table. If a primary route in the routing table fails, DUAL can take a backup route from the topology table and place this into the routing table without necessarily having to talk to other EIGRP neighboring routers to find an alternative path to the destination.

Unlike IGRP, EIGRP supports both automatic and manual summarization. Remember that EIGRP is, at heart, a distance vector protocol, and therefore it will automatically summarize routes across Class A, B, and C network boundaries. You can also manually summarize within a class network, at your discretion. Configuration of summarization is beyond the scope of this book, but it is covered in depth on Cisco's BSCI CCNP and CCDP exams.

One really unique feature of EIGRP is that it supports three routed protocols: IP, IPX, and AppleTalk. In other words, EIGRP can route for all three of these protocols simultaneously. If you are running these routed protocols in your environment, EIGRP is a perfect fit. You only need to run one routing protocol for all three instead of a separate routing protocol for each, definitely reducing your routing overhead.

exam watch

The Cisco-proprietary EIGRP routing process uses the same metrics as IGRP. Unlike IGRP, EIGRP supports multicast and incremental updates, route summarization, and routing for IP, IPX, and AppleTalk. The DUAL algorithm is used to build a loop-free routing topology.

Interaction with Other EIGRP Routers

EIGRP uses hello packets to discover and maintain neighbor relationships, much as OSPF does. EIGRP generates hello packets every 5 seconds on LAN, point-to-point, and multipoint connections with speeds of at least T1/E1 speeds. Otherwise, hellos are generated every 60 seconds. The dead interval period is three times the hello interval. EIGRP uses the multicast address of 224.0.0.10 for the destination in the hello packets.

exam watch

Hello packets are generated every five seconds on LAN interfaces as multicasts (224.0.0.10).

For EIGRP routers to become neighbors, the following information must match:

- The AS number
- The K-values (these enable/disable the different metric components)

Unlike OSPF, the hello and hold-down timers on two routers do not need to match in order for the routers to become neighbors. When two routers determine whether they will become neighbors, they go through the following process:

1. The first router generates a Hello with configuration information.
2. If the configuration information matches, the second router responds with an Update message with topology information.
3. The first router responds with an ACK message, acknowledging the receipt of the second's ACK.
4. The first router sends its topology to the second router via an Update message.
5. The second router responds back with an ACK.

At this point, the two routers have converged. This process is different from OSPF, where routing information is disseminated via a designated router. With EIGRP, any router can share routing information with any other router. As you can see from the preceding steps, EIGRP, like OSPF, is connection-oriented: certain EIGRP messages sent by a router will cause it to expect an acknowledgment (ACK) from the destination(s). Here are the message types for which an EIGRP router expects an ACK back:

- **Update** Contains a routing update
- **Query** Asks a neighboring router to validate routing information
- **Reply** Responds to a query message

exam watch

Become familiar with the five EIGRP message types: hello, update, query, reply, and acknowledgment.

If an EIGRP router doesn't receive an ACK from these three packet types, the router will try a total of 16 times to resend the information. After this, the router declares the neighbor dead. When a router sends a hello packet, no corresponding ACK is expected.

Choosing Routes

EIGRP can use the following metric components when choosing a route: bandwidth, delay, reliability, load, and MTU. By default, however, only bandwidth and delay are activated. These are the K1 and K3 values. Table 11-3 explains important terms used by EIGRP.

EIGRP uses a less complicated approach than OSPF to choosing best-path routes to a destination and is less CPU intensive, but it requires more processing than IGRP or RIP. EIGRP routers keep topology information in a *topology table*, which contains the routes that neighbors are advertising, the advertised distances of the neighbor for these routers, and the feasible distances of this router to reach these destinations. A *successor*

TABLE 11-3 Important EIGRP Terms

Term	Definition
Neighbor table	This contains a list of the EIGRP neighbors and is similar to the adjacencies that are built in OSPF between the DR/BDR and the other routers on a segment. Each routed protocol for EIGRP has its own neighbor table.
Topology table	Similar to OSPF's database, this contains a list of all destinations and paths the EIGRP router learned—it is basically a compilation of the neighboring router's routing tables. There is a separate topology table for each routed protocol.
Successor	This is the best path to reach a destination within the topology table.
Feasible successor	This is the best backup path to reach a destination within the topology table—there can be multiple feasible successors for a particular destination.
Routing table	This is all of the successor routes from the topology table. There is a separate routing table for each routed protocol.
Advertised distance	This is the distance (metric) that a neighboring router is advertising for a specific route.
Feasible distance	This is the distance (metric) that your router will use to reach a specific route: the advertised distance plus the router's interface metric.

route is a path in the topology table that has the best metric compared to all the other alternative paths to the same destination. A *feasible successor* is a backup route to the successor route.

Not just any route can be chosen as a feasible successor. In order for a route to be considered as feasible successor in the topology table, the neighbor router's advertised distance must be *less than* that of the original route's feasible distance.

If a successor route in the routing table fails and a feasible successor exists in the topology table, the EIGRP router goes into a *passive* state—it immediately takes the feasible successor route from the topology table and puts it in the routing table, converging almost instantaneously. If the EIGRP router does not have a feasible successor in the topology table, it will go into an *active* state and generate a query packet for the route in question. This query is sent to the neighbor or neighbors that originally advertised this route.

The concern that EIGRP has with nonfeasible successor routes is that the path these routers are advertising might be part of a routing loop. EIGRP goes into an active state for these paths to verify this by double-checking with these neighbors. The neighbors will verify the information that they have in their topology table and reply to the requester with the appropriate information concerning these alternative paths. The terms *passive* and *active* can be misleading—passive means that a valid alternative route exists and can be used in the routing table, while active indicates that an alternative path exists but might or might not be valid.

Configuring EIGRP

Setting up EIGRP is almost as simple as configuring IGRP:

```
Router(config)# router eigrp autonomous_system_#
Router(config-router)# network IP_network_#
```

As you can see from these commands, EIGRP is configured the same as IGRP: you need to enter an autonomous system number and **`network`** statements for interfaces that will participate in EIGRP. Please note that the network numbers you specify are *classful* network numbers, even though EIGRP is *classless*.

exam watch

You must specify the AS number when configure EIGRP. Even though EIGRP is classless, you must configure it as a classful protocol when specifying your network numbers with the `network` command.

Other Commands

This section briefly touches on two other commands you should be aware of when configuring EIGRP. Since EIGRP uses bandwidth as a metric, make sure that your serial interfaces have the correct bandwidth value configured. Use the **bandwidth** *Interface Subconfiguration* mode command to configure this value. Remember to put the bandwidth value in Kbps. This command was covered in the section "OSPF" earlier in this chapter.

Also, EIGRP, like IGRP, supports load balancing across six paths. By default, EIGRP will do only equal-cost load balancing, like IGRP. However, you can enable *unequal*-cost load balancing of EIGRP routes by using the **variance** and **traffic-share** *Router Subconfiguration* mode commands. These commands were discussed in Chapter 10.

Configuration Example

Let's look at an example, shown previously in Figure 11-4, to help illustrate how to configure EIGRP on a router. Here's the routing configuration of the router:

```
Router(config)# router eigrp 200
Router(config-router)# network 172.16.0.0
Router(config-router)# network 10.0.0.0
```

This router has four interfaces: 172.16.1.1/24, 172.16.2.1/24, 10.1.1.1/24, and 10.1.2.1/24. Remember, when configuring your **network** commands, to put in only the Class A, B, or C network numbers. In the preceding example, the class B and A network numbers were entered, activating EIGRP routing on all four interfaces.

CertCam

11.11. The CD contains a multimedia demonstration of configuring EIGRP on a router.

Troubleshooting EIGRP

Here are some of the main commands you'll use when viewing and troubleshooting EIGRP:

- **show ip protocols**
- **show ip route**
- **show ip eigrp neighbors**
- **show ip eigrp topology**
- **show ip eigrp traffic**
- **debug ip eigrp**

The following sections cover these commands.

The **show ip protocols** Command

You can use the **show ip protocols** command to display the IP routing protocols that have been configured and are running on your router. Here is an example of this command for EIGRP:

```
Router# show ip protocols
Routing Protocol is "eigrp 200"
  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
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
  Redistributing: eigrp 200
  Automatic network summarization is in effect
  Automatic address summarization:
    10.0.0.0/8 for Serial0
  Maximum path: 4
  Routing for Networks:
    10.0.0.0
    192.168.4.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    (this router)         90      00:00:08
    192.168.4.101         90      00:00:06
  Distance: internal 90 external 170
```

In this command, you can see that the AS is 200 and the variance is 1 (only equal-cost load balancing). Two **network** statements were configured: 10.0.0.0 and 192.168.4.0. There is one neighboring router, 192.168.4.101. The administrative distance of internal EIGRP is 90.

CertCam

11.12. The CD contains a multimedia demonstration of the show ip protocols command for EIGRP on a router.

The **show ip route** Command

To view the EIGRP routes in your router's routing table, use the **show ip route** command:

```
Router# 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, * - candidate default,
              U - per-user static route, o - ODR,
              T - traffic engineered route

Gateway of last resort is not set
     10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       10.0.4.0/24 is directly connected, FastEthernet0
D    192.168.100.0/24 [90/2195456] via 192.168.4.101, 00:00:08, Serial0
C    192.168.4.0/24 is directly connected, Serial0
```

exam watch

EIGRP routes show up as `D` in the output of the `show ip route` command.

At the bottom of the display, a *D* in the first column refers to an EIGRP route. In this example, there is one EIGRP route that was learned from 192.168.4.101.

CertCam

11.13. The CD contains a multimedia demonstration of the `show ip route` command for EIGRP on a router.

The show ip eigrp neighbors Command

To view the list of EIGRP neighbors that your router has, use the **show ip eigrp neighbors** command:

```
Router# show ip eigrp neighbors
IP-EIGRP neighbors for process 200
Address         Interface   Hold  Uptime     SRTT  RTO   Q    Seq
                            (sec)            (ms)        Cnt  Num
192.168.4.101  Se0           13   00:02:10   610   3660  0     4
```

This example has one neighbor (192.168.4.101). Table 11-4 explains the output of this command.

CertCam

11.14. The CD contains a multimedia demonstration of the `show ip eigrp neighbors` command for EIGRP on a router.

TABLE 11-4 Fields from the **show ip eigrp neighbors** command

Field	Description
`Process`	AS number of the router
`Address`	IP address of the EIGRP neighbor
`Interface`	Your interface on which you are receiving the neighbor's hellos
`Hold`	The remaining time left before you declare your neighbor dead
`Uptime`	Length of time that you have known your neighbor
`SRTT (smooth round-trip time)`	The amount of time, in milliseconds, that it takes for your router to send EIGRP information to a neighbor and to get an ACK back
`RTO`	The amount of time, in milliseconds, that your router will wait before resending an EIGRP packet from the transmission queue to a neighbor
`Q Cnt`	The number of update/query/reply packets that you have queued up, ready to be sent
`Seq Num`	The sequence number of the last update/query/reply packet that your neighbor last sent

The show ip eigrp topology Command

To see the list of successor and feasible successors, as well as other types of routes, use the **show ip eigrp topology** command:

```
Router# show ip eigrp topology
IP-EIGRP Topology Table for AS(200)/ID(192.168.4.100)

Codes: P - Passive, A - Active, U - Update, Q - Query,
       R - Reply,r - Reply status
P 10.10.10.0 255.255.255.0, 2 successors, FD is 0
          via 10.10.1.1    (46251776/46226176), Ethernet0
          via 10.10.2.1    (46251776/46226176), Ethernet1
          via 10.10.1.3    (46277376/46251776), Ethernet0
```

In this example, there are two successor routes (the first two), but no feasible successor routes (FD = 0). Also notice that 10.10.10.0 is in a passive state, since it has two successor routes.

exam watch

Be able to pick out successor and feasible successor routes from the output of the* show ip eigrp topology *command.

CertCam

11.15. The CD contains a multimedia demonstration of the `show ip eigrp topology` command for EIGRP on a router.

The `show ip eigrp traffic` Command

To see information about traffic statistics for EIGRP, use the following command:

```
Router# show ip eigrp traffic
IP-EIGRP Traffic Statistics for process 200
  Hellos sent/received: 274/139
  Updates sent/received: 3/4
  Queries sent/received: 1/0
  Replies sent/received: 0/1
  Acks sent/received: 4/3
  Input queue high water mark 1, 0 drops
  SIA-Queries sent/received: 0/0
  SIA-Replies sent/received: 0/0
```

As you can see from this output, the router is sending and receiving hellos and updates and is sharing information with neighboring EIGRP routers.

CertCam

11.16. The CD contains a multimedia demonstration of the `show ip eigrp traffic` command for EIGRP on a router.

The `debug ip eigrp` Command

To troubleshoot EIGRP routing problems, you can use **`debug`** commands. The following command displays EIGRP events (there are other parameters available for this command):

```
Router# debug ip eigrp
1w0d: IP-EIGRP: 10.0.4.0/24 - don't advertise out Serial0
1w0d: IP-EIGRP: 192.168.4.0/24 - do advertise out Serial0
1w0d: IP-EIGRP: 10.0.0.0/8 - do advertise out Serial0
1w0d: IP-EIGRP: Int 10.0.0.0/8 metric 28160 - 25600 2560
1w0d: IP-EIGRP: Processing incoming UPDATE packet
1w0d: IP-EIGRP: Int 192.168.100.0/24 M 2195456 - 1657856
          537600 SM 281600 - 56000 25600
1w0d: IP-EIGRP: 192.168.100.0/24 routing table not updated
1w0d: IP-EIGRP: 10.0.4.0/24 - don't advertise out Serial0
1w0d: IP-EIGRP: 192.168.4.0/24 - do advertise out Serial0
1w0d: IP-EIGRP: 10.0.0.0/8 - do advertise out Serial0
1w0d: IP-EIGRP: Int 10.0.0.0/8 metric 28160 - 25600 2560
```

```
1w0d: IP-EIGRP: Processing incoming UPDATE packet
1w0d: IP-EIGRP: Int 10.0.0.0/8 M 4294967295 - 1657856
          4294967295 SM 4294967295 - 1657856 4294967295
```

exam watch

Be familiar with the use of the `debug ip eigrp` command.

In this example, I disabled and re-enabled `Serial0`. As you can see, it is advertising 192.168.4.0 to its neighbor connected to this interface.

CertCam

11.17. The CD contains a multimedia demonstration of the `debug ip eigrp` command for EIGRP on a router.

CERTIFICATION SUMMARY

OSPF is an open-standard routing protocol for IP, which uses cost as a metric. It uses the Dijkstra algorithm (SPF) to provide a loop-free routing topology and uses incremental updates with route summarization support. OSPF is hierarchical, supporting two layers: backbone (area 0) and areas connected to the backbone. Its downside is that OSPF requires more memory and CPU processes than distance vector protocols, as well as being difficult to configure and troubleshoot.

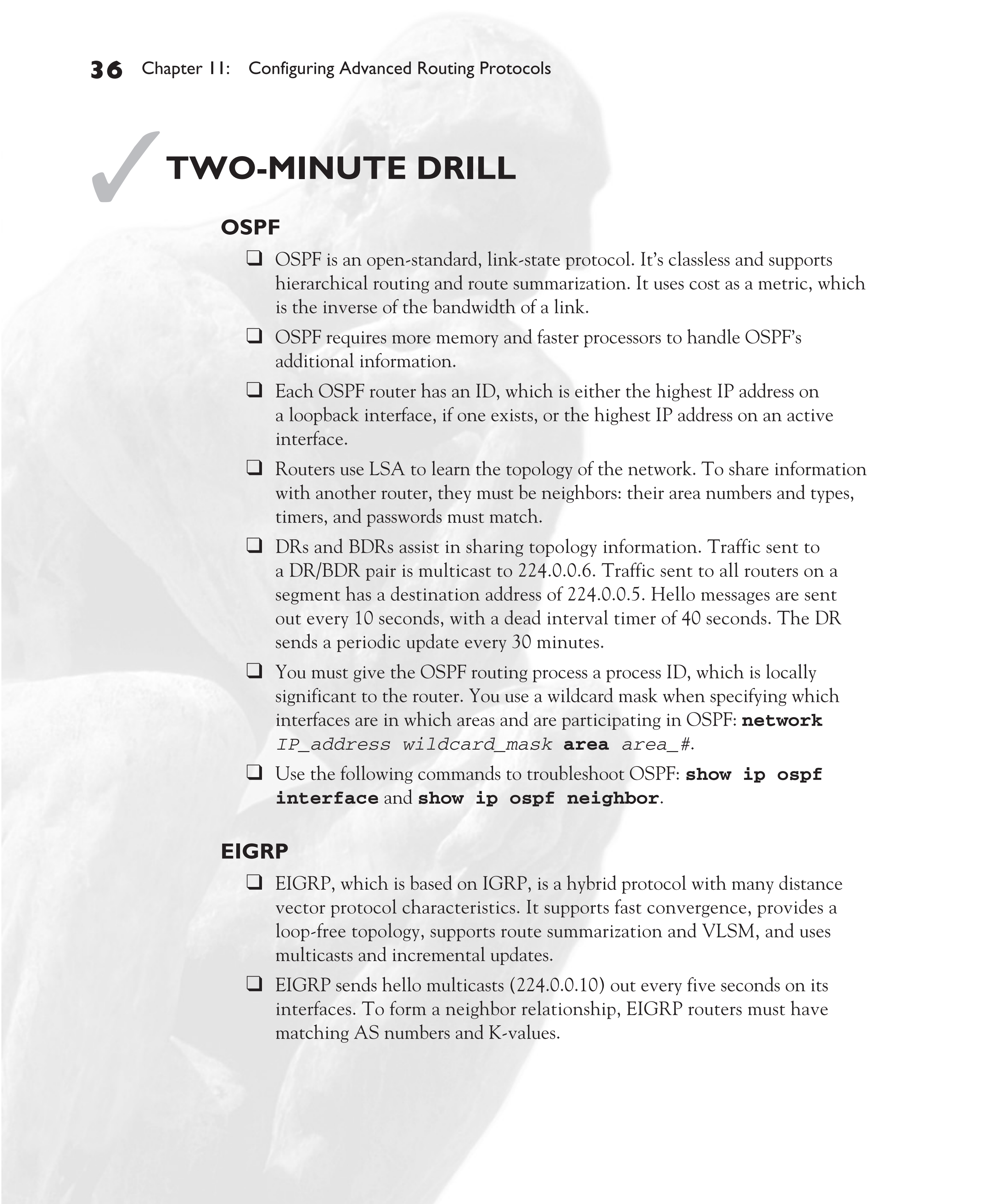
Each OSPF router has a router ID, which is either the highest IP address on a loopback interface or the highest IP address on an active interface. LSAs are used to develop neighbor relationships and are sent as multicasts every ten seconds. For LAN segments, a DR and a BDR are elected (highest router ID) to disseminate routing information. Routers use 224.0.0.6 to send information to the DR/BDR. OSPF is connection-oriented in that any routing information sent to another router requires a responding ACK. When DRs share routing information to their neighbors, the multicast it via 224.0.0.5.

Configuring OSPF requires you to specify a process ID, which is locally significant to the router. When configuring the **`network`** command, you specify an IP address or network number, a wildcard mask (inverted subnet mask), and a number for the area the address or network belongs to. The **`show ip ospf interface`** displays OSPF information about the router's ID, the DR and BDR, and timer information. The **`show ip ospf neighbor`** command displays your router's neighbors as well as their OSPF states.

EIGRP, a Cisco-proprietary protocol, is based on IGRP. Advancements of IGRP include fast convergence, a loop-free topology, route summarization, multicast and incremental updates, and routing for IP, IPX, and AppleTalk. Hellos are sent every five seconds as multicasts to develop and maintain a neighbor relationship. EIGRP's metrics are bandwidth, delay, reliability, load, and MTU.

The DUAL algorithm is used to provide a loop-free topology. This algorithm provides fast convergence by storing a neighbor's routing information locally in a topology table. The best path is called a successor route, and any valid alternative paths are called feasible successors. The advertised distance is a neighbor's metric to reach a destination, while the feasible distance is your router's metric to reach the same destination. There are five EIGRP messages: hello, update, query, reply, and acknowledgment.

Configuring EIGRP is the same as configuring IGRP: you must specify an AS number with the **`router`** command and you enter classful network numbers with the **`network`** command. The **`show ip eigrp topology`** command shows the topology table the DUAL algorithm uses to build the routing table. EIGRP routes show up as *D* in the IP routing table.



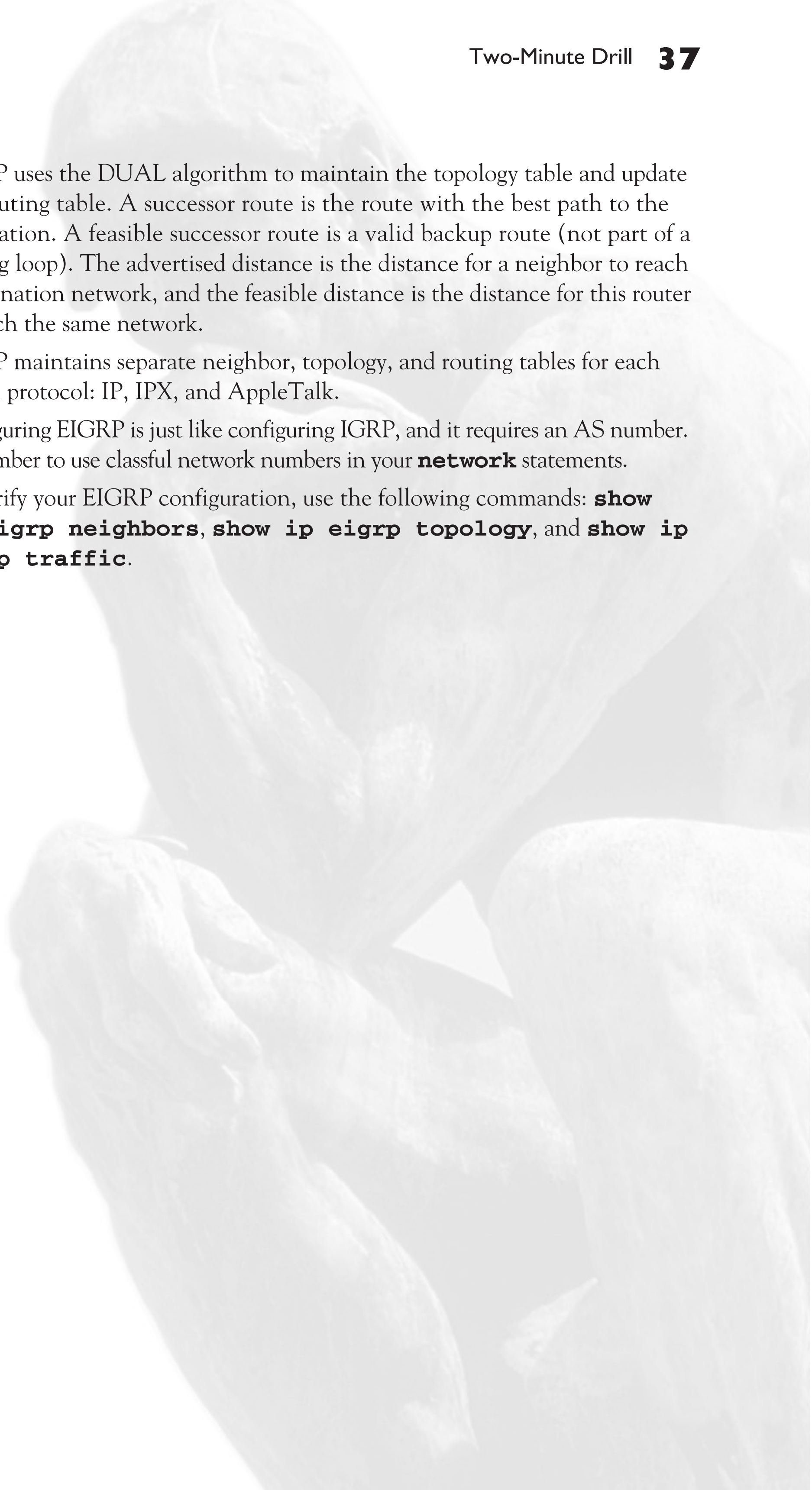
TWO-MINUTE DRILL

OSPF

- □ OSPF is an open-standard, link-state protocol. It's classless and supports hierarchical routing and route summarization. It uses cost as a metric, which is the inverse of the bandwidth of a link.
- □ OSPF requires more memory and faster processors to handle OSPF's additional information.
- □ Each OSPF router has an ID, which is either the highest IP address on a loopback interface, if one exists, or the highest IP address on an active interface.
- □ Routers use LSA to learn the topology of the network. To share information with another router, they must be neighbors: their area numbers and types, timers, and passwords must match.
- □ DRs and BDRs assist in sharing topology information. Traffic sent to a DR/BDR pair is multicast to 224.0.0.6. Traffic sent to all routers on a segment has a destination address of 224.0.0.5. Hello messages are sent out every 10 seconds, with a dead interval timer of 40 seconds. The DR sends a periodic update every 30 minutes.
- □ You must give the OSPF routing process a process ID, which is locally significant to the router. You use a wildcard mask when specifying which interfaces are in which areas and are participating in OSPF: **`network`** *`IP_address wildcard_mask`* **`area`** *`area_#`*.
- □ Use the following commands to troubleshoot OSPF: **`show ip ospf interface`** and **`show ip ospf neighbor`**.

EIGRP

- □ EIGRP, which is based on IGRP, is a hybrid protocol with many distance vector protocol characteristics. It supports fast convergence, provides a loop-free topology, supports route summarization and VLSM, and uses multicasts and incremental updates.
- □ EIGRP sends hello multicasts (224.0.0.10) out every five seconds on its interfaces. To form a neighbor relationship, EIGRP routers must have matching AS numbers and K-values.



- ❑ EIGRP uses the DUAL algorithm to maintain the topology table and update the routing table. A successor route is the route with the best path to the destination. A feasible successor route is a valid backup route (not part of a routing loop). The advertised distance is the distance for a neighbor to reach a destination network, and the feasible distance is the distance for this router to reach the same network.
- ❑ EIGRP maintains separate neighbor, topology, and routing tables for each routed protocol: IP, IPX, and AppleTalk.
- ❑ Configuring EIGRP is just like configuring IGRP, and it requires an AS number. Remember to use classful network numbers in your **`network`** statements.
- ❑ To verify your EIGRP configuration, use the following commands: **`show ip eigrp neighbors`**, **`show ip eigrp topology`**, and **`show ip eigrp traffic`**.

SELF TEST

The following Self Test questions will help you measure your understanding of the material presented in this chapter. Read all the choices carefully, as there may be more than one correct answer. Choose all correct answers for each question.

OSPF

1. Which of the following is false concerning OSPF?

   A. It provides a loop-free topology.

   B. It is a classful protocol and allows for a hierarchical design.

   C. It requires more memory and processing cycles than distance vector protocols.

   D. It is complex to configure and difficult to troubleshoot.

2. The OSPF process ID is __________.

   A. Locally significant and is the router ID

   B. Globally significant and must be configured on every router

   C. Locally significant

   D. OSPF doesn't use a process ID, but an AS number

3. Enter the OSPF command to include all of its interfaces in area 0: __________.

4. OSPF uses __________ as a metric.

   A. Bandwidth

   B. Delay

   C. Cost

   D. Hop count

5. An OSPF's router ID is based on __________.

   A. The lowest IP address on its loopback interface, if configured, or the lowest IP address on its active interfaces

   B. The highest IP address on its loopback interface, if configured, or the highest IP address on its active interfaces

   C. The highest IP address on its active interfaces, if configured, or the highest IP address on its loopback interfaces

   D. The lowest IP address on its active interfaces, if configured, or the lowest IP address on its loopback interfaces

6. OSPF hellos are sent every __________ seconds on a multi-access medium.

A. 5
B. 10
C. 15
D. 40

7. Enter the OSPF command to view all of a router's neighbors: __________.

8. OSPF supports a ________-layer hierarchical design.

A. 1
B. 2
C. 3
D. Multi

EIGRP

9. When examining the IP routing table, an EIGRP route will be shown as what letter?

A. I
B. E
C. O
D. D

10. EIGRP uses the _________ algorithm to update its routing table.

A. Bellman-Ford
B. Dijkstra
C. DUAL
D. Integrated

11. EIGRP generates hellos every _________ seconds on LAN segments.

A. 5
B. 10
C. 15
D. 30

12. EIGRP will route for ___________.

A. IP
B. IP and IPX
C. IP and AppleTalk
D. IP, IPX, and AppleTalk

13. A __________ route is the best path to reach a destination within the topology table.

A. Successor

B. Feasible successor

C. Advertised distance

D. Feasible distance

14. Enter the EIGRP command or commands to include the interfaces with 192.168.1.1/26, 192.168.1.64/26, and 192.168.1.128/26 in the routing process: __________.

15. Enter the EIGRP command to view only the successor routes: __________.

16. Enter the EIGRP command to view both the successor and feasible successor routes: _________.

SELF TEST ANSWERS

OSPF

1. ☑ **B.** OSPF is a classless, not a classful protocol.
☒ **A**, **C**, and **D** are true concerning OSPF.

2. ☑ **C.** The OSPF process ID is locally significant.
☒ **A** is not true, because the router ID is based on the highest IP address of a loopback or active interface. **B** is not true, because it is locally significant. **D** is not true, because OSPF requires a process ID to be configured.

3. **`SYMBOL 254 \f "Wingdings" \s 11  network 0.0.0.0 255.255.255.255 area 0`**.

4. ☑ **C.** OSPF uses bandwidth as a metric.
☒ **A** is used to compute the cost, which is the inverse of the bandwidth. **B** is used by IGRP and EIGRP. **D** is used by RIP.

5. ☑ **B.** An OSPF's router ID is based on the highest IP address on its loopback interface, if configured, or the highest IP address on its active interfaces.
☒ **A** is incorrect because specifies lowest. **C** is incorrect because the loopback is used first, if configured. **D** is incorrect because the loopback is checked first.

6. ☑ **B.** OSPF hellos are sent every ten seconds.
☒ **D** is the dead interval timer, and **A** and **C** are incorrect timers.

7. **`SYMBOL 254 \f "Wingdings" \s 11  show ip ospf neighbor`**.

8. ☑ **B.** OSPF supports a two-layer hierarchical design: the backbone (area 0) and areas off of the backbone.
☒ **A** is incorrect because this is a flat design. **C** is incorrect because only two layers are supported. **D** is true, but **B** is more correct.

EIGRP

9. ☑ **D.** A `D in the routing table indicates an EIGRP route.`
☒ **A** indicates an IGRP route. **B** indicates an EGP route. **C** is an OSPF route.

10. ☑ **C.** EIGRP uses the DUAL algorithm to update its routing table.
☒ **A** is incorrect because this is what distance vector protocols use. **B** is incorrect because this is what link state protocols use. **D** is a nonexistent routing algorithm.

11. ☑ **A.** EIGRP generates hellos every five seconds.
☒ **B**, **C**, and **D** are incorrect hello periods.

12. ☑ **D.** EIGRP supports three routed protocols: IP, IPX, and AppleTalk.
☒ **A** is incorrect because it omits IPX and AppleTalk. **B** is incorrect because it omits AppleTalk. **C** is incorrect because it omits IPX.

13. ☑ **A.** A successor route is the best path to reach a destination within the topology table.
☒ **B** is incorrect because this is a valid backup route. **C** refers to a neighbor's distance to a route. **D** refers to a router's distance to a route.

14. **`SYMBOL 254 \f "Wingdings" \s 11  network 192.168.1.0`**. Remember that this configuration is classful.

15. ☑ Use this command to view EIGRP's successor routes: **`show ip route`**.

16. ☑ Use this command to view both the successor and feasible successor routes: **`show ip eigrp topology`**.