CCNP BSCI
Quick Reference Sheets
Exam 642-901

Brent Stewart
Denise Donohue

The Evolving Network Model

EIGRP

OSPF

IS-IS

Optimizing Routing

BGP

IP Multicast

IPv6 Introduction

shortcut

Your Short Cut to Knowledge

ciscopress.com

About the Authors

Brent Stewart, CCNP, CCDP, MCSE, Certified Cisco Systems Instructor, is a network administrator for CommScope. He participated in the development of BSCI, and has seperately developed training material for ICND, BSCI, BCMSN, BCRAN, and CIT. Brent lives in Hickory, NC, with his wife, Karen and children, Benjamin, Kaitlyn, Madelyn, and William.

Denise Donohue, CCIE No. 9566, is manager of solutions engineering for ePlus Technology in Maryland. She is responsible for designing and implementing data and VoIP networks, supporting companies based in the National Capital region. Prior to this role, she was a systems engineer for the data consulting arm of SBC/AT&T. Denise was a Cisco instructor and course director for Global Knowledge and did network consulting for many years. Her CCIE is in Routing and Switching.

About the Technical Reviewers

Rus Healy, CCIE No. 15025, works as a senior engineer for Annese & Associates, a Cisco partner in Upstate New York. He also holds CCNP and CCDP certifications. His other interests include bicycling, skiing, and camping with his family, as well as competitive Amateur Radio events.

John Mistichelli, CCIE No. 7536, CCSI #20000, CCNP, CCDP, CCIP, MCSE, CNE, is a self-employed Cisco consultant and trainer. He provides network consulting services for businesses and government organizations throughout the United States. John is also a world class technical trainer for Convergent Communications where he teaches Service Provider courses for Cisco Advanced Services Education. John is a coauthor of the book *Cisco Routers 24Seven*.

Icons Used in This Book

The Evolving Network Model

The Hierarchical Design Model

Cisco used the three-level *Hierarchical Design Model* for years. This older model provided a high-level idea of how a reliable network might be conceived, but it was largely conceptual because it didn't provide specific guidance. Figure 1-1 shows the Hierarchical Design Model.

FIGURE 1-1 Hierarchical Design Model

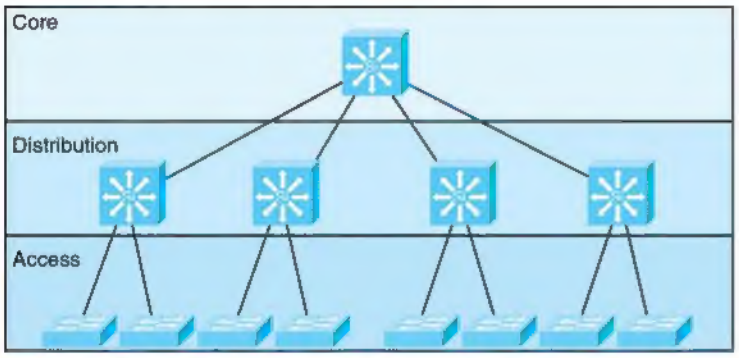
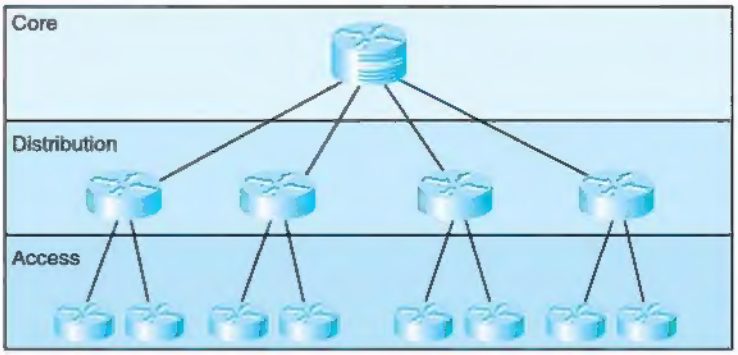


Figure 1-2 is a simple drawing of how the three-layer model might have been built out. A distribution layer-3 switch is used for each building on campus, tying together the access switches on the floors. The core switches link the various buildings together.

This same three-layer hierarchy can be used in the WAN with a central headquarters, division headquarters, and units.

FIGURE 1-2 Three-Layer Network Design



The layers break a network in the following way:

- Access layer—End stations attach to the network using low-cost devices.
- Distribution layer—Intermediate devices apply policies.
  - Route summarization
  - Policies applied, such as:
    - Route selection
    - Access lists
    - Quality of Service (QoS)

- Core layer—The backbone that provides a high-speed path between distribution elements.
  - Distribution devices are interconnected.
  - High speed (there is a lot of traffic).
  - No policies (it is tough enough to keep up).

Later versions of this model include redundant distribution, core devices, and connections, which make the model more fault-tolerant.

Problems with the Hierarchical Design Model

This early model was a good starting point, but it failed to address key issues, such as:

- Where do wireless devices fit in?
- How should Internet access and security be provisioned?
- How do you account for remote access, such as dial-up or VPN?
- Where should workgroup and enterprise services be located?

Enterprise Composite Network Model

The newer Cisco model—the Enterprise Composite Model—is significantly more complex and attempts to address the shortcomings of the Hierarchical Design Model by expanding the older version and making specific recommendations about how and where certain network functions should be implemented. This model is based on the principles described in the Cisco Architecture for Voice, Video, and Integrated Data (AVVID).

The Enterprise Composite Model (see Figure 1-3) is broken into three large sections:

- Enterprise Campus—Switches that make up a LAN
- Enterprise Edge—The portion of the enterprise network connected to the larger world.
- Service Provider Edge—The different public networks that are attached

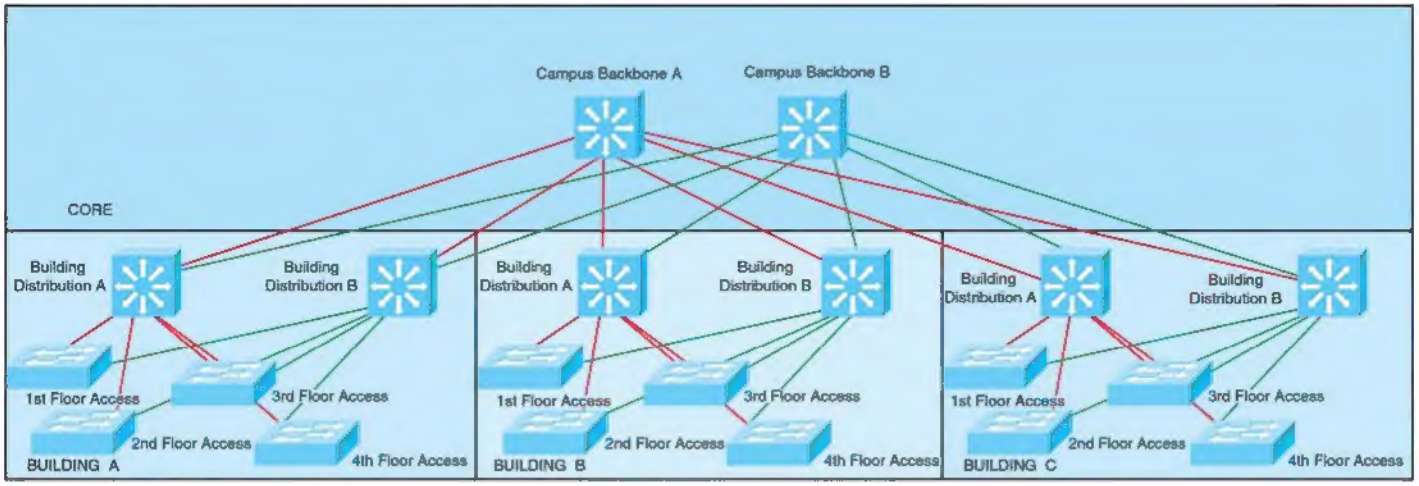
The first section, the Enterprise Campus, looks like the old Hierarchical Design Model with added details. It features six sections:

- Campus Backbone—The core of the LAN
- Building Distribution—Links subnets/VLANs and applies policy
- Building Access—Connects users to network
- Management
- Edge Distribution—A distribution layer out to the WAN
- Server Farm—For Enterprise services

CHAPTER 1

THE EVOLVING NETWORK MODEL

FIGURE 1-3 The Enterprise Composite Model



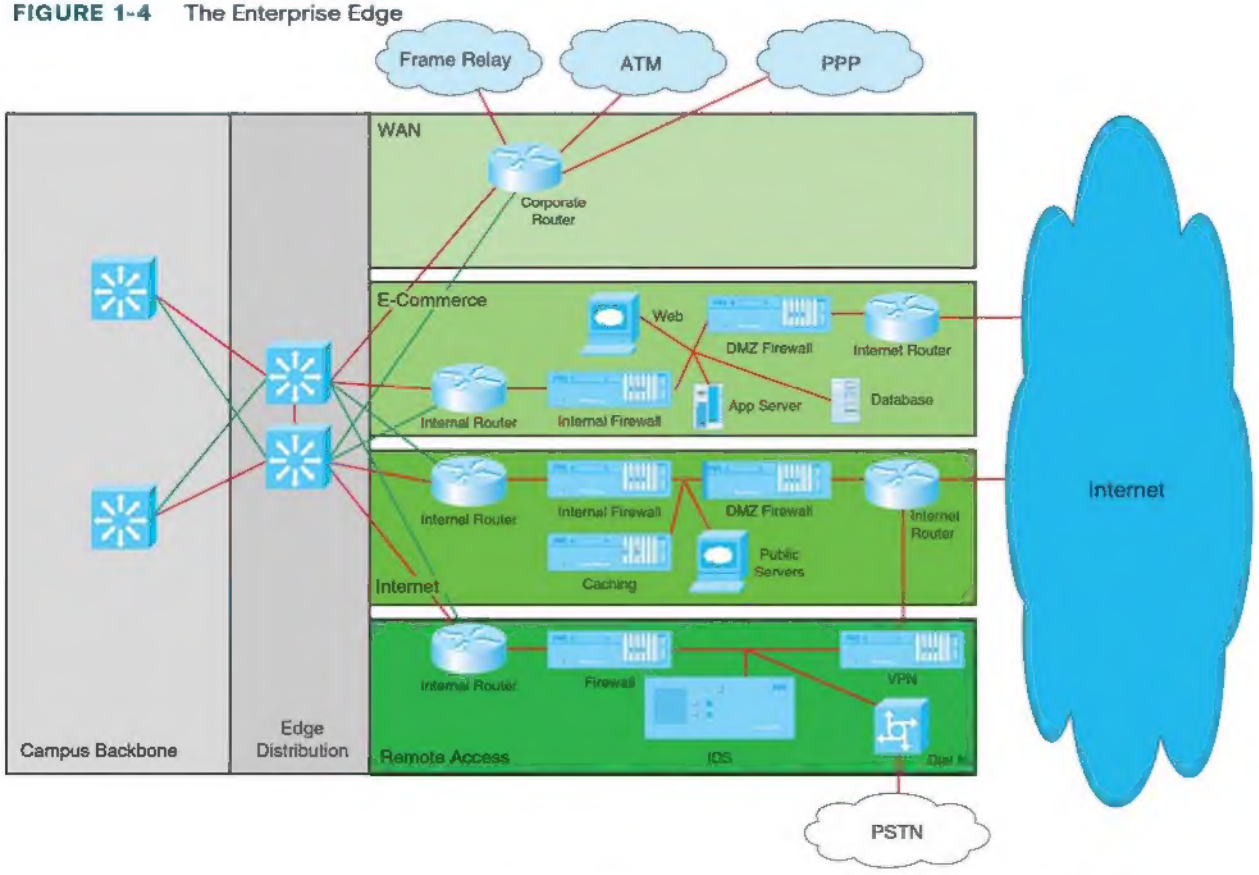
The Enterprise Edge, shown in Figure 1-4, details the connections from the campus to the WAN and includes:

- E-commerce
- Internet connectivity
- Remote access
- WAN

CHAPTER 1

THE EVOLVING NETWORK MODEL

FIGURE 1-4 The Enterprise Edge



Enterprise Edge **Service Provider Edge**

CHAPTER 1

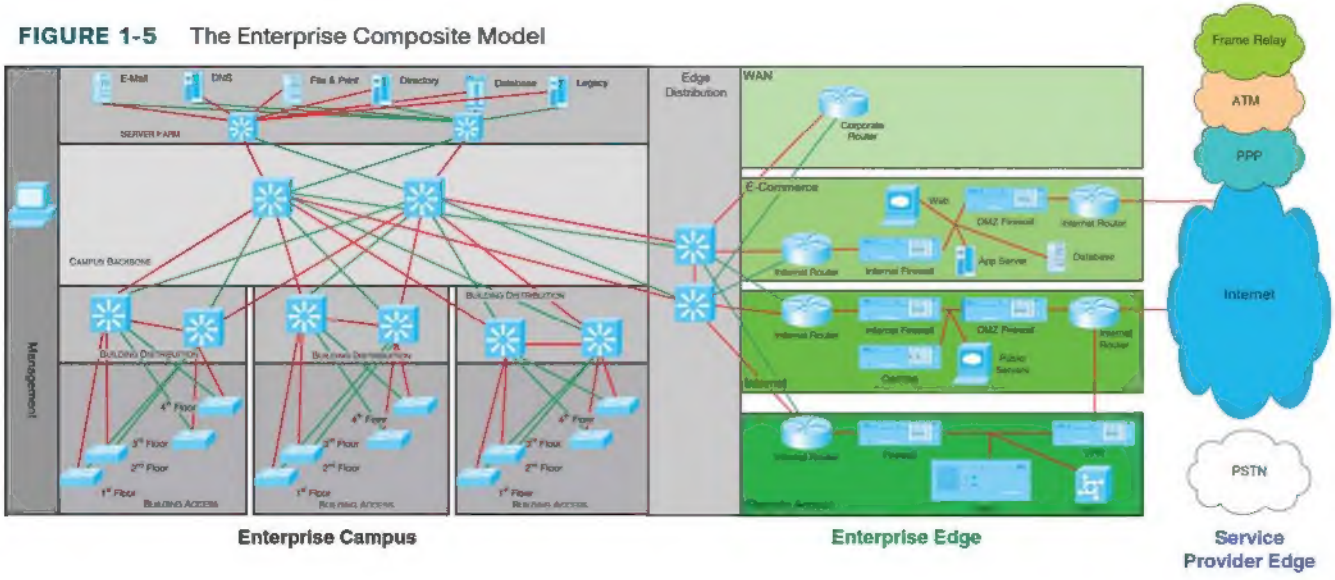
THE EVOLVING NETWORK MODEL

The Service Provider Edge is just a list of the public networks that facilitate wide-area connectivity and include:

- Internet service provider (ISP)
- Public switched telephone network (PSTN)
- Frame Relay, ATM, and PPP

Figure 1-5 puts together the various pieces: Campus, Enterprise Edge, and Service Provider Edge. Security implemented on this model is described in the Cisco SAFE (Security Architecture for Enterprise) blueprint.

FIGURE 1-5 The Enterprise Composite Model



CHAPTER 1

THE EVOLVING NETWORK MODEL

SONA and IIN

Modern converged networks include different traffic types, each with unique requirements for security, QoS, transmission capacity, and delay. These include:

- Voice signaling and bearer
- Core application traffic, such as Enterprise Resource Planning (ERP) or Customer Relationship Management (CRM)
- Database transactions
- Multicast multimedia
- Network management
- Other traffic, such as web pages, e-mail, and file transfer

Cisco routers are able to implement filtering, compression, prioritization, and policing. Except for filtering, these capabilities are referred to collectively as QoS.

Note

The best way to meet capacity requirements is to have twice as much bandwidth as needed. Financial reality, however, usually requires QoS instead.

Although QoS is wonderful, it is not the only way to address bandwidth shortage. Cisco espouses an idea called the Intelligent Information Network (IIN).

IIN describes an evolutionary vision of a network that integrates network and application functionality cooperatively and allows the network to be smart about how it handles traffic to minimize the footprint of applications. IIN is built on top of the Enterprise Composite Model and describes structures overlaid on to the Composite design as needed in three phases.

Phase 1, "Integrated Transport," describes a converged network, which is built along the lines of the Composite model and based on open standards. This is the phase that the industry has been transitioning to recently. The Cisco Integrated Services Routers (ISR) are an example of this trend.

Phase 2, "Integrated Services," attempts to virtualize resources, such as servers, storage, and network access. It is a move to an "on-demand" model.

By "virtualize," Cisco means that the services are not associated with a particular device or location. Instead, many services can reside in one device to ease management, or many devices can provide one service that is more reliable.

An ISR brings together routing, switching, voice, security, and wireless. It is an example of many services existing on one device. A load balancer, which makes many servers look like one, is an example of one service residing on many devices.

VRFs are an example of taking one resource and making it look like many. Some versions of IOS are capable of having a router present itself as many virtual router (VRF) instances, allowing your company to deliver different logical topologies on the same physical infrastructure. Server virtualization is another example. The classic example of taking one resource and making it appear to be many resources is the use of a virtual LAN (VLAN) and a virtual storage area network (VSAN).

CHAPTER 1

THE EVOLVING NETWORK MODEL

Virtualization provides flexibility in configuration and management.

Phase 3, "Integrated Applications," uses application-oriented networking (AON) to make the network application-aware and to allow the network to actively participate in service delivery.

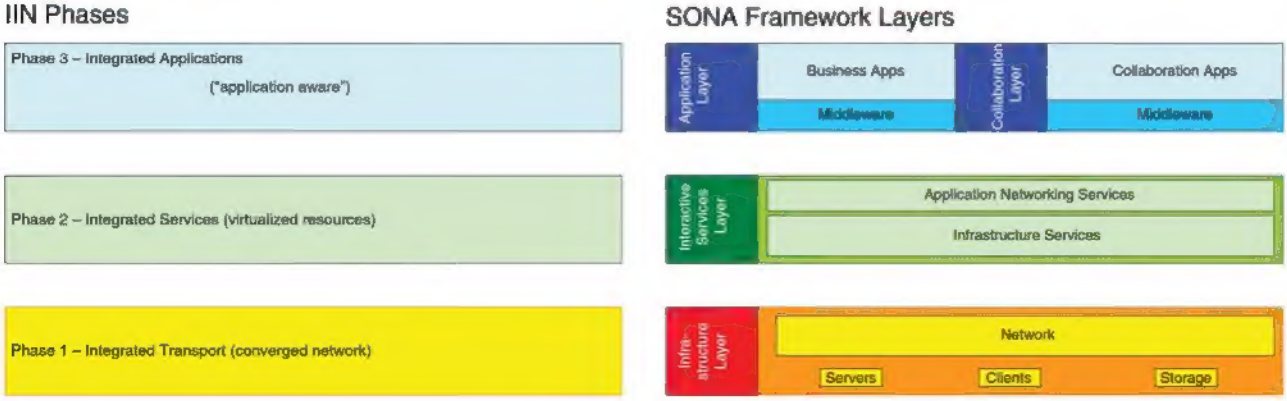
An example of this Phase 3 IIN systems approach to service delivery is Network Admission Control (NAC). Before NAC, authentication, VLAN assignment, and anti-virus updates were separately managed. With NAC in place, the network is able to check the policy stance of a client and admit, deny, or remediate based on policies.

IIN allows the network to deconstruct packets, parse fields, and take actions based on the values it finds. An ISR equipped with an AON blade might be set up to route traffic from a business partner. The AON blade can examine traffic, recognize the application, and rebuild XML files in memory. Corrupted XML fields might represent an attack (called *schema poisoning*), so the AON blade can react by blocking that source from further communication. In this example, routing, an awareness of the application data flow, and security are combined to allow the network to contribute to the success of the application.

Services-Oriented Network Architecture (SONA) applies the IIN ideal to Enterprise networks. SONA breaks down the IIN functions into three layers:

- Network Infrastructure—Hierarchical converged network and attached end systems.
- Interactive Services—Resources allocated to applications.
- Applications—Includes business policy and logic.

FIGURE 1-6 IIN and SONA



CHAPTER 1

THE EVOLVING NETWORK MODEL

IP Routing Protocols

Routing protocols are used to pass information about the structure of the network between routers. Cisco routers support the following IP routing protocols RIP (versions 1 and 2), IGRP, EIGRP, IS-IS, OSPF, and BGP. This section compares routing protocols and calls out key differences between them

Administrative Distance

Cisco routers are capable of supporting several IP routing protocols concurrently. When identical prefixes are discovered from two or more separate sources, Administrative Distance (AD) is used to discriminate between the paths. AD is a poor choice of words; *trustworthiness* is a better name. Routers use paths with the lower AD.

Table 1-1 lists the default values for various routing protocols. Of course, there are several ways to change AD for a routing protocol or for a specific route.

TABLE 1-1 Routing Protocols and Their Default Administrative Distance

Information Source	AD
Connected	0
Static	1
External BGP (Border Gateway Protocol)	20
Internal EIGRP (Enhanced IGRP)	90
IGRP (Internet Gateway Routing Protocol)	100
OSPF (Open Shortest Path First)	110
IS-IS (Intermediate System to Intermediate System)	115
RIP (Routing Information Protocol)	120
ODR (On Demand Routing)	160
External EIGRP	170
Internal BGP	200
Unknown	255

Building the Routing Table

The router builds a routing table by ruling out invalid routes and considering the remaining advertisements. The procedure is:

1. For each route received, verify the next hop. If invalid, discard the route.
2. If multiple, valid routes are advertised by a routing protocol, choose the lowest metric.
3. Routes are identical if they advertise the same prefix and mask, so 192.168.0.0/16 and 192.168.0.0/24 are separate paths and are each placed into the routing table.
4. If more than one specific valid route is advertised by different routing protocols, choose the path with the lowest AD.

CHAPTER 1

THE EVOLVING NETWORK MODEL

Comparing Routing Protocols

Two things should always be considered in choosing a routing protocol: fast convergence speed and support for VLSM. EIGRP, OSPF, and IS-IS meet these criteria. Although all three meet the minimum, there are still important distinctions, as described below:

- EIGRP is proprietary, but it is simple to configure and support.
- OSPF is an open standard, but it is difficult to implement and support.
- There are few books on IS-IS and even fewer engineers with experience who use it. IS-IS is therefore uncommon.

Table 1-2 compares routing protocols.

TABLE 1-2 Comparison of Routing Protocols

Property	EIGRP	OSPF	IS-IS	BGP
Method	Advanced distance vector	Link state	Link state	Path vector
Summary	Auto and arbitrary	Arbitrary	Arbitrary	Auto and arbitrary
VLSM	Yes	Yes	Yes	Yes
Converge	Seconds	Seconds	Seconds	Minutes
Timers, Update (hello/dead)	Triggered (LAN 5/15, WAN 60/180)	Triggered, but LSA refreshes every 30 minutes (NBMA 30/120, LAN 10/40)	Triggered (10/30)	Triggered (60/180)

CHAPTER 2

EIGRP

Enhanced Interior Gateway Routing Protocol (EIGRP) is a Cisco proprietary classless routing protocol that uses a complex metric based on bandwidth and delay. The following are some features of EIGRP:

- Fast convergence
- Support for VLSM
- Partial updates conserve network bandwidth
- Support for IP, AppleTalk, and IPX
- Support for all layer 2 (data link layer) protocols and topologies
- Sophisticated metric that supports unequal-metric proportional load-balancing
- Use of multicasts (and unicasts where appropriate) instead of broadcasts
- Support for authentication

EIGRP Overview

EIGRP's function is controlled by four key technologies:

- Neighbor discovery and maintenance—Uses periodic hello messages
- The Reliable Transport Protocol (RTP)—Controls sending, tracking, and acknowledging EIGRP messages
- Diffusing Update Algorithm (DUAL)—Determines the best loop-free route
- Protocol-independent modules (PDM)—Modules are "plug-ins" for IP, IPX, and AppleTalk versions of EIGRP

EIGRP uses three tables:

- The neighbor table is built from EIGRP hellos and used for reliable delivery.
- The topology table contains EIGRP routing information for best paths and loop-free alternatives.
- EIGRP places best routes from its topology table into the common routing table.

EIGRP Messages

EIGRP uses various message types to initiate and maintain neighbor relationships, and to maintain an accurate routing table. It is designed to conserve bandwidth and router resources by sending messages only when needed, and only to those neighbors that need to receive them.

CHAPTER 2

EIGRP

Packet Types

EIGRP uses five packet types:

- Hello—Identifies neighbors and serves as a keepalive mechanism
- Update—Reliably sends route information
- Query—Reliably requests specific route information
- Reply—Reliably responds to a query
- ACK—Acknowledgment

EIGRP is reliable, but hellos and ACKs are not acknowledged. The acknowledgement to a query is a reply.

If a reliable packet is not acknowledged, EIGRP periodically retransmits the packet to the nonresponding neighbor as a unicast. EIGRP has a window size of one, so no other traffic is sent to this neighbor until it responds. After 16 unacknowledged retransmissions, the neighbor is removed from the neighbor table.

Neighbor Discovery and Route Exchange

When EIGRP first starts, it uses hellos to build a neighbor table. Neighbors are directly attached routers that have a matching AS number and k values (the timers don't have to agree). The process of neighbor discovery and route exchange between two EIGRP routers is as follows:

Step 1. Router A sends out a hello.

Step 2. Router B sends back a hello and an update. The update contains routing information.

Step 3. Router A acknowledges the update.

Step 4. Router A sends its update.

Step 5. Router B acknowledges.

Once two routers are EIGRP neighbors, they use hellos between them as keepalives. Additional route information is sent only if a route is lost or a new route is discovered. A neighbor is considered lost if no hello is received within three hello periods (called the *hold time*). The default hello/hold timers are as follows:

- 5 seconds/15 seconds for multipoint circuits with bandwidth greater than T1 and for point-to-point media
- 60 seconds/180 seconds for multipoint circuits with bandwidth less than or equal to T1

The exchange process can be viewed using **debug ip eigrp packets**, and the update process can be seen using **debug ip eigrp**. The neighbor table can be seen with the command **show ip eigrp neighbors**.

CHAPTER 2

EIGRP

EIGRP Route Selection

An EIGRP router receives advertisements from each neighbor that lists the advertised distance (AD) and feasible distance (FD) to a route. The *AD* is the metric from the neighbor to the network. *FD* is the metric from this router, through the neighbor, to the network.

EIGRP Metric

The EIGRP metric is shown in Figure 2-1.

FIGURE 2-1 EIGRP Metric

$$metric = 256\left(k1 \times \frac{10^7}{BW_{min}} + \frac{k2 \times BW_{min}}{256 - load} + k3 \times \sum delays \times \frac{k5}{reliability + k4}\right)$$

The k values are constants. Their default values are: k1 = 1, k2 = 0, k3 = 1, k4 = 0, and k5 = 0. If k5 = 0, the final part of the equation (k5 / [rel + k4]) is ignored.

BW^{min} is the minimum bandwidth along the path—the choke point bandwidth.

Delay values are associated with each interface. The sum of the delays (in tens of microseconds) is used in the equation.

Taking the default k values into account, the equation simplifies to the one shown in Figure 2-2.

FIGURE 2-2 EIGRP Metric Simplified

$$metric = 256\left(\frac{10^7}{BW_{min}} + \sum delays\right)$$

If default k values are used, this works out to be 256 (BW + cumulative delay).

Bandwidth is the largest contributor to the metric. The delay value allows us to choose a more direct path when bandwidth is equivalent.

The EIGRP metric is 256 times the IGRP metric. The two automatically redistribute and algorithmically adjust metrics if they are configured on the same router for the same autonomous system.

Diffusing Update Algorithm (DUAL)

DUAL is the algorithm used by EIGRP to choose best paths by looking at AD and FD. The path with the lowest metric is called the *successor* path. EIGRP paths with a lower AD than the FD of the successor path are guaranteed loop-free and called *feasible successors*. If the successor path is lost, the router can use the feasible successor immediately without risk of loops.

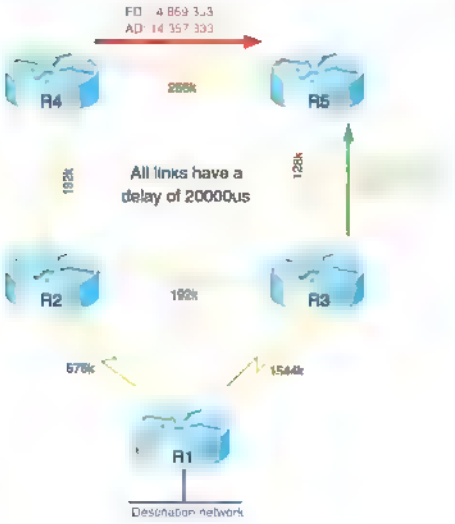
After the router has chosen a path to a network, it is *passive* for that route. If a successor path is lost and no feasible successor is identified, the router sends out queries on all interfaces in an attempt to identify an alternate path. It is *active* for that route. No successor can be chosen until the router receives a reply to all queries. If a reply is missing for

three minutes, the router becomes *stuck in active (SIA)*. In that case, it resets the neighbor relationship with the neighbor that did not reply.

Route Selection Example

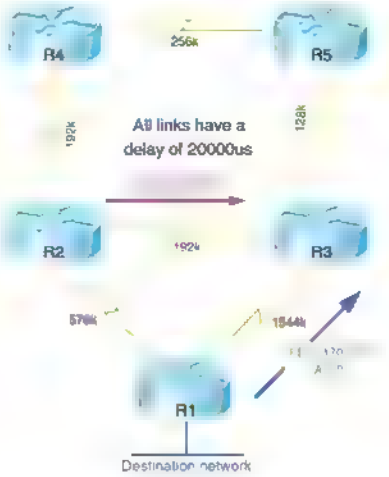
The following diagrams show EIGRP advertisements to R3 and R5 about a destination network connected to R1. In Figure 2-3, R5 chooses R4 as the successor path because it offers the lowest feasible distance. The AD from R3 indicates that passing traffic through R3 will not loop, so R3 is a feasible successor

FIGURE 2-3 EIGRP Path Selection, Part One



How does R3 choose its path? Figure 2-4 shows the path selection process for R3.

FIGURE 2-4 EIGRP Path Selection, Part Two



R1 will be its successor because it has the lowest metric. However, no feasible successor exists because R2's AD is greater than the successor path metric. If the direct path to R1 is lost, then R3 has to query its neighbors to discover an alternative path. It must wait to hear back from R2 and R5, and will ultimately decide that R2 is the new successor.

CHAPTER 2

EIGRP

Basic EIGRP Configuration

EIGRP is configured by entering router configuration mode and identifying the networks within which it should run. When setting up EIGRP, an autonomous system number must be used (7 is used in the example). Autonomous system numbers must agree for two routers to form a neighbor relationship and to exchange routes

```
Router(config)#router eigrp 7
Router(config-router)#network 192.168.1.0
```

The wildcard mask option can be used with the network command to more precisely identify EIGRP interfaces. For instance, if a router has two interfaces—fa0/0 (192.168.1.1/27) and fa0/1 (192.168.1.33/27)—and needs to run only EIGRP on fa0/0, the following command can be used:

```
Router(config-router)#network 192.168.1.0 0.0.0.1
```

In this command, a wildcard mask of 0.0.0.1 matches only two IP addresses in network 192.168.1.0–192.168.1.0 and 192.168.1.1. Therefore, only interface fa0/0 is included in EIGRP routing.

Creating an EIGRP Default Route

Figure 2-5 shows a simple two-router network. You can configure EIGRP on R1 to advertise a default route to R3 in three ways:

- R1 can specify a default network:

  ```
  R1(config)#ip default-network 10.0.0.0
  ```

  R3 now sees a default network with a next hop of R1.

- Produce a summary route:

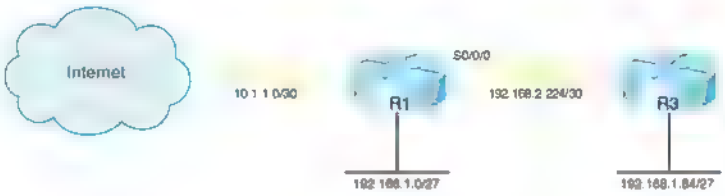
  ```
  R1(config)#interface s0/0/0
  R1(config-if)#ip summary-address eigrp 7 0.0.0.0 0.0.0.0
  ```

  This passes a default route from R1 out its serial0 interface toward R3.

- Create a static default route and then include network 0.0.0.0 in EIGRP:

  ```
  R1(config)#ip route 0.0.0.0 0.0.0.0 10.1.1.2
  R1(config)#router eigrp 7
  R1(config-router)#network 0.0.0.0
  ```

FIGURE 2-5 EIGRP Default Route



CHAPTER 2

EIGRP

Troubleshooting EIGRP

The most straightforward way to troubleshoot EIGRP is to inspect the routing table—**show ip route**. To filter the routing table and show only the routes learned from EIGRP, use the **show ip route eigrp** command. The **show ip protocols** command verifies autonomous system, timer values, identified networks, and EIGRP neighbors (routing information sources)

The command **show ip eigrp topology** shows the EIGRP topology table and identifies successors and feasible successors. Use **show ip eigrp neighbors** to verify that the correct routers are neighbors, and use **show ip eigrp traffic** to show the amount and types of EIGRP messages.

Advanced EIGRP Configuration

EIGRP provides some ways to customize its operation, such as route summarization, unequal-metric load balancing, controlling the percent of interface bandwidth used, and authentication. This section describes how to configure these.

Summarization

EIGRP defaults to automatically summarizing at classful network boundaries. Automatic summarization is usually disabled using the following command:

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

Summaries can be produced manually on any interface. When a summary is produced, a matching route to null0 also becomes active as a loop prevention mechanism. Configure a summary route out a particular interface using the **ip summary-address eigrp** *autonomous_system* command. The following example advertises a default route out FastEthernet0/1 and the summary route 172.16.104.0/22 out Serial0/0/0 for EIGRP AS 7.

```
Router(config)#int fa0/1
Router(config-if)#ip summary-address eigrp 7 0.0.0.0 0.0.0.0
!
Router(config)#int s0/0/0
Router(config-if)#ip summary-address eigrp 7 172.16.104.0
 255.255.252.0
```

Load Balancing

EIGRP, like most IP routing protocols, automatically load balances over equal metric paths. What makes EIGRP unique is that you can configure it to proportionally load balance over *unequal* metric paths. The **variance** command is used to configure load balancing over up to six loop-free paths with a metric lower than the product of the variance and the best metric. Figure 2-3, in the "Route Selection Example" section, shows routers advertising a path to the network connected to R1.

By default, R5 uses the path through R4 because it offers the lowest metric (14,869,333). To set up unequal cost load balancing, assign a variance of 2 under the EIGRP process on R5. R5 multiplies the best metric of 14,869,333 by 2, to get 29,738,666. R5 then uses all loop-free

paths with a metric less than 29,738,666, which includes the path through R3. By default, R5 load balances over these paths, sending traffic along each path in proportion to its metric.

```
R5(config)#router eigrp 7
R5(config-router)#variance 2
```

WAN Bandwidth

By default, EIGRP limits itself to bursting to half the link bandwidth. This limit is configurable per interface using the **ip bandwidth-percent** command. The following example assumes EIGRP AS 7 and limits EIGRP to one quarter of the link bandwidth:

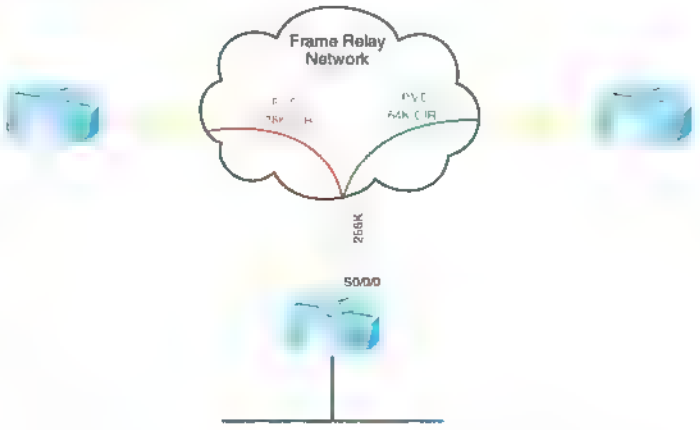
```
Router(config)#int s0/0/0
Router(config-if)#ip bandwidth-percent eigrp 7 25
```

The real issue with WAN links is that the router assumes that each link has 1544 kbps bandwidth. If interface Serial0/0/0 is attached to a 128 k fractional T1, EIGRP assumes it can burst to 768 k and could overwhelm the line. This is rectified by correctly identifying link bandwidth.

```
Router (config)#int serial 0/0/0
Router (config-if)#bandwidth 128
```

Figure 2-6 shows a situation in which these techniques can be combined—Frame Relay.

FIGURE 2-6 EIGRP with Frame Relay



In this example, R1 has a 256 kbps connection to the Frame Relay network and two permanent virtual circuits (PVCs) with committed information rates (CIR) of 128 Kpbs and 64 Kbps. EIGRP divides the interface bandwidth evenly between the number of neighbors on that interface. What value should be used for the interface bandwidth in this case? The usual suggestion is to use the CIR, but the two PVCs have different CIRs. You could use the bandwidth-percent command to allow SNMP reporting of the true bandwidth value, while adjusting the interface burst rate to 25 percent, or 64 kbps.

CHAPTER 2

EIGRP

```
R1(config)#int serial 0/0/0
R1 (config-if)#bandwidth 256
R1 (config-if)#ip bandwidth-percent eigrp 7 25
```

A better solution is to use subinterfaces and identify bandwidth separately In the following example, s0/0/0.1 bursts to 64 k, and s0/0/0.2 bursts to 32 k, using EIGRP's default value of half the bandwidth.

```
R1(config)#int serial 0/0/0.1
R1 (config if)#bandwidth 128
!
R1(config)#int serial 0/0/0.2
R1 (config-if)#bandwidth 64
```

In cases where the hub interface bandwidth is oversubscribed, it may be necessary to set bandwidth for each subinterface arbitrarily low, and then specify an EIGRP bandwidth percent value over 100 in order to allow EIGRP to use half the PVC bandwidth.

EIGRP Authentication

By default, no authentication is used for any routing protocol. Some protocols, such as RIPv2, IS-IS, and OSPF, can be configured to do simple password authentication between neighboring routers. In this type of authentication, a clear-text password is used. EIGRP does not support simple authentication. However, it can be configured to authenticate each packet exchanged, using an MD5 hash. This is more secure than clear text, as only the message digest is exchanged, not the password.

EIGRP authenticates each of its packets by including the hash in each one. This helps verify the source of each routing update.

To configure EIGRP authentication, follow these steps:

Step 1. Configure a key chain to group the keys.

Step 2. Configure a key within that key chain.

Step 3. Configure the password or authentication string for that key. Repeat Steps 2 and 3 to add more keys if desired.

Step 4. Optionally configure a lifetime for the keys within that key chain. If you do this, be sure that the time is synchronized between the two routers

Step 5. Enable authentication and assign a key chain to an interface.

Step 6. Designate MD5 as the type of authentication.

Example 2-1 shows a router configured with EIGRP authentication. It shows configuring a lifetime for packets sent using key 1 that starts at 10:15 and lasts for 300 seconds. It also shows configuring a lifetime for packets received using key 1 that starts at 10:00 and lasts until 10:05.

EXAMPLE 2-1 Configuring EIGRP Authentication

```
Router(config)#key chain RTR_Auth
Router(config-keychain)#key 1
Router(config-keychain-key)#key-string mykey
Router(config-keychain-key)#send-lifetime 10:15:00 300
Router(config keychain key)#accept-lifetime 10:00:00 10:05:00
!
Router(config)#interface s0/0/0
Router(config-if)#ip authentication mode eigrp 10 md5
Router(config-if)#ip authentication key-chain eigrp 10 RTR_Auth
```

CHAPTER 2

EIGRP

Verify your configuration with the **show ip eigrp neighbors** command, as no neighbor relationship will be formed if authentication fails. Using the **debug eigrp packets** command should show packets containing authentication information sent and received, and it will allow you to troubleshoot configuration issues.

EIGRP Scalability

Four factors influence EIGRP's scalability:

- The number of routes that must be exchanged
- The number of routers that must know of a topology change
- The number of alternate routes to a network
- The number of hops from one end of the network to the other

To improve scalability, summarize routes when possible, try to have a network depth of no more than seven hops, and limit the scope of EIGRP queries.

Stub routing is one way to limit queries. A stub router is one that is connected to no more than two neighbors and should never be a transit router. When a router is configured as an EIGRP stub, it notifies its neighbors. The neighbors then do not query that router for a lost route. Under router configuration mode, use the command **eigrp stub [receive-only|connected|static|summary]**. An EIGRP stub router still receives all routes from its neighbors by default.

Routers use *SIA Queries* and *SIA Replies* to prevent loss of a neighbor unnecessarily during SIA conditions. A router sends its neighbor a SIA-Query after no reply to a normal query. If the neighbor responds with a SIA-Reply, then the router does not terminate the neighbor relationship after three minutes, because it knows the neighbor is available.

Graceful shutdown is another feature that speeds network convergence. Whenever the EIGRP process is shut down, the router sends a "goodbye" message to its neighbors. The neighbors can then immediately recalculate any paths that used the router as the next hop, rather than waiting for the hold timer to expire.

CHAPTER 3

OSPF

OSPF Overview

OSPF is an open-standard, classless routing protocol that converges quickly and uses cost as a metric (Cisco IOS automatically associates cost with bandwidth).

OSPF is a link-state routing protocol and uses Dijkstra's Shortest Path First (SPF) algorithm to determine its best path to each network. The first responsibility of a link-state router is to create a database that reflects the structure of the network. Link state routing protocols learn more information on the structure of the network than other routing protocols, and thus are able to make more informed routing decisions.

OSPF routers exchange hellos with each neighbor, learning Router ID (RID) and cost. Neighbor information is kept in the adjacency database.

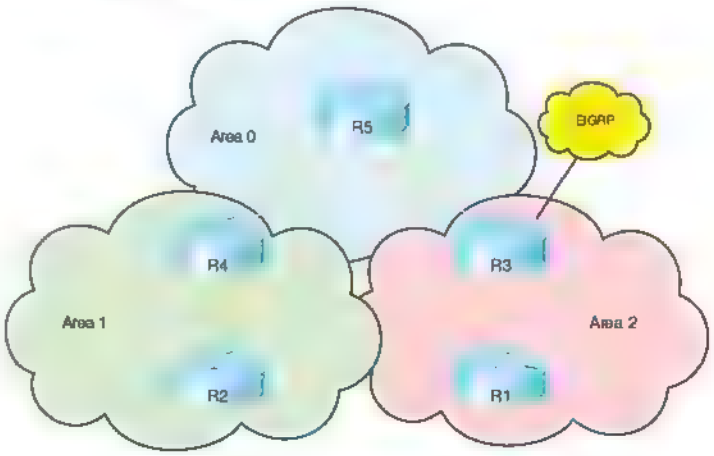
The router then constructs the appropriate Link State Advertisements (LSA), which include information such as the RIDs of, and cost to, each neighbor. Each router in the routing domain shares its LSAs with all other routers. Each router keeps the complete set of LSAs in a table—the Link State Database (LSDB).

Each router runs the SPF algorithm to compute best paths. It then submits these paths for inclusion in the routing table, or forwarding database.

OSPF Network Structure

OSPF routing domains are broken up into areas. An OSPF network must contain an area 0, and may contain other areas. The SPF algorithm runs within an area, and inter-area routes are passed between areas. A two-level hierarchy to OSPF areas exists: area 0 is designed as a transit area, and other areas should be attached directly to area 0 and only to area 0. The link-state database must be identical for each router in an area. OSPF areas typically contain a maximum of 50–100 routers, depending on network volatility. Figure 3-1 shows a network of five routers that has been divided into three areas: area 0, area 1, and area 2.

FIGURE 3-1 OSPF Areas



CHAPTER 3

OSPF

Dividing an OSPF network into areas does the following:

- Minimizes the number of routing table entries.
- Contains LSA flooding to a reasonable area.
- Minimizes the impact of a topology change.
- Enforces the concept of a hierarchical network design.

OSPF defines router roles as well. One router can have multiple roles.

- An internal router has all interfaces in one area. In Figure 3-1, R1, R2, and R5 are all internal area routers.
- Backbone routers have at least one interface assigned to area 0. R3, R4, and R5 are backbone routers.
- An Area Border Router (ABR) has interfaces in two or more areas. In Figure 3-1, R3 and R4 are ABRs.
- An Autonomous System Boundary Router (ASBR) has interfaces inside and outside the OSPF routing domain. In Figure 3-1, R3 also functions as an ASBR because it has an interface in an EIGRP routing domain.

OSPF Metric

By default, Cisco assigns a cost to each interface that is inversely proportional to 100 Mbps. The cost for each link is then accrued as the route advertisement for that link traverses the network. Figure 3-2 shows the default OSPF formula.

FIGURE 3-2 OSPF Cost Formula

$$Cost = \frac{100\ Mbps}{Bandwidth}$$

The default formula doesn't differentiate between interfaces with speeds faster than 100 Mbps. It assigns the same cost to a Fast Ethernet interface and a Gigabit Ethernet interface, for example. In such cases, the cost formula can be adjusted using the **auto-cost** command under the OSPF routing process. Values for bandwidth (in kbps) up to 4,294,967 are permitted (1 Gbps is shown in the following line):

```
Router(config-router)#auto-cost reference-bandwidth 1000
```

The cost can also be manually assigned under the interface configuration mode. The cost is a 16-bit number, so it can be any value from 1 to 65,535.

```
Router(config-router)#ip ospf cost 27
```

LSAs

Each router maintains a database of the latest received LSAs. Each LSA is numbered with a sequence number, and a timer is run to age out old LSAs.

When a LSA is received, it's compared to the LSDB. If it is new, it is added to the database and the SPF algorithm is run. If it is from a Router ID that is already in the database, then the sequence number is compared, and older LSAs are discarded. If it is a new LSA, it is incorporated in the database, and the SPF algorithm is run. If it is an older LSA, the newer LSA in memory is sent back to whoever sent the old one.

CHAPTER 3

OSPF

OSPF sequence numbers are 32 bits. The first legal sequence number is 0x80000001. Larger numbers are more recent. The sequence number changes only under two conditions:

- The LSA changes because a route is added or deleted.
- The LSA ages out (LSAs are updated every half hour, even if nothing changes).

The command **show ip ospf database** shows the age (in seconds) and sequence number for each RID.

LSDB Overload Protection

Because each router sends an LSA for each link, routers in large networks may receive—and must process—numerous LSAs. This can tax the router's CPU and memory resources, and adversely affect its other functions. You can protect your router by configuring OSPF LSDB overload protection. LDSB overload protection monitors the number of LSAs received and placed into the LSDB. If the specified threshold is exceeded for one minute, the router enters the "ignore" state by dropping all adjacencies and clearing the OSPF database. The router resumes OSPF operations after things have been normal for a specified period. Be careful when using this command, as it disrupts routing when invoked.

Configure LSDB overload protection with the OSPF router process command **max-lsa** *maximum-number* [*threshold-percentage*] [**warningonly**][**ignore-time** *minutes*] [**ignore-count** *number*] [**reset-time** *minutes*]. The meaning of the keywords of this command are:

- *Maximum-number*—The threshold. This is the most nonlocal LSAs that the router can maintain in its LSDB
- *Threshold-percentage*—A warning message is sent when this percentage of the threshold number is reached. The default is 75 percent.
- **Warningonly**—This causes the router to send only a warning; it does not enter the ignore state.
- **Ignore-time** *minutes*—Specifies the length of time to stay in the ignore state. The default is five minutes.
- **Ignore-count** *number*—Specifies the maximum number of times a router can go into the ignore state. When this number is exceeded, OSPF processing stays down and must be manually restarted. The default is five times.
- **Reset-time** *minutes*—The length of time to stay in the ignore state. The default is ten minutes.

LSA Types

OSPF uses different types of LSAs to advertise different types of routes, such as internal area or external routing domain. Many of these are represented in the routing table with a distinctive prefix. Table 3-1 describes these LSA types.

CHAPTER 3

OSPF

TABLE 3-1 OSPF LSA Types

Type	Description	Routing Table Symbol
1	Router LSA. Advertises intra-area routes. Generated by each OSPF router. Flooded only within the area.	O
2	Network LSA. Advertises routers on a multi-access link. Generated by a DR. Flooded only within the area.	O
3	Summary LSA. Advertises inter-area routes. Generated by an ABR. Flooded to adjacent areas.	O IA
4	Summary LSA. Advertises the route to an ASBR. Generated by an ABR. Flooded to adjacent areas.	O IA
5	External LSA. Advertises routes in another routing domain. Generated by an ASBR. Flooded to adjacent areas	O E1—The metric increases as it is passed through the network
		O E2—The metric does not increase (default).
6	Multicast LSA. Used in multicast OSPF operations.	
7	Not-so-stubby area (NSSA) LSA. Advertises routes in another routing domain. Generated by an ASBR within a not-so-stubby area.	O N1—The metric increases as it is passed through the network.
		O N2—The metric does not increase (default)
8	External attributes LSA. Used in OSPF and BGP interworking.	
9, 10, 11	Opaque LSAs. Used for specific applications, such as OSPF and MPLS interworking.	

CHAPTER 3

OSPF

OSPF Operation

OSPF uses several different message types to establish and maintain its neighbor relationships, and to maintain correct routing information. When preparing for the exam, be sure you understand each OSPF packet type, and the OSPF neighbor establishment procedure.

OSPF Packets

OSPF uses five packet types. It does not use UDP or TCP for transmitting its packets. Instead, it runs directly over IP (IP protocol 89) using an OSPF header. One field in this header identifies the type of packet being carried. The five OSPF packet types are:

- **Hello**—Identifies neighbors and serves as a keepalive.
- **Link State Request (LSR)**—A request for an Link State Update (LSU). Contains the type of LSU requested and the ID of the router requesting it.
- **Database Description (DBD)**—A summary of the LSDB, including the RID and sequence number of each LSA in the LSDB.
- **Link State Update (LSU)**—Contains a full LSA entry. An LSA includes topology information; for example, the RID of this router and the RID and cost to each neighbor. One LSU can contain multiple LSAs.
- **Link State Acknowledgment (LSAck)**—Acknowledges all other OSPF packets (except hellos)

OSPF traffic is multicast to either of two addresses: 224.0.0.5 for all OSPF routers or 224.0.0.6 for all OSPF DRs.

OSPF Neighbor Relationships

OSPF routers send out periodic multicast packets to introduce themselves to other routers on a link. They become neighbors when they see their own router ID included in the Neighbor field of the hello from another router. Seeing this tells each router that they have bidirectional communication. In addition, two routers must be on a common subnet for a neighbor relationship to be formed. (Virtual links are sometimes an exception to this rule.)

Certain parameters within the OSPF hellos must also match in order for two routers to become neighbors. They include:

- Hello/dead timers
- Area ID
- Authentication type and password
- Stub area flag

OSPF routers can be neighbors without being adjacent. Only adjacent neighbors exchange routing updates and synchronize their databases. On a point-to-point link, an adjacency is established between the two routers when they can communicate. On a multiaccess link, each router establishes an adjacency only with the DR and the backup DR (BDR).

Hellos also serve as keepalives. A neighbor is considered lost if no Hello is received within four Hello periods (called the dead time). The default hello/dead timers are as follows:

- 10 seconds/40 seconds for LAN and point-to-point interfaces
- 30 seconds/120 seconds for nonbroadcast multiaccess (NBMA) interfaces

Establishing Neighbors and Exchanging Routes

The process of neighbor establishment and route exchange between two OSPF routers is as follows:

Step 1. **Down state**—OSPF process not yet started, so no hellos sent.

Step 2. **Init state**—Router sends hello packets out all OSPF interfaces.

Step 3. **Two-way state**—Router receives a hello from another router that contains its own router ID in the neighbor list. All other required elements match, so routers can become neighbors.

Step 4. **Exstart state**—If routers become adjacent (exchange routes), they determine who will start the exchange process.

Step 5. **Exchange state**—Routers exchange DBDs listing the LSAs in their LSD by RID and sequence number.

Step 6. **Loading state**—Each router compares the DBD received to the contents of its LS database. It then sends a LSR for missing or outdated LSAs. Each router responds to its neighbor's LSR with a Link State Update. Each LSU is acknowledged.

Step 7. **Full state**—The LSDB has been synchronized with the adjacent neighbor.

Basic OSPF Configuration

OSPF is configured by entering router configuration mode and identifying the range of interface addresses on which it should run and the areas they are in. When setting up OSPF, a process ID must be used (8 is used in the example), but the process ID does not have to agree on different OSPF devices for them to exchange information. The network statement uses a wildcard mask and can specify any range from a single address to all addresses. Unlike EIGRP, the wildcard mask is not optional. The following example shows a router configured as an ABR. Interfaces falling with the 192.168.1.0 network are placed in area 0, and interfaces falling within the 172.16.1.0 network are placed in area 1.

```
Router(config)#router ospf 8
Router(config-router)#network 192.168.1.0 0.0.0.255 area 0
Router(config-router)#network 172.16.1.0 0.0.0.255 area 1
```

CHAPTER 3

OSPF

Router ID

The SPF algorithm is used to map the shortest path between a series of nodes. This causes an issue with IP, because an IP router is not identified by a single IP address—its interfaces are. For this reason, a single IP address is designated as the "name" of the router—the RID.

By default, the RID is the highest loopback IP address. If no loopback addresses are configured, the RID is the highest IP address on an active interface when the OSPF process is started. The RID is selected when OSPF starts and—for reasons of stability—is not changed until OSPF restarts. The OSPF process can be restarted by rebooting or by using the command **clear ip ospf process**. Either choice affects routing in your network for a period of time and should be used only with caution.

A loopback interface is a virtual interface, so it is more stable than a physical interface for RID use. A loopback address is configured by creating an interface and assigning an IP address.

```
Router(config)#interface loopback0
Router(config-if)#ip address 10.0.0.1 255.255.255.255
```

The loopback address does not have to be included in the OSPF routing process, but if you advertise it, you are able to ping or trace to it. This can help in troubleshooting.

A way to override the default RID selection is to statically assign it using the OSPF **router-id** command

```
Router(config)#router ospf 8
Router(config-router)#router-id 10.0.0.1
```

Troubleshooting OSPF

The neighbor initialization process can be viewed using the **debug ip ospf adjacencies** command. The neighbor table can be seen with **show ip ospf neighbors**, which also identifies adjacency status, and reveals the designated router and backup designated router. Use the **debug ip ospf packet** command to view all OSPF packets in real time.

Often, the first place OSPF issues are noticed is when inspecting the routing table—**show ip route**. To filter the routing table and show only the routes learned from OSPF, use **show ip route ospf**.

The command **show ip protocols** offers a wealth of information for any routing protocol issue. Use this command to verify parameters, timer values, identified networks, and OSPF neighbors (routing information sources).

Use **show ip ospf** to verify the RID, timers, and counters. Because wildcard masks sometimes incorrectly group interfaces to areas, another good place to check is **show ip ospf interface**. This shows the interfaces on which OSPF runs and their current correct assigned area.

OSPF Network Types

The SPF algorithm builds a directed graph—paths made up of a series of points connected by direct links. One of the consequences of this directed graph approach is that the algorithm has no way to handle a multiaccess network, such as an Ethernet VLAN. The solution used by OSPF is to elect one router, called the Designated Router (DR), to

represent the entire segment. Point-to-point links fit the SPF model perfectly and don't need any special modeling method. On a point-to-point link, no DR is elected and all traffic is multicast to 224.0.0.5.

OSPF supports five network types:

- **NBMA**—Default for multipoint serial interfaces. RFC-compliant mode that uses DRs and requires manual neighbor configuration.
- **Point-to-multipoint (P2MP)**—Doesn't use DRs so adjacencies increase logarithmically with routers. Resilient RFC compliant mode that automatically discovers neighbors.
- **Point-to-multipoint nonbroadcast (P2MNB)**—Proprietary mode that is used on Layer 2 facilities where dynamic neighbor discovery is not supported. Requires manual neighbor configuration.
- **Broadcast**—Default mode for LANs. Uses DRs and automatic neighbor discovery. Proprietary when used on WAN interface.
- **Point-to-point (P2P)**—Proprietary mode that discovers neighbors and doesn't require a DR.

If the default interface type is unsatisfactory, you can statically configure it with the command **ip ospf network** under interface configuration mode:

```
Router(config-if)#ip ospf network point-to-multipoint
```

When using the NBMA or P2MP nonbroadcast mode, neighbors must be manually defined under the routing process:

```
Router(config-router)#neighbor 172.16.0.1
```

Designated Routers

On a multiaccess link, one of the routers is elected as a DR and another as a backup DR (BDR). All other routers on that link become adjacent only to the DR and BDR, not to each other (they stop at the two-way state). The DR is responsible for creating and flooding a network LSA (type 2) advertising the multiaccess link. NonDR (DROTHER) routers communicate with DRs using the IP address 224.0.0.6. The DRs use IP address 224.0.0.5 to pass information to other routers.

The DR and BDR are elected as follows:

Step 1. A router starting the OSPF process listens for OSPF hellos. If none are heard within the dead time, it declares itself the DR.

Step 2. If hellos from any other routers are heard, the router with the highest OSPF priority is elected DR, and the election process starts again for BDR. A priority of zero removes a router from the election.

Step 3. If two or more routers have the same OSPF priority, the router with the highest RID is elected DR, and the election process starts again for BDR.

After a DR is elected, elections do not take place again unless the DR or BDR are lost. Because of this, the DR is sometimes the first device that comes online with a nonzero priority.

The best way to control DR election is to set OSPF priority for the DR and BDR for other routers. The default priority is one. A priority of

zero means that a router cannot act as DR or BDR; it can be a DROTHER only. Priority can be set with the **ip ospf priority** command in interface configuration mode.

```
Router(config)#int fa 0/1
Router(config-if)#ip ospf priority 2
```

Nonbroadcast Multiaccess (NBMA) Networks

Routing protocols assume that multiaccess links support broadcast and have full-mesh connectivity from any device to any device. In terms of OSPF, this means the following.

- All Frame Relay or ATM maps should include the broadcast attribute.
- The DR and BDR should have full virtual circuit connectivity to all other devices
- Hub-and-spoke environments should either configure the DR as the hub or use point-to-point subinterfaces, which require no DR
- Partial-mesh environments should be configured using point-to-point subinterfaces, especially when no single device has full connectivity to all other devices. If there is a subset of the topology with full connectivity, then that subset can use a multipoint subinterface.
- Full mesh environments can be configured using the physical interface, but often logical interfaces are used to take advantage of the other benefits of subinterfaces.
- It may be necessary to statically identify neighbor IP addresses.

Advanced OSPF Configuration

OSPF provides many different ways to customize its operation to fit your network needs. This section discusses route summarization, default routes, stub areas, and virtual links

OSPF Summarization

Summarization helps all routing protocols scale to larger networks, but OSPF especially benefits because its processes tax the memory and CPU resources of the routers. The SPF algorithm consumes all CPU resources when it runs. Summarization prevents topology changes from being passed outside an area and thus saves routers in other areas from having to run the SPF algorithm. OSPF's multiple databases use more memory the larger they are. Summarization decreases the number of routes exchanged, and thus the size of the databases. OSPF can produce summaries within a classful network (VLSM) or summaries of blocks of classful networks (CIDR). There are two types of summarizations

- **Inter-area route summarizations** are created on the ABR under the OSPF routing process using the **area range** command. The following command advertises 172.16.0.0/12 from area 1:

```
Router(config-router)#area 1 range 172.16.0.0 255.240.0.0
```

- **External route summarization** is done on an ASBR using the summary-address command under the OSPF routing process. The following example summarizes a range of external routes to 192.168.0.0/16 and injects a single route into OSPF.

```
Router(config-router)#summary-address 192.168.0.0
255.255.0.0
```

Creating a Default Route

The default route is a special type of summarization; it summarizes all networks down to one route announcement. This provides the ultimate benefit of summarization by reducing routing information to a minimum. There are several ways to use the router IOS to place a default route into OSPF.

The best-known way to produce an OSPF default is to use the **default-information** command under the OSPF routing process. This command, without the keyword **always**, readvertises a default route learned from another source into OSPF. If the **always** keyword is present, OSPF advertises a default even if one does not already exist in the routing table. The **metric** keyword sets the starting metric for this route.

```
Router(config-router)#default-information originate [always]
 [metric metric]
```

Alternatively, a default summary route can also be produced using the **summary-address** command or the **area range** command. These commands cause the router to advertise a default route pointing to itself.

Reducing routing information in non-backbone areas is a common requirement because these routers are typically the most vulnerable in terms of processor and speed, and the links that connect them usually have the least bandwidth. A specific concern is that an area will be overwhelmed by external routing information.

Stub and Not-So-Stubby Areas

Another way to reduce the route information advertised is to make an area a stub area. Configuring an area as a stub area forces its ABR to drop all external (type 5) routes and replaces them with a default route. To limit routing information even more, an area can be made totally stubby using the **no-summary** keyword on the ABR only. In that case, all interarea and external routes are dropped by the ABR and replaced by a default route. The default route starts with a cost of 1; to change it, use the **area default-cost** command. The example that follows shows area 2 configured as a totally stubby area, and the default route injected with a cost of 5:

```
Router(config-router)#area 2 stub no-summary
Router(config router)#area 2 default-cost 5
```

Stub areas are attractive because of their low overhead. They do have some limitations, including the following:

- Stub areas can't include a virtual link.
- Stub areas can't include an ASBR.
- Stubbiness must be configured on all routers in the area.

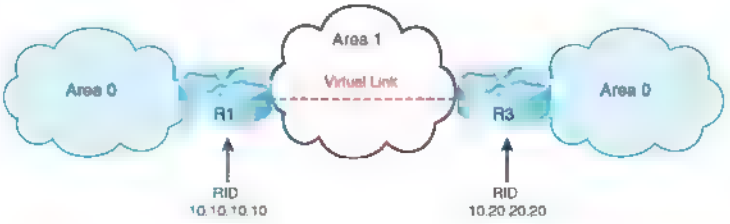
Another kind of stub area is a not-so-stubby area (NSSA). NSSA is like a stub or totally stub area, but allows an ASBR within the area. External routes are advertised as type 7 routes by the ASBR. The ABR converts them to type 5 external routes when it advertises them into adjacent areas. NSSA is configured with the **area nssa** command under the OSPF routing process. The **no-summary** keyword on the ABR configures the area as a totally NSSA area; this is a Cisco proprietary feature. By default, the ABR does not inject a default route back into an NSSA area. Use the **default-information-originate** keyword on the ABR or ASBR to create this route.

```
Router(config-router)#area 7 nssa [no-summary] [default-
 information-originate]
```

Configuring Virtual Links

OSPF requires that all areas be connected to area 0 and that area 0 must be contiguous. When this is not possible, you can use a virtual link to bridge across an intermediate area. Figure 3-3 shows a virtual link connecting two portions of area 0.

FIGURE 3-3 OSPF Virtual Link



Area 1 is the transit area for the virtual link. Configure each end of a virtual link on the ABRs of the transit area with the command **area** *area-number* **virtual-link** *router-id* Each end of the link is identified by its RID. The area listed in the command is the transit area, not the area being joined by the link. The configuration for R1 is:

```
R1(config)#router ospf 1
R1(config-router)#area 1 virtual-link 10.20.20.20
```

The configuration for R2 is:

```
R2(config)#router ospf 1
R2(config-router)#area 1 virtual-link 10.10.10.10
```

Verify that the virtual link is up with the **show ip ospf virtual-links** command. Additionally, virtual interfaces are treated as actual interfaces by the OSPF process, and thus, their status can be verified with the **show ip ospf interface** *interface-id* command.

CHAPTER 3

OSPF

Configuring OSPF Authentication

For security purposes, you can configure OSPF to authenticate every OSPF packet and the source of every OSPF routing update. By default, the router does no authentication. OSPF supports three types of authentication:

- Null authentication for a link that does not use authentication at all
- Simple (plain text) authentication
- MD5 authentication

The following example shows a router configured for simple password authentication in OSPF area 1, using a password of "simple". Note that authentication commands are necessary both under the OSPF process and the interface configuration. All OSPF neighbors reachable through an interface configured for authentication must use the same password. You can, however, use different passwords for different interfaces.

```
Router(config)#int gi0/0
Router(config-if)#ip ospf authentication-key simple
Router(config-if)#ip ospf authentication
Router(config-if)#!
Router(config-if)#router ospf 1
Router(config-router)#area 1 authentication
```

The next example shows the same router configured for OSPF MD5 authentication for area 0, using a password of "secure". Note that the commands are slightly different. The optional keyword **message-digest** is required in two of the commands, and a key number must be specified. Any neighbors reachable through the Gi0/1 interface must also be configured with the same key.

```
Router(config router)#int gi0/1
Router(config-if)#ip ospf message-digest-key 2 md5 secure
Router(config-if)#ip ospf authentication message-digest
Router(config-if)#!
Router(config-if)#router ospf 1
Router(config-router)#area 0 authentication message-digest
```

CHAPTER 4

IS-IS

Intermediate System-to-Intermediate System (IS-IS) is a link state routing protocol that is part of the OSI family of protocols. Like OSPF, it uses Dijkstra's SPF algorithm to choose routes. IS-IS is a classless interior gateway protocol that uses router resources efficiently and scales to large networks, such as large Internet service providers (ISP).

Table 4-1 lists some IS-IS terms, acronyms, and their meanings.

TABLE 4-1 IS-IS Acronyms

Term	Acronym	Description
Circuit ID		Identifies a physical interface on the router
Complete Sequence Number PDU	CSNP	A summary of a router's complete LSDB.
Connectionless Network Protocol	CLNP	OSI protocol used to provide the connectionless services.
Connectionless Network Services	CNLS	OSI data delivery service that provides best-effort delivery
End System	ES	A host, such as a computer.
Intermediate System	IS	The OSI name for a router.
Intermediate System hello	ISH	Sent by routers to hosts.
IS to IS hello	IIH	Hellos exchanged between routers. Seperate level 1 and level 2 IIHs exist.
Link State Database	LSDB	A database containing all the LSAs the router knows about, and it keeps a separate LSDB for each area it belongs to.
Link State PDU	LSP	A routing update.
Network Entity Title	NET	A router's NSAP. The last byte of a NET is always zero.

continues

CHAPTER 4

IS-IS

TABLE 4-1 IS-IS Acronyms Continued

Term	Acronym	Description
Network Service Access Point	NSAP	Address of a CLNS device. Addresses are assigned per device, not per interface as with IP.
NSAP Selector	NSEL	The last byte of a NSAP address. Identifies the process on the device, such as routing.
Protocol Data Unit	PDU	A unit of data.
Partial Route Calculation	PRC	Used to determine end system and IP subnet reachability.
Partial Sequence Number PDU	PSNP	Used to acknowledge receipt of a CSNP and to request more information about a network contained in a CSNP.
Sequence Number Protocol Data Unit	SNP	An IS-IS packet that is sequenced and must be acknowledged. The sequence number helps a router maintain the most recent link state information.
Subnetwork Point of Attachment	SNPA	Layer 2 identification for a router's interface, such as MAC address or DLCI.
Type Length Value	TLV	Fields in the IS-IS updates that contain IP subnet, authentication, and end-system information.

IS-IS Overview

Integrated IS-IS can carry IP network information, but does not use IP as its transport protocol. It uses OSI protocols CLNS and CLNP to deliver its updates. IS-IS sends its messages in PDUs. There are four IS-IS PDU types: Hello, LSP, PSNP, and CSNP.

Types of IS-IS Routers

Figure 4-1 shows an IS-IS network divided into areas. The IS-IS backbone is not a specific area, as in OSPF, but an unbroken chain of routers doing Level 2 routing. R3, R6, and R4 are the backbone in Figure 4-1.

CHAPTER 4

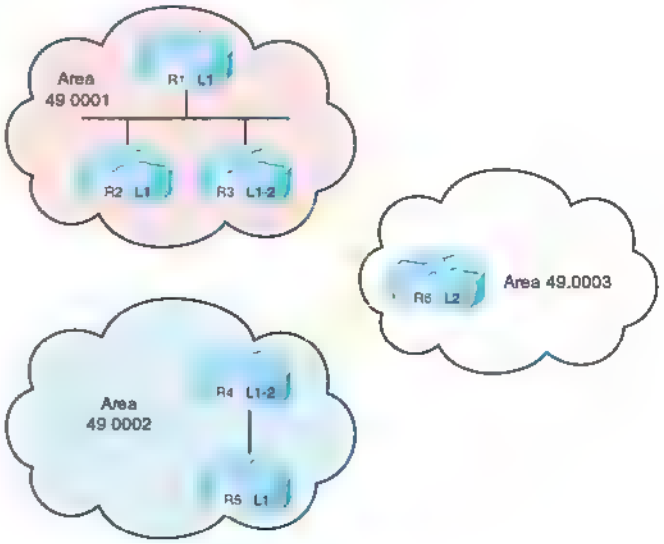
IS-IS

Within an area, routers can be one of three types:

- **Level 1 (L1) router**—R1, R2, and R5 in the figure. Routes to networks only within the local area (intra-area routing). Uses a default route to the nearest Level 2 router for traffic bound outside the area. Keeps one LSDB for the local area. When routing, compares the area of the destination to its area. If they are the same, routes based on system ID. If not, sends traffic to Level 1-2 router.
- **Level 2 (L2) router**—R6 in the figure. Routes to networks in other areas (interarea routing). The routing is based on area ID. Keeps one LSDB for routing to other areas.
- **Level 1-2 (L1-2) router**—R3 and R4 in this figure. Acts as a gateway into and out of an area. Does Level 1 routing within the area and Level 2 routing between areas. Keeps two LSDB: one for the local area and one for interarea routing.

The IS-IS method of selecting routes can result in suboptimal routing between areas. To solve this, RFC 2966 introduces route leaking, which allows some L2 routes to be advertised (or leaked) into L1 areas.

FIGURE 4-1 IS-IS Network Structure



NSAP Address Structure

In the Cisco implementation of integrated IS-IS, NSAP addresses have three parts: the area ID, the system ID, and the NSEL. They are written in hexadecimal and have a maximum size of 20 bytes.

- Area IDs vary from 1 to 13 bytes. Those that begin with 49 designate private area addressing.
- The Cisco system ID must be exactly six bytes. MAC addresses or IP addresses padded with 0s are often used as system IDs.
- The NSEL is exactly one byte in size. A router always has a NSEL of 00.

Figure 4-2 shows the composition of an NSAP address.

FIGURE 4-2 IS-IS NSAP Address

Adjacency Formation in IS-IS

IS-IS routers form adjacencies based on the level of IS routing they are doing and their area number. This is a CLNS adjacency and can be formed even if IP addresses don't match.

- Level 1 routers form adjacencies only with L1 and L1-2 devices in their own area. (In Figure 4-1, R1 becomes adjacent with R2 and R3.)
- Level 2 routers form adjacencies only with Level 2-capable devices (either L2 or L1-2 routers). These can be in the local area or in other areas. (In Figure 4-1, R6 becomes adjacent with R3 and R4.)
- Level 1-2 routers form Level 1 adjacencies with L1 routers in their own area, and Level 2 adjacencies with routers in other areas. (In Figure 4-1, R4 has a L1 adjacency with R5 and a L2 adjacency with R6.)

IS-IS Network Types

IS-IS recognizes only broadcast and point-to-point links. In Frame Relay, multipoint interfaces must be fully meshed. Use point-to-point subinterfaces to avoid this.

On a broadcast network, IS-IS routers elect a Designated Intermediate System (DIS). The DIS is elected based on priority, with MAC address as the tie breaker (the lowest number wins for both priority and MAC address). Routers form adjacencies with the DIS and all other routers on the LAN. The DIS creates a pseudonode to represent the network and sends out an advertisement to represent the LAN. All routers advertise only an adjacency to the pseudonode. If the DIS fails, another is elected; no backup DIS exists. The DIS sends Hellos every 3.3 seconds; other routers send them every 10 seconds. The DIS also multicasts a CSNP every 10 seconds.

No DIS exists on a point-to-point link. When an adjacency is first formed over the link, the routers exchange CSNPs. If one of the routers needs more information about a specific network, it sends a PSNP requesting that. After the initial exchange, LSPs are sent to describe link changes, and they are acknowledged with PSNPs. Hellos are sent every 10 seconds.

CHAPTER 4

IS-IS

Configuring IS-IS

The essential tasks to begin IS-IS routing are:

- Enable IS-IS on the router:

```
Router(config)#router isis
```

- Configure each router's NET:

```
Router(config-router)#net 49.0010.1111.2222.3333.00
```

- Enable IS IS on the router's interfaces.

```
Router(config)#interface s0/0/0
```

```
Router(config int)#ip router isis
```

You may wish to do some tuning of IS-IS routing. Following are the tasks:

- **Set the IS level.** Cisco routers are L1-2 by default. If the router is completely an internal area router, set the IS level to L1. If the router routes only to other areas and has no internal area interfaces, set the IS level to L2. If the router has both internal and external area interfaces, leave the IS level at L1-2.

```
Router(config-router)is-type {level-1 | level 1-2 | level-2-only}
```

- **Set the circuit type on L1-2 routers.** On L1-2 routers, all interfaces send out both L1 and L2 hellos, trying to establish both types of adjacencies. This can waste bandwidth. If only an L1 router is attached to an interface, then change the circuit type for that interface to L1, so that only L1 hellos are sent. If there is only a L2 router attached to an interface, change the circuit type for that interface to L2:

```
Router(config-int)#isis circuit-type {level-1 | level 1-2 |
level-2-only}
```

- **Summarize addresses.** Although IS IS does CLNS routing, it can summarize the IP addresses that it carries. Summarized routes can be designated as Level 1, Level 2, or Level 1-2 routes. The default is Level 2:

```
Router(config-router)#summary-address prefix mask [level-1 |
level-2 | level-1-2]
```

- **Adjust the metric.** IS-IS uses a metric of 10 for each interface. You can manually assign a metric that more accurately reflects the interface characteristics, such as bandwidth:

```
Router(config-int)#isis metric metric {level-1 | level-2}
```

Verifying and Troubleshooting IS-IS

Table 4-2 shows some IS-IS verification and troubleshooting commands, and describes the information you obtain from these commands.

CHAPTER 4

IS-IS

TABLE 4-2 IS-IS **show** Commands

Command	Description
show isis topology	Displays the topology database and least cost paths.
show clns route	Displays the L2 routing table.
show isis route	Displays the L1 routing table. Requires that CLNS routing is enabled
show clns protocol	Displays the router's IS type, system ID, area ID, interfaces running IS-IS, and any redistribution.
show clns neighbors	Displays the adjacent neighbors and their IS level.
show clns interface	Displays IS-IS details for each interface, such as circuit type, metric, and priority
show ip protocols	Displays the integrated IS-IS settings.

CHAPTER 5

Optimizing Routing

There are times when you need to go beyond just turning on a routing protocol in your network. You may need to use multiple protocols, control exactly which routes are advertised or redistributed, or which paths are chosen. Most networks use DHCP; your router may need to be a DHCP server, or relay DHCP broadcasts.

Using Multiple Routing Protocols

There are several reasons you may need to run multiple routing protocols in your network. Some include:

- Migrating from one routing protocol to another, where both protocols will run in the network temporarily
- Applications that run under certain routing protocols but not others
- Areas of the network under different administrative control ("layer 8" issues)
- A multi-vendor environment in which some parts of the network require a standards-based protocol

Configuring Route Redistribution

If routing information must be exchanged among the different protocols or routing domains, redistribution can be used. Only routes that are in the routing table and learned via the specified protocol are redistributed. Each protocol has some unique characteristics when redistributing, as shown in Table 5-1.

TABLE 5-1 Route Redistribution Characteristics

Protocol	Redistribution Characteristics
RIP	Metric must be set, except when redistributing static or connected routes, which have a metric of 1
OSPF	Default metric is 20. Can specify the metric type; the default is E2. Must use subnets keyword or only classful networks are redistributed.
EIGRP	Metric must be set, except when redistributing static or connected routes, which get their metric from the interface. Metric value is "bandwidth, delay, reliability, load, MTU." Redistributed routes have a higher administrative distance than internal ones.
IS-IS	Default metric is 0. Can specify route level; default is L2. Can choose to redistribute only external or internal routes into IS- IS from OSPF and into OSPF from IS-IS
Static/Connected	To include local networks not running the routing protocol, you must redistribute connected interfaces. You can also redistribute static routes into a dynamic protocol.

CHAPTER 6

OPTIMIZING ROUTING

You can redistribute only between protocols that use the same protocol stack, such as IP protocols, which cannot advertise IPX routes. To configure redistribution, issue this command under the routing process that is to receive the new routes:

```
Router(config-router)#redistribute {route-source} [metric metric]
 [route-map tag]
```

Seed Metric

Redistribution involves configuring a routing protocol to advertise routes learned by another routing process. Normally, protocols base their metric on an interface value, such as bandwidth, but no interface for a redistributed route exists. Protocols use incompatible metrics, so the redistributed routes must be assigned a new metric compatible with the new protocol.

A route's starting metric is called its *seed metric*. Set the seed metric for all redistributed routes with the **default-metric** [*metric*] command under the routing process. To set the metric for specific routes, either use the **metric** keyword when redistributing or use the **route-map** keyword to link a route map to the redistribution. After the seed metric is specified, it increments normally as the route is advertised through the network (except for certain OSPF routes).

Tools for Controlling/ Preventing Routing Updates

Cisco IOS provides several ways to control routing updates. They include:

- Passive interface
- Default and/or static routes
- Distribute list
- Route map
- Change administrative distance

Passive Interface

The **passive-interface** command prevents routing updates from being sent out an interface that runs the routing protocol. RIP and IGRP do not send updates out an interface. It prevents other routing protocols from sending hellos out of an interface; thus, they don't discover neighbors or form an adjacency out that interface. To disable the protocol on one interface, use the command **passive-interface** *interface*. To turn off the protocol on all interfaces, use **passive-interface default**. You can then use **no passive-interface** *interface* for the ones that should run the protocol, as shown:

```
Router(config)#router eigrp 7
Router(config-router)#passive-interface default
Router(config-router)#no passive-interface s0/0/0
```

CHAPTER 6

OPTIMIZING ROUTING

Distribute Lists

A distribute list allows you to filter routing updates and also routes being redistributed through an access list. Configure an access list that permits the routes to be advertised or redistributed, and then link that access list to the routing process with the **distribute-list** command, given under router configuration mode. This command has two options:

- **distribute-list** *access-list* **in**—Filters updates as they come in an interface. For OSPF, this controls routes placed in the routing table but not the database. For other protocols, this controls the routes the protocol knows about.
- **distribute-list** *access-list* **out** –Filters updates going out of an interface and also updates being redistributed out of another routing protocol into this one.

Route Maps

Route maps are a bit like programs that use a "if/then/else" decision-making capability. They *match* traffic against certain conditions, and then set specified options for that traffic. Each statement has a sequence number, statements are read from the lowest number to highest, and the router stops reading when it gets a match. The sequence number can be used to insert or delete statements. Like an access list, there is an implicit "deny" at the end of each route map; any traffic not matched with a route map statement is denied. Some uses for route maps include:

- Filtering redistributed routes—Use the **route-map** keyword in the redistribute command.
- Policy-based routing—To specify which traffic should be policy routed, based on very granular controls.
- BGP policy—To control routing updates and to manipulate path attributes.

Route Map Syntax

Route maps are created with the global command:

```
Router(config)#route-map {tag} permit | deny [sequence_number]
```

Each statement in a route map begins this same way, with the same route map name but different sequence numbers, and with match and/or set conditions below it. *Permit* means that any traffic matching the match conditions is used. *Deny* means that any traffic matching the match conditions is not used.

Match and Set Conditions

Each route map statement can have from none to multiple **match** and **set** conditions. If no **match** condition exists, the statement matches anything, similar to a "permit any" in an access list. If there is no **set** condition, the matching traffic is either permitted or denied, with no other conditions being set.

CHAPTER 6

OPTIMIZING ROUTING

Multiple match conditions on the same line use a logical OR. For example, the router interprets **match a b c** as "**match a** or **b** or **c**." Multiple match conditions on different lines use a logical AND. For example, the router interprets the following route map statement as "**match a** and **b** and **c**:"

```
route-map Logical-AND permit 10
 match a
 match b
 match c
```

In route redistribution, some common conditions to **match** include:

- **ip address**—Refers the router to an access list that permits or denies networks.
- **ip next-hop**—Refers the router to an access list that permits or denies next-hop IP addresses.
- **ip route-source**—Refers the router to an access list that permits or denies advertising router IP addresses.
- **metric**—Permits or denies routes with the specified metric from being redistributed.
- **route-type**—Permits or denies redistribution of the route type listed, such as internal or external
- **tag**—Routes can be labeled (tagged) with a number, and route maps can look for that number.

In route redistribution, some common conditions to **set** are:

- **metric**—Sets the metric for redistributed routes.
- **metric-type**—Sets the type, such as E1 for OSPF
- **tag**—Tags a route with a number that can be matched on later by other route maps.
- **level**—For IS-IS, sets the IS level for this route.

The following configuration example shows a route map named BGP-LP with three statements that are used to control which routes will be redistributed from OSPF into BGP. The router has already been configured with two access lists, numbered 23 and 103 (not shown.) The first route map statement, with sequence number 10, is a *permit* statement. The **match** condition tells it to use access list 23. Any traffic permitted by access list 23 matches this statement and will be redistributed into BGP. Any traffic explicitly denied by access list 23 will not be redistributed into BGP. The **set** condition tells it to set a BGP local preference for all traffic that matches statement 10. Traffic not matching access list 23 will be checked against the second route map statement.

The second route map statement, sequence number 20, is a *deny* statement that matches access list 103. Any traffic permitted by access list 103 will be denied by this statement, and thus will not be redistributed. Any traffic explicitly denied by access list 103 will be ignored by this statement, and checked against the next route map statement. This route map statement has no **set** conditions. Traffic not matching route map statements 10 or 20 will be checked against statement 30.

The third route map statement, sequence number 30, is a *permit* statement with no **match** or **set** conditions. This statement matches everything and sets nothing, thus permitting all other traffic without changing it. Without this statement, all other traffic would be denied.

Lastly, the route map is applied to the redistribution command, to filter routes redistributed from OSPF into BGP.

```
Router(config)#route-map BGP-LP permit 10
Router(config-route-map)#match ip address 23
Router(config-route-map)#set local-preference 200
Router(config-route-map)#!
Router(config-route map)#route-map BGP-LP deny 20
Router(config-route-map)#match ip address 103
Router(config route map)#!
Router(config-route-map)#route-map BGP-LP permit 30
!
Router(config)#router bgp 65001
Router(config-router)#redistribute ospf 1 route-map BGP-LP
```

Manipulating Administrative Distance

When a router receives routes to the same destination network from more than one routing process, it decides which to put in the routing table by looking at the administrative distance (AD) value assigned to the routing process. The route with the lowest AD is chosen. Table 5-2 shows administrative distance values

TABLE 5-2 Administrative Distance

Routing Information Source	Administrative Distance
Connected interface	0
Static route	1
EIGRP summarized route	5
BGP external route	20
EIGRP internal route	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
EIGRP external route	170
BGP internal route	200
Unknown	255

AD can be changed for all routes of a process or only for specific routes within a process. The command for all IGPs except EIGRP is:

```
Router(config-router)#distance administrative_distance {address
 wildcard-mask} [access-list-number | name]
```

Using the **address/mask** keywords in the command changes the AD of routes learned from the neighbor with that IP address. An entry of **0.0.0.0 255.255.255.255** changes the AD of all routes. Specifying an access list number or name changes the AD only on networks permitted in the ACL.

CHAPTER 5

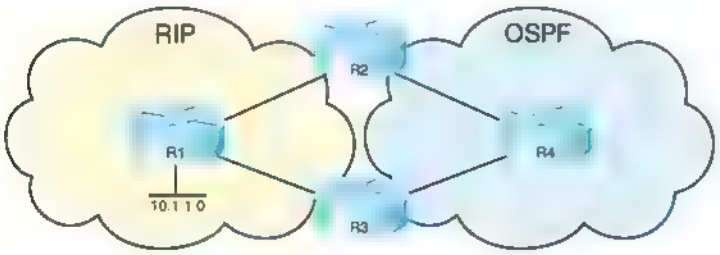
OPTIMIZING ROUTING

EIGRP and BGP have different AD values for internal and external routes, so you have to list those separately when using the command with those protocols. BGP also allows you to change the AD for locally generated routes. For these protocols, the commands are:

```
Router(config-router)#distance eigrp internal-distance external-
 distance
Router(config-router)#distance bgp external-distance internal-
 distance local distance
```

Route redistribution can cause suboptimal routing; one way to correct this is to adjust AD. Figure 5-1 shows a network with two routing domains: RIP and OSPF.

FIGURE 5-1 Controlling Routing with AD



R2 redistributes its RIP routes into OSPF. These routes inherit OSPF's AD when they are advertised to R4. R4 then advertises them to R3 as OSPF routes.

R3 now knows about the 10.1.1.0 network from two routing processes: RIP, with an AD of 120, and OSPF, with an AD of 110. The shortest path is the RIP route through R1. The OSPF path goes through R4 and R2, and then to R1—a much longer path. But, based on AD, R3 puts the OSPF path in its routing table.

To prevent this, increase the AD of the redistributed RIP routes when OSPF advertises them. Note that this doesn't change all OSPF routes, just the ones learned from RIP. The commands given on R2 (the router doing the initial redistribution) are shown in the following:

```
Router(config)#access-list 10 permit 10.1.1.0
!
Router(config)#router ospf 1
Router(config-router)#redistribute rip subnets
Router(config-router)#distance 125 0.0.0.0 255.255.255.255 10
```

The AD is increased to 125 for routes from all neighbors, if they match the network permitted in access list 10. Now R3 hears about the 10.1.1.0 network from RIP with an AD of 120, and from OSPF with an AD of 125. The RIP route is put into the routing table based on its lower AD.

DHCP

DHCP automates the assignment of IP addresses to network hosts. DHCP addresses can be allocated:

- Manually—A specific IP address is assigned to a MAC address.
- Automatic—An IP address is permanently assigned to a host.
- Dynamic—The IP address is assigned for a limited amount of time or until the client releases it.

CHAPTER 6

OPTIMIZING ROUTING

The process of acquiring an IP address from a DHCP server has four steps.

Step 1. The host broadcasts a DHCPDISCOVER message.

Step 2. The server responds with a DHCPOFFER message containing IP address and optionally other settings.

Step 3. The client broadcasts a DHCPREQUEST message, requesting the offered IP address.

Step 4. The server sends a DHCPACK confirming the address assignment.

Configuring DHCP

Cisco routers can be DHCP clients, servers, or relay agents. To configure an IOS device as a DHCP client, use the **ip address dhcp** command on the interface that needs to obtain the DHCP address. To configure a router as a DHCP server, you must create an IP address pool and assign a network or subnet to that pool. You can optionally add information, such as default gateway, DNS server, lease duration, or options such as Option 150 for Cisco IP phones. Exclude any static IP addresses within the pool, such as the router's address. You may also want to identify an external server to hold the DHCP database of IP address bindings.

Cisco routers have an auto-configuration feature that allows the downloading of some DHCP information from a central server. This saves the trouble of configuring every router with complete DHCP information. To do this, one interface on the router must have a DHCP address.

The following example shows a router configured as a DHCP server that imports its domain name, DNS servers, and other information from another DHCP server off interface Gi0/0. The IP address range of 10.6.3.1–10.6.3.5 is excluded from the pool.

```
Router(config)#ip dhcp excluded-address 10.6.3.1 10.6.3.5
!
Router(config)#ip dhcp pool Gator
Router(dhcp-config)#network 10.6.3.0 /24
Router(dhcp-config)#default-router 10.6.3.1
Router(dhcp-config)#import all
!
Router(config)#int gi 0/0
Router(config-if)#ip address dhcp
```

DHCP Relay Agent

Hosts discover their DHCP server by sending broadcasts. If that server is on a different subnet, those broadcasts must be routed to the server as unicasts. You can configure a router to relay DHCP messages with the **ip helper-address** interface command. It is important to understand that this command must be given on the interface that receives the host broadcasts. A Cisco DHCP relay agent functions as follows:

Step 1. A client broadcasts a DHCP request, which is seen by the IOS device (a router, for instance).

Step 2. The router changes the destination address of the packet to the unicast address of the server. It optionally adds option 82 (relay agent option) information.

Step 3. The router sends the unicast packet to the server.

Step 4. The server responds with the IP address and other parameters, such as the default gateway assigned to the client.

Step 5. The router gets the packet from the server, removes any option 82 information, and forwards it to the client.

The **ip helper-address** command enables the relaying of UDP broadcasts only. By default, eight broadcast types are enabled:

- Time, port 37
- TACACS, port 49
- DNS, port 53
- BOOTP/DHCP server, port 67
- BOOTP/DHCP client, port 68
- TFTP, port 69
- NetBIOS name service, port 137
- NetBIOS datagram service, port 138

To disable the forwarding any of these protocols, use the interface command **no ip forward-protocol udp** *port-number*. To add UDP protocols to be relayed, use the interface command **ip forward-protocol udp** *port-number*.

Verify your DHCP configuration with the commands **show ip dhcp database**, **show ip dhcp server statistics**, and **show ip dhcp binding**. Delete address assignments with the **clear ip dhcp binding** {*address* | *}} command.

CHAPTER 6

BGP

BGP is an external gateway protocol, meant to be used between different networks. It is the protocol used on the internet. It was built for reliability, scalability and control, not speed. Because of this, it behaves differently from the protocols covered thus far in this book.

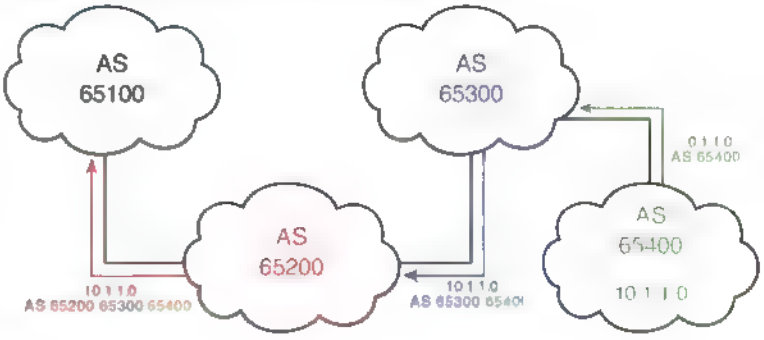
BGP Overview

- BGP stands for Border Gateway Protocol.
- BGP uses the concept of autonomous systems. An *autonomous system* is a group of networks under a common administration.
- Autonomous systems run Interior Gateway Protocols (IGP) within the system. They run an Exterior Gateway Protocol (EGP) between them.
- BGP version 4 is the only EGP currently in use.
- BGP neighbors are called peers and must be statically configured.
- BGP uses TCP port 179.
- BGP is a path-vector protocol. Its route to a network consists of a list of autonomous systems on the path to that network.
- BGP's loop prevention mechanism is autonomous system number.

When an update about a network leaves an autonomous system, that autonomous system's number is prepended to the list of autonomous systems that have handled that update. When an autonomous system receives an update, it examines the autonomous system list. If it finds its own autonomous system number in that list, the update is discarded.

In Figure 6-1, BGP routers in AS 65100 see network 10 1.1 0 as having an autonomous system path of 65200 65300 65400.

FIGURE 6-1 BGP AS-Path Advertisement



CHAPTER 6

BGP

Multihoming

Multihoming means connecting to more than one ISP at the same time. It is done for redundancy and backup in case one ISP fails and for better performance if one ISP provides a better path to often used networks. Three ways exist to receive routes from each ISP:

- Default routes from each provider—This results in low use of bandwidth and router resources. The internal network's IGP metric determines the exit router for all traffic bound outside the autonomous system.
- Default routes plus some more specific routes—This results in medium use of bandwidth and router resources. This allows you to manipulate the exit path for specific routes using BGP, but the IGP metric chooses the exit path for default routes.
- All routes from all providers—This requires the highest use of bandwidth and router resources. It is typically done by large enterprises and ISPs. Path selection for all external routes can be controlled via BGP policy routing tools.

BGP Databases

BGP uses three databases. The first two listed are BGP-specific; the third is shared by all routing processes on the router:

- **Neighbor database**—This is a list of all configured BGP neighbors. To view it, use the **show ip bgp summary** command.
- **BGP database, or RIB (Routing Information Base)**—This is a list of networks known by BGP, along with their paths and attributes. To view it, use the **show ip bgp** command.
- **Routing table**—This is a list of the paths to each network used by the router, and the next hop for each network. To view it, use the **show ip route** command.

BGP Message Types

BGP has four types of messages:

- **Open** After a neighbor is configured, BGP sends an open message to try to establish peering with that neighbor. Includes information such as autonomous system number, router ID, and hold time.
- **Update**—Message used to transfer routing information between peers.
- **Keepalive**—BGP peers exchange keepalive messages every 60 seconds by default. These keep the peering session active.
- **Notification**—When a problem occurs that causes a router to end the BGP peering session, a notification message is sent to the BGP neighbor and the connection is closed.

CHAPTER 6

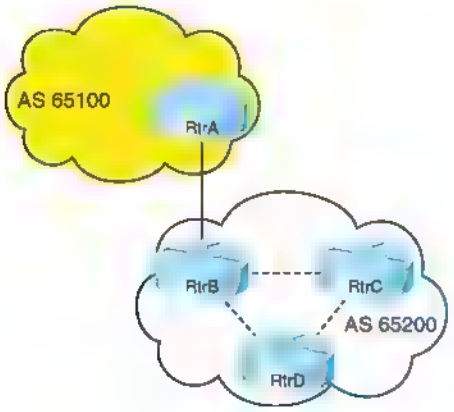
BGP

Internal and External BGP

Internal BGP (IBGP) is BGP peering relationship between routers in the same autonomous system. External BGP (EBGP) is BGP peering relationship between routers in different autonomous systems. BGP treats updates from internal peers differently than updates from external peers

In Figure 6-2, routers A and B are EBGP peers. Routers B, C, and D are IBGP peers.

FIGURE 6-2 Identifying EBGP and IBGP Peers



BGP Next Hop Selection

The next hop for a route received from an EBGP neighbor is the IP address of the neighbor that sent the update.

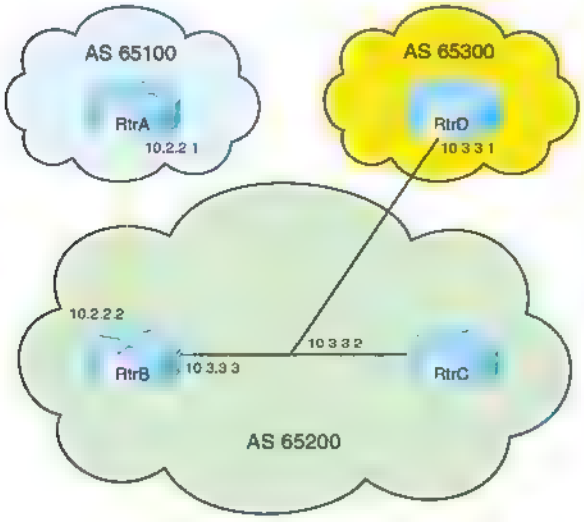
When a BGP router receives an update from an EBGP neighbor, it must pass that update to its IBGP neighbors without changing the next-hop attribute. The next-hop IP address is the IP address of an edge router belonging to the next-hop autonomous system. Therefore, IBGP routers must have a route to the network connecting their autonomous system to that edge router. For example, in Figure 6-3, RtrA sends an update to RtrB, listing a next hop of 10.2.2.1, its serial interface. When RtrB forwards that update to RtrC, the next-hop IP address will still be 10.2.2.1. RtrC needs to have a route to the 10.2.2.0 network in order to have a valid next hop

To change this behavior, use the **neighbor** {*ip address*} **next-hop-self** command in BGP configuration mode. In Figure 6-3, this configuration goes on RtrB. After you give this command, RtrB will advertise its IP address to RtrC as the next hop for networks from AS 65100, rather than the address of RtrA. Thus, RtrC does not have to know about the external network between RtrA and RtrB (network 10.2.2.0).

CHAPTER 6

BGP

FIGURE 6-3 BGP Next-Hop Behavior



BGP Next Hop on a Multiaccess Network

On a multi-access network, BGP can adjust the next-hop attribute to avoid an extra hop. In Figure 6-3, RtrC and RtrD are EBGP peers, and RtrC is an IBGP peer with RtrB. When C sends an update to D about network 10.2.2.0, it normally gives its interface IP address as the next hop for D to use. But because B, C, and D are all on the same multiaccess network, it is inefficient for D to send traffic to C, and C to then send it on to B. This process unnecessarily adds an extra hop to the path. So, by default, RtrC advertises a next hop of 10.3.3.3 (RtrB's interface) for the 10.2.2.0 network. This behavior can also be adjusted with the **neighbor** [*ip address*] **next-hop-self** command.

BGP Synchronization Rule

The BGP synchronization rule requires that when a BGP router receives information about a network from an IBGP neighbor, it does not use that information until a matching route is learned via an IGP or static route. It also does not advertise that route to an EBGP neighbor unless a matching route is in the routing table. In Figure 6-3, if RtrB advertises a route to RtrC, then RtrC does not submit it to the routing table or advertise it to RtrD unless it also learns the route from some other IGP source.

Recent IOS versions have synchronization disabled by default. It is usually safe to turn off synchronization when all routers in the autonomous system run BGP. To turn it off in earlier IOS versions, use the command **no synchronization** under BGP router configuration mode.

CHAPTER 6

BGP

Configuring BGP

Table 6-1 lists the basic BGP configuration commands and their functions.

TABLE 6-1 Basic BGP Configuration Commands

Command	Description
router bgp *AS number*	Starts the BGP routing process on the router.
neighbor *ip-address* **remote-as** *AS number*	Sets up peering between BGP routers.
neighbor *peer-group name* **peer-group**	Creates a peer group to which you can then assign neighbors
neighbor *ip-address* **peer-group** *peer group name*	Assigns a neighbor to a peer group.
neighbor *ip-address* **next-hop-self**	Configures a router to advertise its connected interface as the next hop for all routes to this neighbor.
neighbor *ip address* **update-source** *interface-type number*	Configures a router to use the IP address of a specific interface as the source for its advertisements to this neighbor.
no synchronization	Turns off BGP synchronization.
network prefix [**mask** subnet mask]	Initiates the advertisement of a network in BGP.

The BGP Network Command

In most IGPs, the network command starts the routing process on an interface. In BGP, the command tells the router to originate an advertisement for that network. The network does not have to be connected to the router; it just has to be in the routing table. In theory, it could even be a network in a different autonomous system (not usually recommended)

When advertising a network, BGP assumes you are using the default classful subnet mask. If you want to advertise a subnet, you must use the optional keyword **mask** and specify the subnet mask to use. Note that this is a subnet mask, not the inverse mask used by OSPF and EIGRP network statements. The routing table must contain an exact match (prefix and subnet mask) to the network listed in the network statement before BGP will advertise the route.

BGP Peering

BGP assumes that external neighbors are directly connected and that they are peering with the IP address of the directly connected interface of their neighbor. If not, you must tell BGP to look more than one hop away for its neighbor, with the **neighbor** *ip-address* **ebgp-multihop** *number-of-hops* command. You might use this command if you are peering with loopback interface IP addresses, for instance. BGP assumes that internal neighbors might not be directly connected, so this command is not needed with IBGP.

CHAPTER 6

BGP

BGP Peering States

The command **show ip bgp neighbors** shows a list of peers, and the status of their peering session. This status can include the following states:

- **Idle**—No peering; router is looking for neighbor. Idle (admin) means that the neighbor relationship has been administratively shut down.
- **Connect**—TCP handshake completed.
- **OpenSent, or Active**—An open message was sent to try to establish the peering.
- **OpenConfirm**—Router has received a reply to the open message.
- **Established**—Routers have a BGP peering session. This is the desired state.

You can troubleshoot session establishment with debug commands. Use **debug ip bgp events** or **debug ip bgp ipv4 unicast** (in IOS versions 12.4 and up) to see where the process fails. Some common failure causes include AS number misconfiguration, neighbor IP address misconfiguration, neighbor with no neighbor statement for your router, and neighbor with no route to the source address of your router's BGP messages.

BGP Path Selection

IGP, such as EIGRP or OSPF, choose routes based on lowest metric. They attempt to find the shortest, fastest way to get traffic to its destination. BGP, however, has a very different way of route selection. It assigns various attributes to each path; these attributes can be administratively manipulated in order to control the path that is selected. It then examines the value of these attributes in an ordered fashion until it is able to narrow all the possible routes down to one path.

BGP Attributes

BGP chooses a route to network based on the attributes of its path. Four categories of attributes exist:

- **Well-known mandatory**—Must be recognized by all BGP routers, present in all BGP updates, and passed on to other BGP routers. For example, AS path, origin, and next hop.
- **Well-known discretionary**—Must be recognized by all BGP routers and passed on to other BGP routers, but need not be present in an update. For example, local preference.
- **Optional transitive**—Might or might not be recognized by a BGP router, but is passed on to other BGP routers. If not recognized, it is marked as partial. For example, aggregator, community
- **Optional nontransitive**—Might or might not be recognized by a BGP router and is not passed on to other routers. For example, Multi Exit Discriminator (MED), originator ID

Table 6-2 lists common BGP attributes, their meanings, and their category.

TABLE 6-2 BGP Attributes

Attribute	Meaning
AS path	An ordered list of all the autonomous systems through which this update has passed. Well-known, mandatory.
Origin	How BGP learned of this network. i = by **network** command, e = from EGP, ? = redistributed from other source Well known, mandatory.
Next hop	The IP address of the next-hop router. Well-known, mandatory
Local preference	A value telling IBGP peers which path to select for traffic leaving the AS. Well-known, discretionary.
Multi-Exit Discriminator (MED)	Suggests to a neighboring autonomous system which of multiple paths to select for traffic bound into your autonomous system. Optional, non-transitive.
Weight	Cisco proprietary, to tell a router which of multiple local paths to select for traffic leaving the AS. Only has local significance.

Influencing BGP Path Selection

BGP was not created to be a fast protocol; it was created to allow as much administrative control over route path selection as possible. Path selection is controlled by manipulating BGP attributes, usually using route maps. You can set a default local preference by using the command **bgp default local-preference** and a default MED for redistributed routes with the **default-metric** command under the BGP routing process. But by using route maps, you can change attributes for certain neighbors only or for certain routes only. Click here to see a previous example that shows a route map setting a local preference of 200 for specific redistributed routes. This is higher than the default local preference of 120, so routers within the AS are more likely to prefer that path than others.

Route maps can also be applied to routes sent to or received from a neighbor. The following example shows a simple route map that sets MED on all routes advertised out to an EBGP neighbor:

```
route-map MED permit 10
set metric 50
!
router bgp 65001
neighbor 10.1.1.1 route-map MED out
```

When attributes are changed, you must tell BGP to apply the changes. Either clear the BGP session (**clear ip bgp** *) or do a soft reset (**clear ip bgp** * **soft in** | **out**). Routers using recent IOS versions will do a route refresh when the session in cleared inbound.

BGP Path Selection Criteria

BGP tries to narrow its path selection down to one best path; it does not load balance by default. To do so, it examines the path attributes of any loop-free, synchronized (if synchronization is enabled) routes with a reachable next-hop in the following order:

1. Choose the route with the highest weight.
2. If weight is not set, choose the route with the highest local preference.
3. Choose routes that you advertise.
4. Choose the path with the shortest autonomous system path.
5. Choose the path with the lowest origin code (i is lowest, e is next, ? is last).
6. Choose the route with the lowest MED, if the same autonomous system advertises the possible routes.
7. Choose an eBGP route over an iBGP route.
8. Choose the route through the nearest IGP neighbor.
9. Choose the oldest route.
10. Choose a path through the neighbor with the lowest router ID
11. Choose a path through the neighbor with the lowest IP address.

To enable BGP to load balance over more than one path, you must enter the command **maximum-paths** *number-of-paths*. BGP can load balance over a maximum of six paths.

BGP Authentication

BGP supports MD5 authentication between neighbors, using a shared password. It is configured under BGP router configuration mode with the command **neighbor** {*ip-address* | *peer-group-name*} **password** *password*. When authentication is configured, BGP authenticates every TCP segment from its peer and checks the source of each routing update. Most ISPs require authentication for their EBGP peers.

Peering will succeed only if both routers are configured for authentication and have the same password. If your router has authentication configured and the neighbor does not, your router will display the error message "%TCP-6-BADAUTH: No MD5 digest from *peer's-IP-address*:11003 to *local-router's-IP-address*:179."

If the neighbor router is configured with a nonmatching password, your router will display the error message "%TCP 6-BADAUTH: Invalid MD5 digest from *peer's-IP-address*:11004 to *local-router's-IP-address*:179."

If a router has a password configured for a neighbor, but the neighbor router does not, a message such as the following will display on the console while the routers attempt to establish a BGP session between them:

```
%TCP-6-BADAUTH: No MD5 digest from [peer's IP address]:11003 to
 [local router's IP address]:179
```

Similarly, if the two routers have different passwords configured, a message such as the following will display on the screen:

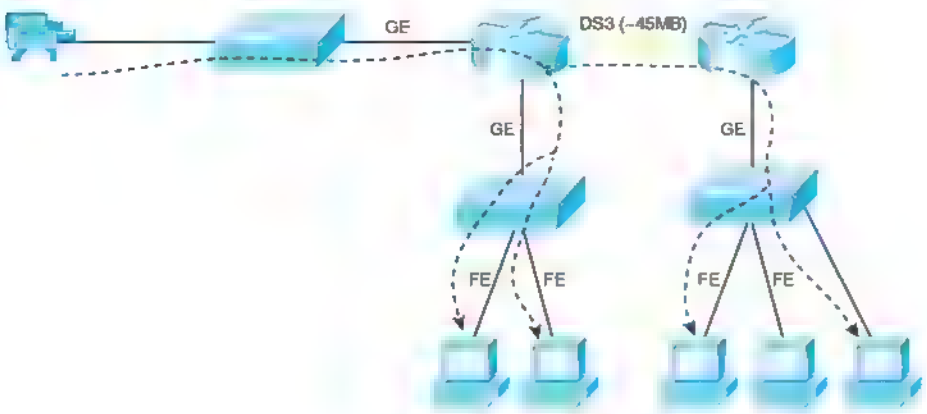
```
%TCP-6-BADAUTH: Invalid MD5 digest from [peer's IP address]:11004
 to [local router's IP address]:179
```

CHAPTER 7

IP Multicast

A multicast is a single data stream sent from one source to a group of recipients. Examples might be a stock ticker or live video feed. Figure 7-3 shows an example multicast topology, as contrasted to unicast and broadcast.

FIGURE 7-1 Multicast Topology

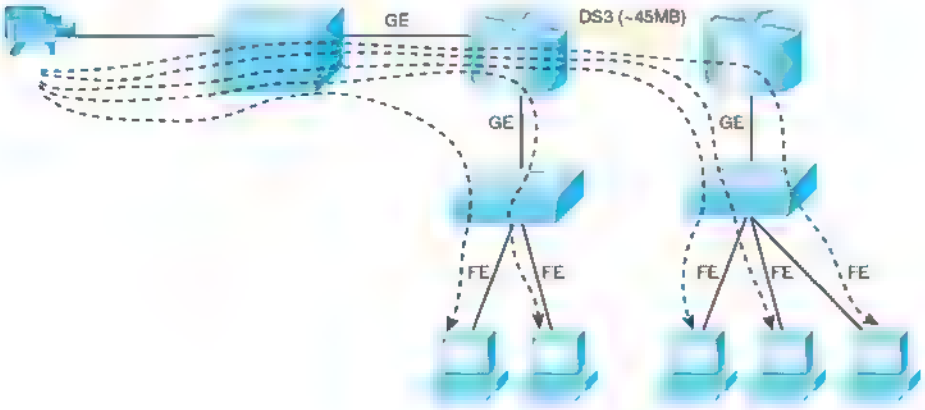


CHAPTER 7

IP MULTICAST

In contrast, a unicast is traffic from one source to one destination (see Figure 7-2).

FIGURE 7-2 Unicast Topology

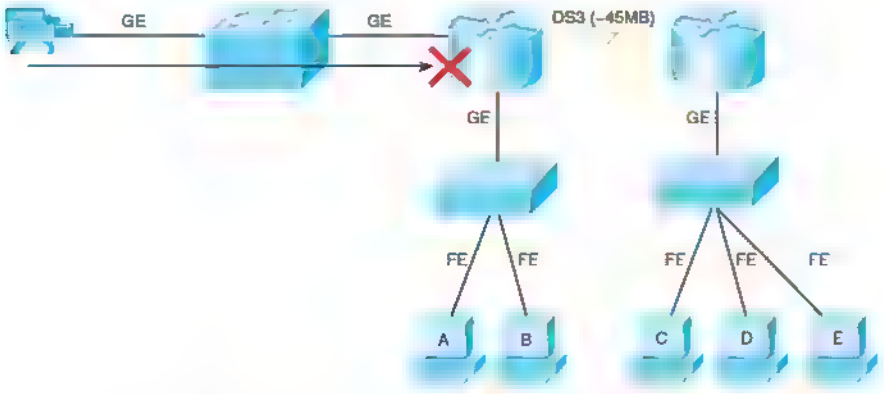


CHAPTER 7

IP MULTICAST

A broadcast is traffic from one source to all destinations (see Figure 7-3). Broadcasts are not routed!

FIGURE 7-3 Broadcasting



Some features of multicast traffic are:

- Multicast uses UDP, so reliability must be handled by the end host.
- The sending host does not know the identity of the receiving hosts; it knows just a group IP addresses.
- Group membership is dynamic. Hosts join a group, notify their upstream router, and the router begins forwarding data to them.
- Hosts can belong to more than one group.
- Hosts in a group can be located in many different places.

CHAPTER 7
IP MULTICAST

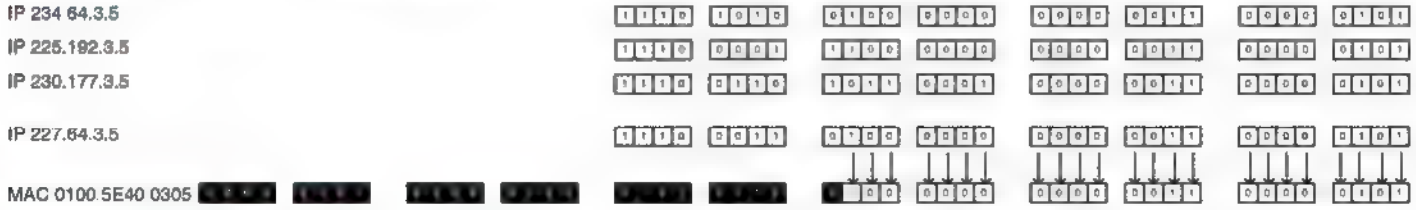
Multicast MAC Address

Multicast MAC addresses all start with the first 25 bits 01005E. The last 23 bits are the left-most bits from the IP address. Example 7-4 shows how a MAC address of 0100.5E40.0305 maps to a multicast IP address of 227.64.3.5.

FIGURE 7-4 Computing a Multicast MAC Address

The first four bits of multicast IP addresses are always 1110, and the last 23 bits map to the MAC. That leaves five bits that are dropped. Remember that this is an issue, because every multicast MAC maps to many multicast IPs! Example 7-5 shows how a MAC address of 0100.5E40.0305 could map to several different multicast IP addresses. Notice that the first two octets can vary in the IP addresses.

FIGURE 7-5 Overlapping Multicast MAC Addresses



Multicast IP Addresses

Multicasts use the IP address range 224.0.0.0 to 239.255.255.255. The first four bits of the first octet are always binary 1110, and the remaining 28 bits identify the multicast group. Some addresses are reserved:

- 224.0.0.1 is the all-hosts group.
- 224.0.0.2 is the all-routers group.

CHAPTER 7

IP MULTICAST

- The rest of the 224.0.0.0/24 range is reserved for link local protocols.
- 224.0.1.0 to 238.255.255.255 are for use over the Internet, and they are called globally-scoped addresses.
- Source-specific multicast uses 232 0.0.0/8 addresses.
- 233.0.0.0/8 is used to assign static multicast addresses for use by an organization. The second and third octets of the address are the organization's Autonomous System number. This is called GLOP, which is a combination of global and scope.
- 239.0.0.0/8 is for local use within an organization, and it is called a limited scope or an administratively scoped address.

Multicast Distribution Trees

Multicasts use two different ways to distribute data between a server and hosts:

- A source-based tree is the simplest kind. Its root is the server, and it forms branches throughout the network to all the members of the multicast group. A source tree is identified by (S,G) where S is the IP address of the server and G is the group multicast address. It creates optimal paths between the server and the hosts, but takes more router resources. Every router along the path must maintain path information for every server.
- A shared tree selects a common root called a rendezvous point (RP). The server sends traffic to the RP, which forwards it to hosts belonging to the group. The tree is identified by (*,G) where * means any source and G is the group multicast address. Shared trees use less router resources, but can result in suboptimal paths.

Reverse Path Forwarding

Multicast routers identify upstream ports (pointing to the server or RP) and downstream ports (pointing to other receivers) for each multicast group. The upstream port is found using Reverse Path Forwarding (RPF). RPF involves looking at the routing table to see which interface the router uses to send unicast traffic to that server or RP. That interface is the upstream port, or RPF port, for the multicast group. The RFP check is done every five seconds. It is used in this way:

- If a multicast packet arrives on the RPF port, the router forwards the packet out to the interfaces listed in the outgoing interface list of a multicast routing table.
- If the packet does not arrive on the RPF port, the packet is discarded to prevent loops.

CHAPTER 7

IP MULTICAST

Protocol Independent Multicast (PIM)

PIM is a protocol used between routers to keep track of where to forward traffic for each multicast group. It can use information gathered from any routing protocol. PIM can run in dense mode or sparse mode.

PIM Dense Mode

PIM dense mode uses source-based trees. When running in dense mode, PIM assumes that every router needs to receive multicasts. Any router that doesn't want to receive it must send a prune message upstream to the server. PIM dense mode is most appropriate when:

- Multicast servers and receivers are near each other.
- There are just a few servers and many receivers.
- You have a high volume of multicast traffic.
- The multicast stream is fairly constant.

PIM Sparse Mode

PIM sparse mode uses shared distribution trees. It does not assume that any routers want to receive the multicast, but instead waits to hear an explicit message from them, joining the group. Then it adds branches to the tree to reach the hosts behind those routers. PIM sparse mode uses RPs to connect hosts and servers. After the connection is made, PIM switches over to a source tree. Sparse mode is used when.

- Pockets of users are widely dispersed around the network.
- Multicast traffic is intermittent.

PIM Sparse-Dense Mode

An interface can be configured in sparse-dense mode. Then, if the router knows of an RP for its group, it uses sparse mode. Otherwise, it uses dense mode. In addition, it makes the interface capable of receiving multicasts from both sparse and dense-mode groups.

Configuring Multicast Routing and PIM

Use the following command to enable multicast routing:

```
(config)# ip multicast-routing
```

PIM mode must be configured at each interface with the following command. Configuring PIM on an interface also enables Internet Group Management Protocol (IGMP) on that interface:

```
(config-if)# ip pim {sparse-mode | dense-mode | sparse-dense-mode}
```

When using sparse mode, an RP must be specified. A router knows that it is an RP when it sees its own address in the command:

```
(config)# ip pim rp-address ip-address
```

CHAPTER 7

IP MULTICAST

Auto-RP

Auto-RP automates the discovery of RPs in a sparse or sparse-dense PIM network. RPs advertise themselves to a router designated as an RP mapping agent. The mapping agent then decides on one RP per group and sends that information to the other routers.

To configure a router as an RP, type the following:

```
(config)# ip pim send-rp-announce type number scope ttl group-
 list access list number
```

To configure a router as a mapping agent, type the following:

```
(config)# ip pim send-rp-discovery scope ttl
```

PIM Version 2

Cisco routers with recent versions of the IOS use PIM Version 2 by default. Some differences between PIM Version 1 and PIM Version 2 include:

- PIM Version 1 is Cisco proprietary, whereas PIM Version 2 is standards-based
- Both versions can dynamically map RPs to multicast groups. PIM Version 1 uses an auto-RP mapping agent, and PIM Version 2 uses a bootstrap router (BSR)
- PIM Version 1 uses a Time-to-Live value to bound its announcements, and PIM Version 2 uses a configured domain border.
- In PIM Version 2, sparse and dense mode are group properties, not interface properties.

To configure PIM Version 2, configure at least one router as a BSR, and selected routers as RPs. To configure a BSR, use the following:

```
(config)# ip pim bsr-candidate interface hash-mask length [priority]
```

To configure a router as a candidate RP, use the following:

```
(config)# ip pim rp-candidate type number ttl group-list access-
 list-number
```

IGMP

When a host wishes to join a multicast group, it sends an Internet Group Management Protocol (IGMP) message to the router. The router periodically checks for group members on each segment. There are three versions of IGMP.

IGMP Version 1

Multicast routers query each segment periodically to see if there are still hosts in multicast groups with a query sent to the all-hosts address of 224.0.0.1. One host on the segment responds. Hosts silently leave a group; the router doesn't know they are gone until it queries and no one responds.

CHAPTER 7

IP MULTICAST

IGMP Version 2

Version 2 adds explicit leave messages that hosts send when they leave a group. Queries are sent to specific multicast group addresses, not the all-hosts address.

IGMP Version 3

Hosts are able to tell the router not only which multicast groups they belong to, but also from which sources they will accept multicasts. It adds two modes for requesting membership in a multicast group:

- Include mode—The receiver lists the group or groups to which it will belong and the servers it will use.
- Exclude mode—The receiver lists the group or groups to which it will belong and the servers it will *not* use.

CGMP

Switches flood multicasts by default. Cisco Group Management Protocol (CGMP) lets a router tell a switch which hosts belong to which multicast group, so the switch can add that information to its port-to-MAC address mapping. Then when a multicast comes in, the switch forwards it only to ports that have hosts belonging to that group. CGMP is Cisco proprietary.

IGMP Snooping

IGMP snooping is another way for the switch to find out which ports have multicast hosts. When it is enabled, the switch opens all multicast packets, looking for IGMP join or leave messages. When it finds one, it records that information and uses it for forwarding multicasts. Because every multicast packet has to be opened, this can cause a performance hit on the switch.

Verifying Multicast Routing

Some commands to verify multicast routing include the following:

- **`show ip mroute`**—This shows the contents of the multicast routing table. For each group, it lists the mode, the RPF neighbor, the group identifier, and the outgoing interfaces.
- **`show ip mroute summary`**—Lists each multicast group without as much detail.
- **`show ip mroute active`**—Shows the active sources and the sending rate of each.
- **`show ip mroute count`**—Shows traffic statistics for each multicast group.
- **`show ip pim interface`**—Lists each interface doing multicasting, its PIM mode, and number of neighbors.
- **`show ip pim rp`**—Lists the RPs the router knows.
- **`show ip pim rp-hash`**—Shows the RP selected for each multicast group.
- **`show ip pim bsr`**—Lists the current BSR.

CHAPTER 8

IPv6 Introduction

IPv6 is an extension of IP with several advanced features:

- Larger address space
- Simpler header
- Autoconfiguration
- Extension headers
- Flow labels
- Mobility
- "Baked in" security

Of these, many capabilities have been backported to IPv4. The primary adoption of IPv6 will be driven by the need for more addresses. Given the growth in Internet use and the emergence of large groups of Internet users in developing countries, this is a significant requirement.

IPv6 Routing Prefix

IPv4 addresses are 32 bits long, whereas IPv6 addresses are 128 bits. IPv6 addresses are composed of the following elements (see Figure 8-1):

- The first three bits (/3) of unicast always 001.
- The next 13 bits (/16) are Top-Level Aggregator (TLA) the upstream ISP.
- The next 24 bits (/40) are the next-level aggregator or regional ISP
- Enterprises are assigned /48 and have 16 bits of subnetting.

FIGURE 8-1 RFC 2374 IPv6 Address Structure

CHAPTER 8

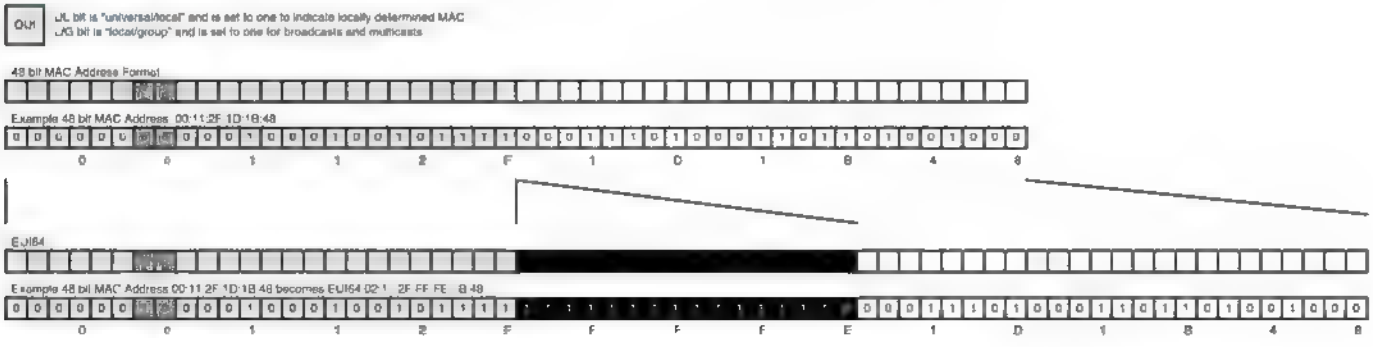
IPv6 INTRODUCTION

IPv6 Interface ID

The host portion of the address is last 64 bits. It can be assigned manually, using DHCP v6, or using stateless autoconfiguration.

An end-system uses stateless autoconfiguration by waiting for a router to advertise the local prefix. If the end system has a 64-bit MAC, it concatenates the prefix and its MAC to form an IPv6 address. If the end system has a 48-bit MAC, it flips the global/local bit and inserts 0xFFEE into the middle of the MAC. The resulting 64-bit number is called the EUI64. The prefix and EUI64 are concatenated to form the address. Figure 8-2 shows how a host uses its MAC address to create its IPv6 address.

FIGURE 8-2 EUI64



Simplified Presentation of IPv6 Address

There are two ways to shorten the representation of an IPv6 address. Take the example address 4001:0000:0001:0002:0000:0000:0000:ABCD.

- Leading zeros may be omitted. This makes the example 4001:0:1:2:0:0:0:ABCD.
- Sequential zeros may be shown as double colons once per address. This makes the example 4001:0:1:2::ABCD.

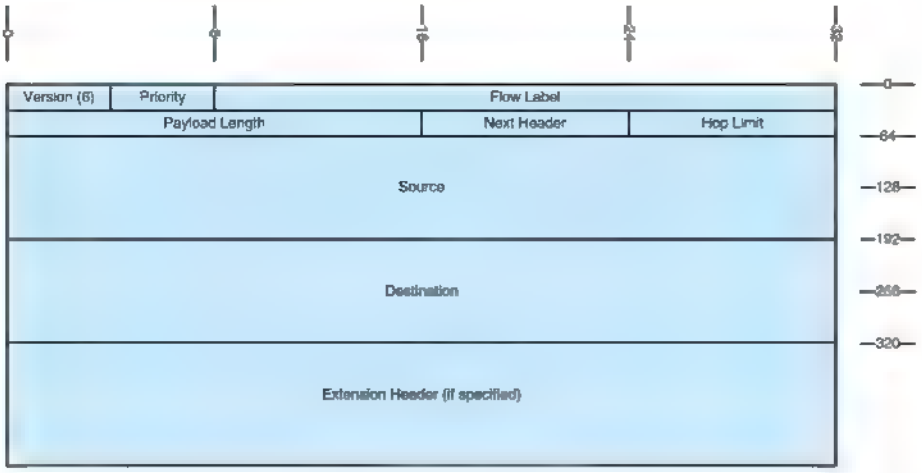
CHAPTER 8

IPv6 INTRODUCTION

IPv6 Header

The IPv6 header is similar to the IPv4 header. The largest changes have to do with the larger addresses, aligning fields to 64-bit boundaries and moving fragmentation to an extension header.

FIGURE 8-3 IPv6 Header



The fields are:

- Version—6.
- Priority—Similar to DSCP in version 4, this eight-bit field is used to describe relative priority.
- Flow—20-bit flow label allows tagging in a manner similar to MPLS.
- Length—The length of the data in the packet.
- Next Header—Indicates how the bits after the IP header should be interpreted. Could indicate TCP or UDP, or it could show an extension header.
- Hop Limit—Similar to TTL.
- Source and Destination—IPv6 addresses.

Zero or more extension headers could follow, including:

- Hop-by-hop options—Options for intermediate devices.
- Destination options—Options for the end node.
- Source routing—Specifies "way stations" that the route must include
- Fragmentation—Used to divide packets.
- Authentication—Used to attest to source. Replaces the AH header from IPSec.
- Encryption—Replaces the IPSec ESP header.

CHAPTER 8

IPv6 INTRODUCTION

Advanced Features

"Advanced" features are elements that are not available in IPv4 or have significantly changed. For instance, it's important to know that the idea of broadcasts has been abandoned and that devices will typically respond to a set of IPv6 addresses.

Specifying Destinations

IPv6 does not support broadcasts, but replaces broadcasts with multicasts. IPv6 also uses Anycast, which involves using the same address on two devices. Anycast can be used to implement redundancy and has been backported to IPv4.

Specifying Sources

Each IPv6 system must recognize the following addresses:

- Unicast address
- Link local address (FE80/10 | EUI64)
- Loopback (::1)
- All-nodes multicast (FF00::1)
- Site-local multicast (FF02::2)
- Solicited-nodes multicast (FF02::1:FF00/104)

Additionally, some systems will also use the following addresses:

- IPv4 mapped address (0::FFFF | 32-bit, IPv4 address).
- Second unicast address shared with another system (anycast).
- Additional multicast groups.
- Routers must support subnet-router anycast (all zeros EUI64)
- Routers must support local all-routers multicast (FF01::2), link-local (FF02::2), and site-local (FF05:2).
- Routers must support routing protocol multicast groups.

Renumbering

IPv6 supports easy network renumbering. A router sends out a "router advertisement" with a new prefix and a token that instructs end systems to perform stateless autoconfiguration. Hosts then recognize the command and update their addresses.

Anyone who has had to renumber a large range of IPv4 addresses can testify to what a boon this feature will be!

Mobility

IPv6 also includes better support for roaming systems. Using IPv6 Mobility, roamers keep in touch with a "home agent," which is their home router. Traffic sent to the "home address" is forwarded by the agent to the current address. The roamer then sends back a binding

update to its corresponding agent so that future traffic is sent directly to the roaming address.

IPv6 Routing

IPv6 is not enabled by default on Cisco routers. To enable IPv6 routing, the command is `Router(config)#ipv6 unicast-routing`.

After IPv6 is enabled, addresses are assigned to interfaces much like version 4:

```
Router(config-if)#ipv6 address prefix/prefix-length
```

To make this less abstract, a more complete example that shows an IPv6 implementation is shown in Example 8-1.

EXAMPLE 8-1 Enabling IPv6 Routing and Assigning Addresses

```
RouterA#configure terminal
RouterA(config)#ipv6 unicast-routing
RouterA(config)#interface fastethernet0/0
RouterA(config-if)#description Local LAN
RouterA(config-if)#ipv6 address 4001:0:1:1::2/64
RouterA(config-if)#interface serial 1/0
RouterA(config-if)#description point-to-point line to Internet
RouterA(config-if)#ipv6 address 4001:0:1:5::1/64
```

Static Routing

Static routing with IPv6 works exactly like it does with version 4. Aside from understanding the address format, there are no differences. Static routes are not currently on the BSCI test. The syntax for the IPv6 static route command is shown below, and Example 8-2 is supplied so that the command may be viewed in context as it might be applied.

```
Router(config)# ipv6 route ipv6-prefix/prefix-length {ipv6-address
 | interface-type interface-number [ipv6-address]} [administrative-
 distance] [administrative-multicast-distance | unicast |
 multicast] [tag tag]
```

EXAMPLE 8-2 Configuring Static IPv6 Routes

```
RouterA(config)#ipv6 route 4001:0:1:2::/64 4001:0:1:1::1
RouterA(config)#ipv6 route ::/0 serial1/0
```

RIPng for IPv6

RIPng is the IPv6 of RIP and is defined in RFC 2080. Like RIPv2 for IPv4, RIPng is a distance vector routing protocol that uses a hop count for its metric and has a maximum hop count of 15. RIPng also uses periodic multicast updates—every 30 seconds—to advertise routes. The multicast address is FF02::9.

RIPng is not on the BSCI exam at present, but it is presented here for completeness and to round out your appreciation for IPv6 routing and to prepare the reader for trial implementations of IPv6.

CHAPTER 8

IPv6 INTRODUCTION

There are two important differences between the old RIP and the next-generation RIP. First, RIPng supports multiple concurrent processes, each identified by a process number (this is similar to OSPFv2). Second, RIPng is initialized in global configuration mode and then enabled on specific interfaces.

Example 8-3 shows the syntax used to apply RIPng to a configuration. Notice that the syntax is very similar to traditional RIP.

EXAMPLE 8-3 RIPng

```
Router(config)#ipv6 router rip process
Router(config-rtr)#interface type number
Router(config-if)#ipv6 rip process enable
```

Like RIP for IPv4, troubleshoot RIPng by looking at the routing table (**show ipv6 route**), by reviewing the routing protocols (**show ipv6 protocols**), and by watching routing updates propagated between routers (**debug ipv6 rip**).

EIGRP

EIGRP has been expanded to support IPv6, although you'll need to verify that a specific version of IOS is capable of doing this. EIGRP for IPv6 is based on the IPv4 version. EIGRP is still an advanced distance vector routing protocol that uses a complex metric. EIGRP still has a reliable update mechanism and uses DUAL to retain fall back paths. Like EIGRP in IPv4, it sends multicast hellos every five seconds (but the multicast address is now FF02::A). EIGRP is enabled as described in the following:

```
Router(config)#ipv6 router eigrp as
Router(config-rtr)#router-id ipv4-address|ipv6-address
Router(config-rtr)#interface type number
Router(config-if)#ipv6 eigrp as
```

Like EIGRP for IPv4, troubleshoot by looking at the routing table (**show ipv6 route**), by reviewing the routing protocols (**show ipv6 protocols**), and by monitoring neighbors (**show ipv6 eigrp neighbors**). Example 8-4 shows the configuration for IPv6 EIGRP. Notice that the routing protocol must be enabled under each interface.

EXAMPLE 8-4 Configuring EIGRP for IPv6

```
RouterA#configure terminal
RouterA(config)#ipv6 unicast-routing
RouterA(config)#ipv6 router eigrp 1
RouterA(config-rtr)#router-id 10.255.255.1
RouterA(config)#interface fastethernet0/0
RouterA(config-if)#description Local LAN
RouterA(config-if)#ipv6 address 4001:0:1:1::2/64
RouterA(config-if)#ipv6 eigrp 1
RouterA(config-if)#interface serial 1/0
RouterA(config-if)#description point-to-point line to Internet
RouterA(config-if)#ipv6 address 4001:0:1:5::1/64
RouterA(config-if)#ipv6 eigrp 1
```

MP-BGP for IPv6

Multiprotocol BGP (RFC 2858) involves two new extensions to BGP4 that allow BGP to carry reachability information for other protocols,

such as IPv6, multicast IPv4, and MPLS. The extensions allow NEXT_HOP to carry IPv6 addresses and NLRI (network layer reachability information) to an IPv6 prefix.

Example 8-5 shows the BGP commands as they might be applied.

EXAMPLE 8-5 Configuring BGP IPv6 Routes

```
RouterA#configure terminal
RouterA(config)#ipv6 unicast-routing
RouterA(config)#router bgp 65000
RouterA(config-rtr)#neighbor 4001:0:1:1:5::4 remote-as 65001
RouterA(config-rtr)#address-family ipv6 unicast
RouterA(config-rtr-af)#neighbor 4001:0:1:5::4 activate
RouterA(config-rtr-af)#network 4001:0:1::/48
```

OSPFv3

OSPFv3 is one of the first routing protocols available for IPv6 and. Due to its open-standard heritage, it is widely supported in IPv6. OSPFv3 is the only routing protocol discussed on the BSCI test, so it is covered in more depth here.

OSPFv3, which supports IPv6, is documented in RFC 2740. Like OSPFv2, it is a link-state routing protocol that uses the Dijkstra algorithm to select paths. Routers are organized into areas, with all areas touching area 0.

OSPF speakers meet and greet their neighbors using Hellos, exchange LSAs (link-state advertisements) and DBDs (database descriptors), and run SPF against the accumulated link-state database.

OSPFv3 participants use the same packet types as OSPFv2, form neighbors in the same way, flood and age LSAs identically, and support the same NBMA topologies and rare techniques such as NSSA and on-demand circuits.

OSPFv3 differs from its predecessors principally in its new address format. OSPFv3 advertises using multicast addresses FF02::5 and FF02::6, but uses its link-local address as the source address of its advertisements. Authentication is no longer built in, but relies on the underlying capabilities of IPv6.

OSPFv3 LSAs

OSPFv3 and OSPFv2 use a similar set of LSAs, but version 3 has a few changes from OSPFv2. Types 3 and 4 have been slightly renamed, but still fulfill the same functionality as they did with OSPFv2. Type 8 is new and assists in discovering neighbors. Types 1 and 2 no longer pass routes. Instead they pass router IDs. Prefixes are associated as leaf objects that hang off those nodes and are advertised using Type 9, which is also new.

LSAs are sourced from the link-local address of an interface and destined for a multicast address. FF02::5 is the "all OSPF routers" address and FF02::6 is the "all OSPF DRs" address.

The OSPFv3 LSA types are collected together in Table 8-1. Notice that types one through seven exactly match their OSPFv2 predecessor, while type 8 and type 9 are new to OSPFv3.

CHAPTER 8

IPv6 INTRODUCTION

TABLE 8-1 OSPF LSA Types

LSA Type	Name	Description
1	Router-LSA	Advertise RIDs within area
2	Network-LSA	Advertise RIDs within area from DR
3	Inter-Area-Prefix-LSA	Advertise prefixes between areas
4	Inter-Area-Router-LSA	Advertise location of ASBR
5	AS-External-LSA	Advertise redistributed routes
6	Group-Membership	Multicast information
7	Type-7-LSA	Pass external routes through an NSSA
8	Link-LSA	Advertise link-local address to neighbors
9	Intra-Area-Prefix-LSA	Advertise prefixes associated with RID

Configuration

OSPF configuration is similar to RIPng and EIGRP. The routing process is created and routing properties are assigned to it. Interfaces are then associated with the process under interface configuration mode. Assuming that **ipv6 unicast-routing** and interface IP addresses are already in place, the commands to implement OSPFv3 are shown in Example 8-6.

EXAMPLE 8-6 Configuring OSPF IPv6 Routes

```
Router(config)#ipv6 router ospf process-id
Router(config-rtr)#router-id 32bit-address
Router(config-rtr)#area area range summary-range/prefix-length
Router(config-rtr)#interface type number
Router(config-if)#ipv6 ospf process area area
Router(config-if)#ipv6 ospf process priority priority
Router(config-if)#ipv6 ospf process cost cost
```

Cost may be overridden with the **ipv6 ospf cost** command as shown in Example 8-7.

The **summary-range** command is shown to demonstrate summarization.

EXAMPLE 8-7 Configuring OSPF IPv6 Routes

```
RouterA#configure terminal
RouterA(config)#ipv6 unicast-routing
RouterA(config)#ipv6 router ospf 1
RouterA(config-rtr)#router-id 10.255.255.1
RouterA(config-rtr)#area 1 range 4001:0:1::/80
RouterA(config-rtr)#interface fastethernet0/0
RouterA(config-if)#description Local LAN
RouterA(config-if)#ipv6 address 4001:0:1:1::2/64
RouterA(config-if)#ipv6 ospf 1 area 1
RouterA(config-if)#ipv6 ospf cost 10
RouterA(config-if)#ipv6 ospf priority 20
RouterA(config-if)#interface serial 1/0
RouterA(config-if)#description multi-point line to Internet
RouterA(config-if)#ipv6 address 4001:0:1:5::1/64
RouterA(config-if)#ipv6 ospf 1 area 1
RouterA(config-if)#ipv6 ospf cost 10
RouterA(config-if)#ipv6 ospf priority 20
```

Troubleshooting

Troubleshoot OSPFv3 just like OSPFv2. Start by looking at **show ipv6 route** to verify routes have been advertised. Assuming the route is in the routing table, test reachability using **ping ipv6**. You can also look at the ospf setup using `show ipv6 ospf 1 interface`, `show ipv6 ospf`, or `show ipv6 ospf database`.

Integrating IPv4 and IPv6

There are several strategies for migrating from IPv4 to IPv6. Each of these strategies should be considered when organizations decide to make the move to IPv6 because each has positive points to aiding a smooth migration. It should also be said that there does not have to be a global decision on strategy—your organization may choose to run dual-stack in the U.S., go completely to IPv6 in Japan, and use tunneling in Europe. The transition mechanisms include:

- Dual stack—Running IPv6 and IPv4 concurrently.
- IPv6 to IPv4 tunneling (6-to-4)—Routers that straddle the IPv4 and IPv6 worlds to encapsulate the IPv6 traffic inside IPv4 packets.
- Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)—This protocol is similar to 6-to-4, but it treats the IPv4 network as an NBMA network.
- Teredo/Shipworm—Encapsulates IPv6 packets in IPv4/UDP segments.

NAT-PT, ALG, and BIA/BIS

Instead of replacing IPv4, there are several ways to coordinate the functioning of IPv4 and v6 concurrently. NAT-protocol translation is an example of this coexistence strategy. NAT-PT maps IPv6 addresses to IPv4 addresses. If IPv6 is used on the inside of your network, a NAT-PT device will receive IPv6 traffic on its inside interface and replace the IPv6 header with an IPv4 header before sending it to an outside interface. Reply traffic will be able to follow the mapping backward to enable two-way communication.

NAT-PT is able to interpret application traffic and understand when IP information is included in the application data.

It is also possible to connect IPv4 and IPv6 routing domains using application-level gateways (ALG), proxies, or Bump-in-the-API (BIA) and Bump-in-the-Stack (BIS), which are NAT-PT implementations within a host.

CCNP BSCI Quick Reference Sheets

Brent Stewart
Denise Donohue

Published by: Cisco Press

800 East 96th Street

Indianapolis, IN 46240 USA

Printed in the United States of America 1 2 3 4 5 6 7 8 9 0

First Printing October 2006

ISBN: 1-58705-312-8

Warning and Disclaimer

This digital short cut is designed to provide information about networking. Every effort has been made to make this digital short cut as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an "as is" basis. The authors, Cisco Press, and Cisco Systems, Inc., shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the authors and are not necessarily those of Cisco Systems, Inc.

Trademark Acknowledgments

All terms mentioned in this digital short cut that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc., cannot attest to the accuracy of this information. Use of a term in this digital short cut should not be regarded as affecting the validity of any trademark or service mark.

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this digital short cut or otherwise alter it to better suit your needs, you can contact us through e-mail at feedback@ciscopress.com. Please make sure to include the digital shortcut title and ISBN in your message.

We greatly appreciate your assistance.

Corporate and Government Sales

Cisco Press offers excellent discounts on this digital short cut when ordered in quantity for bulk purchases or special sales. For more information please contact: U.S. Corporate and Government Sales 1-800-382-3419 corpsales@pearsontechgroup.com

For sales outside the U.S. please contact: International Sales international@pearsoned.com

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Asia Pacific Headquarters
Cisco Systems, Inc.
168 Robinson Road
#28-01 Capital Tower
Singapore 068912
www.cisco.com
Tel: +65 6317 7777
Fax: +65 6317 7799

Europe Headquarters
Cisco Systems International BV
Haarlerbergpark
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands
www-europe.cisco.com
Tel: +31 0 800 020 0791
Fax: +31 0 20 357 1100

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**