

Routing updates are sent to all nearest neighbors

Although Figure 5.4 is concerned with updates related to the problem with the connection to Network A, remember that the routers send updates to all their nearest neighbors. So, while Router 1 is updating Router 2, Router 2 also sends an update to Router 1 as well as Router 3 when it sends its updated routing table.

FIGURE 5.5

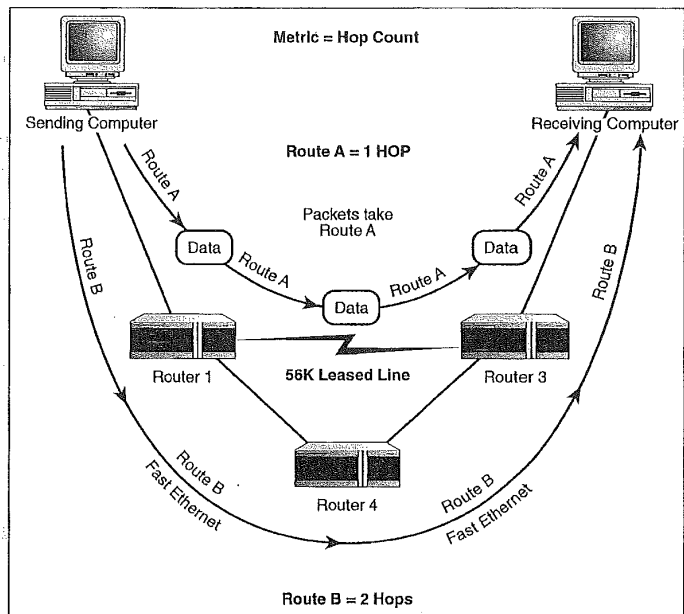
Routing algorithms use a metric, such as hop count, to determine the optimum path for data packets.

Hybrid routing protocols

Some routing protocols, such as OSPF, are considered hybrids because they use distance-vector and link-state information to update routing tables.

Routing algorithms use a *metric* to determine the suitability of one path over another. The metric can be several different things such as the path length, the actual cost of sending the packets over a certain route, or the reliability of a particular route between the sending and receiving computers.

For example, RIP, a distance vector routing protocol, uses *hop count* as its metric. A hop is the movement of the packets from one router to another router. If two paths are available to get the packets from one location to another, RIP will choose the most desirable path based on the smallest number of hop counts. Figure 5.5 shows an internetwork where two paths are possible for the routing of packets between the sending and receiving computers. Because Route A requires only one hop, it is considered the optimum route for the packets.



The problem with routing protocols that use only one metric (such as hop count) is that they become very single minded in their pursuit of the best route for a particular set of packets. RIP, for example, doesn't take the speed or reliability of the lines into account when it chooses the best path, just the number of hops. So, as shown in Figure 5.5, even though Route A is the best path according to the number of hops (and RIP), you are forced to route your packets over a slower line (the 56-kilobit leased line). This line is not only slow, it also costs you money. Route B is actually over wire that the company owns (part of the network infrastructure) and is actually a faster medium (fast Ethernet at 100Mbps). However, when you use a routing protocol that uses hop count as the metric it will choose Route A.

To overcome the lack of flexibility provided by hop count as a metric, several other routing protocols that use more sophisticated metrics are available. For example, the Interior Gateway Routing Protocol (IGRP) is another distance-vector routing protocol that can actually use 1 to 255 metrics depending on the number set by the network administrator. These metrics can include bandwidth (the capacity of the lines involved), load (the amount of traffic already being handled by a particular router in the path), and communication cost (packets are sent along the least expensive route). When several routing metrics are used together to choose the path for packets, a much more sophisticated determination is made. For example, in the case of Figure 5.5 a routing protocol that uses metrics other than hop counts (such as communication cost) would choose the route with more hops but less cost to move the packets to their destination.

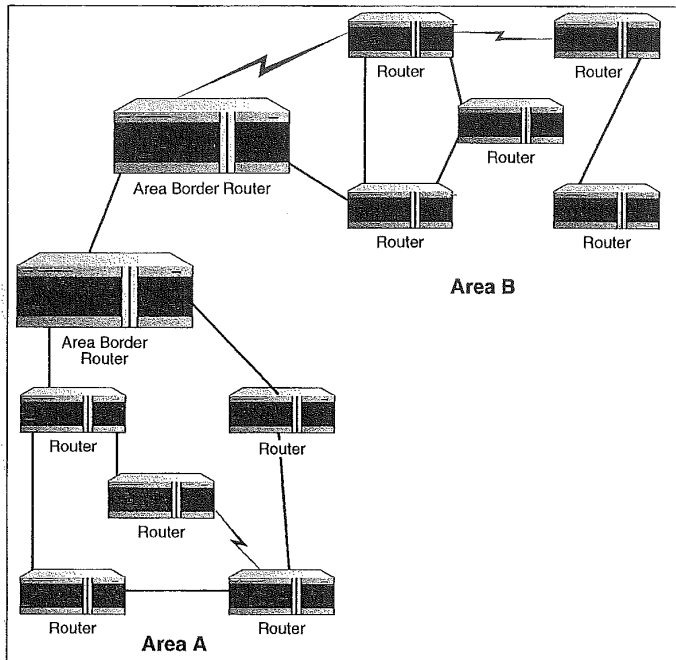
Types of Routing Protocols

Real-world internetworks (particularly those for an entire enterprise) will consist of several routers that provide the mechanism for moving packets between the various subnets found on the network. To move packets efficiently it's not uncommon to divide several connected routers into subsets of the internetwork. A subset containing several member routers is referred to as an *area*. When several areas are grouped into a higher-level subset, this organizational level is called a *routing domain*.

Figure 5.6 shows an internetwork divided into areas. Each area is terminated by a high-end router called a border router (or core router as mentioned in the sidebar). The two border routers are connected to each other, which, in effect, connects the two routing domains (or autonomous systems on an IP internetwork).

FIGURE 5.6

Internetworks can be divided into areas that are connected by area border routers.



IP internetworks can be divided into routing domains

In cases where link-state routing protocols are used that require greater memory and processing capabilities from the routers on the network, it's not uncommon to divide the internetwork into routing domains. In IP networks, a routing domain is referred to as an *autonomous system*. Routing domains (or if you prefer, autonomous systems) are typically connected by a higher-end router called a *border router* or *core router*.

The fact that internetworks can be divided into logical groupings such as routing domains (or autonomous systems) gives rise to two different kinds of routing protocols: routing protocols that provide the routing of packets between routers in a routing domain and routing protocols that provide the routing of packets between routing domains.

Interior Gateway Protocols (IGPs) provide the routing of packets within the routing domain. IGPs such as RIP or IGRP would be configured on each of the routers in the router domain.

Protocols that move data between the routing domains are called *Exterior Gateway Protocols* (EGPs). Examples of EGPs are Border Gateway Protocol (BGP) and Exterior Gateway Protocol (EGP).

Interior Gateway Protocols

The Interior Gateway Protocols consist of distance-vector and link-state routing protocols. Several different IGPs are available and vary on the number of metrics used to determine optimum routing paths. The oldest IGP is the Routing Information Protocol and is discussed in the following section, along with some of the other commonly used IGPs.

Routing Information Protocol

Routing Information Protocol (RIP) is a distance-vector, IP-routing protocol that uses hop count as its metric. And although it is the oldest IGP, RIP is still in use.

RIP sends out a routing update message every 30 seconds (by Cisco default), which consists of the router's entire routing table. RIP uses the User Datagram Protocol—UDP—(part of the TCP/IP stack) as the encapsulation method for the sending of routing advertisements.

RIP is limited, however, in that the maximum number of hops that it will allow for the routing of specific packets is 15. This means that RIP is fine for smaller, homogenous internetworks, but doesn't provide the metric flexibility needed on larger networks.

SEE ALSO

➤ For information on configuring RIP on a Cisco router, see page 202.

Interior Gateway Routing Protocol

The *Interior Gateway Routing Protocol* (IGRP) was developed by Cisco in the 1980s. IGRP is a distance-vector routing protocol.

IGRP uses a composite metric that takes into account several variables; it also overcomes certain limitations of RIP, such as the hop count metric and the inability of RIP to route packets on networks that require more than 15 hops.

A real-world example

If you have a small or medium-sized company that has an internetwork, your entire network could be considered a routing domain. It would use Interior Gateway Protocols such as RIP or IGRP to move packets between the subnets or areas in the domain. Your connection to the Internet (the global internetwork) would be managed by an Exterior Gateway Protocol such as Border Gateway Protocol. More about these individual routing protocols is provided in the remainder of the chapter.

Implementing RIP

RIP is an IP network routing protocol. The logical division of IP networks is the subnet. Proper subnetting and a consistent use of IP subnet masks is crucial when using RIP on your routers. Subnetting and IP subnet masks will be discussed in Chapter 11.

IGRP is all Cisco

Because IGRP was developed by Cisco and remains a Cisco proprietary protocol, IGRP will only be available on Cisco routers. In comparison, RIP is a universal routing protocol that you will find on IP networks whether they are routed using Cisco boxes or products from another vendor such as 3Com.

Enhanced IGRP builds on IGRP's capabilities

Cisco now provides an enhanced version of IGRP called Enhanced IGRP (EIGRP). Although it uses the same metrics as IGRP, EIGRP provides updates at irregular intervals to reflect that a particular metric such as load or the network topology has changed. And because router updates only include routing information that has changed, EIGRP is less of a bandwidth hog when compared to IGRP.

IGRP (when compared to RIP) also employs a longer time period between routing updates and uses a more efficient format for the update packets that are passed between routers. IGRP also supports the use of *autonomous systems* (similar to the areas discussed earlier in the chapter), so routers running IGRP can be sequestered into domains where the router traffic in a particular domain remains local. This cuts down on the amount of router broadcast communications using up valuable bandwidth throughout the entire internet-work.

IGRP's metric consists of a composite that takes into consideration bandwidth, delay, load, and reliability when determining the best route for data moving from a sending node to a particular destination node. The following list describes how each of these network parameters is viewed by IGRP when the routing algorithm is used to build or update a router's routing table:

- **Bandwidth** is the capacity of a particular interface in kilobits. A serial interface may have a bandwidth of 100,000 kilobits (this would be a serial interface connected to an ATM switch, which typically supplies this amount of bandwidth). Unfortunately, the bandwidth of a particular interface isn't measured dynamically (measuring the actual bandwidth available at a particular time) but set statically by the network administrator using the `bandwidth` command. More about setting serial interfaces will be discussed in Chapter 15, "Configuring WAN Protocols."
- **Delay** is the amount of time it takes to move a packet from the interface to the intended destination. Delay is measured in microseconds and is a static figure set by the network administrator using the `delay` command. Several delays have been computed for common interfaces such as Fast Ethernet and IBM Token Ring. For example, the delay for a Fast Ethernet interface is 100 microseconds.
- **Reliability** is the ratio of expected-to-received keepalives on a particular router interface. (*Keepalives* are messages sent by network devices to tell other network devices, such as routers, that the link between them still exists.) Reliability is measured dynamically and is shown as a fraction when the `show interface` command is used on the router. For example, the fraction 255/255 represents a 100% reliable link.

- *Load* is the current amount of data traffic on a particular interface. Load is measured dynamically and is represented as a fraction of 255. For example, 1/255 would be an interface with a minimal amount of traffic, whereas 250/255 would be a fairly congested interface. Load can be viewed on the router using the `show interface` command.

As you can see, IGRP takes a lot of information into consideration when it uses its algorithm to update a router's routing table. It is often implemented in larger internetworks where RIP would be ineffectual.

SEE ALSO

➤ For information on configuring IGRP on a Cisco router, see page 204.

Open Shortest Path First Routing Protocol

Open Shortest Path First (OSPF) is a link-state protocol developed by the Internet Engineering Task Force (IETF) as a replacement for RIP. Basically, OSPF uses a shortest path first algorithm that enables it to compute the shortest path from source to destination when it determines the route for a specific group of packets.

OSPF employs the Hello Protocol as the mechanism by which routers identify their neighbors. Hello packet intervals can be configured for each interface on the router that is using OSPF (the default is every 10 seconds). The command for adjusting the Hello Interval is `ip ospf hello-interval`.

OSPF routing networks can also take full advantage of the autonomous systems feature on large IP networks (also discussed in this chapter as areas and domains), which keeps link-state advertisement of member routers local to a particular autonomous system. Area border routers are used to connect the various autonomous system areas into one internetwork.

Exterior Gateway Protocols

As mentioned earlier, Exterior Gateway Protocols (EGPs) are used to route traffic between autonomous systems (routing domains). *Border Gateway Protocol (BGP)* is a commonly used routing protocol for inter-domain routing. It is the standard EGP for the Internet.

Why does 255 keep popping up?

You probably noticed that both reliability and load use the number 255 as the measure of a completely reliable or completely congested (in terms of load) router interface. This is because, even though hop count isn't one of IGRP's metrics, IGRP can only move packets on internetworks that consist of 255 hops or fewer. This means that theoretically packets could be moved between 255 devices as the packets move from source to destination. This is why the number 255 is used in the measurement of reliability and load.

BGP handles the routing between two or more routers that serve as the border routers for particular autonomous systems. These border routers are also referred to as *core routers*. Basically, these core routers serve as neighbors and share routing table information with each other. This enables the core routers to build a list of all the paths to a particular network.

BGP uses a single metric to determine the best route to a particular network. Each link is assigned an arbitrary number that specifies the degree of preference for that link. The preference degree number for a particular link is assigned by the network administrator.

part



ROUTER DESIGN AND BASIC CONFIGURATION

Understanding Router Interfaces 99

Setting Up a New Router 111

Basic Router Configuration 123

Working with the Cisco IOS 141

6

7

8

9

chapter

6

Understanding Router Interfaces

Router Interfaces

LAN Interfaces

Serial Interfaces

Logical Interfaces



Router Interfaces

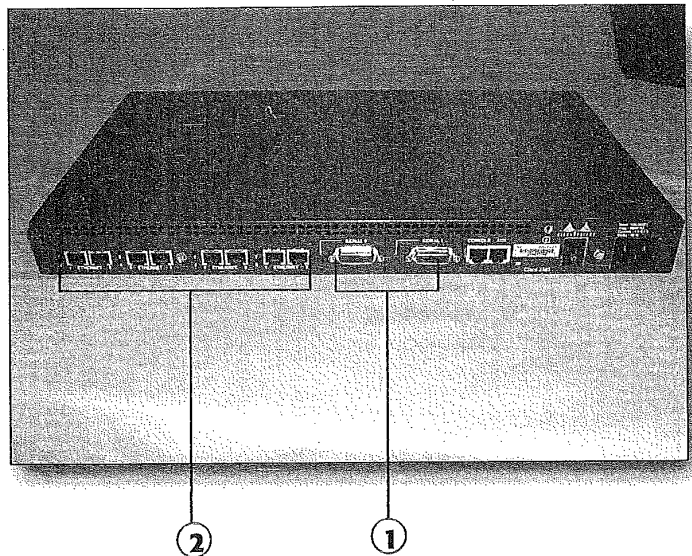
A router interface supplies the physical connection between the router and a particular network medium type. Cisco interfaces are often referred to as *ports*, and each port will be physically designed to appropriately connect the type of network technology it is supposed to serve. For example, a *LAN interface*, such as an Ethernet port on the router, will consist of a female RJ-45 connector (which is connected to an Ethernet hub using a twisted-pair cable with male RJ-45 connectors on either end).

Built-in ports are designated by their connection type followed by a number. For example, the first Ethernet port on a router would be designated as E0. The second Ethernet port would be E1, and so on (in some cases, the Ethernet port will be set up as a hub, such as on the 2505 router). Serial ports are designated likewise with the first serial port being S0. Figure 6.1 shows two serial ports and their numeric designation on a Cisco 2505 router and the Ethernet 0 hub ports (1 through 8).

FIGURE 6.1

Ports such as serial ports are designated by a number, starting with 0. Ethernet interfaces can be set up as Hub ports.

- ① Serial ports
- ② Ethernet hub ports



Cisco routers such as those in the 2500 Series family basically are off-the-shelf routers that come with a predetermined number of LAN, WAN, and serial ports. Higher-end routers like the Cisco 4500 are modular and actually contain empty slots that can be filled with several different interface cards.

Not only are different interface cards available (such as LAN versus WAN), but the number of ports on the card can also be selected. For example, one of the three empty slots on the 4500 router can be filled with an Ethernet card that contains six Ethernet ports. Figure 6.2 shows the Cisco ConfigMaker hardware configuration screen for the Cisco 4500 router (you will work with ConfigMaker in Chapter 16). Three slots are available (shown on the right of the screen) and can be filled with several different cards (listed on the left of the screen).

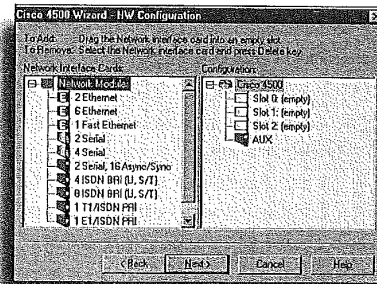


FIGURE 6.2

Modular routers such as the 4500 allow you to fill empty slots with different interface cards.

Modular routers (like the 4500) designate their ports by connection type, followed by slot number, followed by port number. For example, the first Ethernet port on an Ethernet card placed in the router's first slot would be designated as Ethernet 1/0 (the slot is designated first, followed by the port number).

Viewing the interfaces (and their status) on a particular router is handled by the `show interfaces` command. Figure 6.3 shows the results of the `show interfaces` command on a 2505 router that has one Ethernet port (E0) and two serial ports (S0 and S1). The status of the various ports is related to whether the ports have been connected (physically to the internetwork) and whether they have been configured.

FIGURE 6.3

The interfaces on a router can be quickly checked with the `show interfaces` command.

```

router2>show interfaces
Ethernet0 is up, line protocol is up, using hub 0
Hardware is Tance, address is 0010.7b3a.50b3 (bia 0010.7b3a.50b3)
Internet address is 10.40.1.0 255.248.0.0
MTU 1500 bytes, BW 1000000 bit, DLY 1000 usec, rely 255/255, load 1/255
Encapsulation ARPA, loopback not set, keepalive set (10 sec)
ARP type: ARPA, ARP Timeout 4:00:00
Last input never, output 0:00:00, output hang never
Last clearing of "show interface" counters never
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runs, 0 giants
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
1089 packets output, 102933 bytes, 0 underruns
0 output errors, 0 collisions, 1 interface resets, 0 restarts
0 output buffer failures, 0 output buffers swapped out
Serial10 is down, line protocol is down
Hardware is HD64570
Internet address is 10.32.3.0 255.240.0.0
MTU 1500 bytes, BW 38 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runs, 0 giants
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 310 interface resets, 0 restarts
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
DCD-down DSR-down DTR-down RTS-down CTS-down
Serial11 is down, line protocol is down
Hardware is HD64570
--More--

```

High-end routers use VIP cards

Extremely high-end routers such as the Cisco 12000 series use Versatile Interface Processor cards. Each VIP card can have two available slots for interface cards. These types of routers are custom built, and their interfaces would be mapped directly to your interface needs on a large internetwork. Routers like those in the 12000 series also supply you with hot swappable interfaces, which allow you to add additional cards without disruption to the router (and to the network that is serviced by the router).

Configuring a particular interface depends on the type of network protocol used by the network to which the interface port is connected. For example, an Ethernet port connected to an IP network will be configured for the routing of IP. An Ethernet port connected to an AppleTalk network will be configured for AppleTalk routing. Interface configuration is covered in Chapters 11, "Configuring IP Routing," 12, "Routing Novell IPX," and 13, "Routing AppleTalk."

SEE ALSO

➤ *Connecting LAN and serial ports to network media is discussed on page 119.*

LAN Interfaces

Cisco routers support several commonly used LAN networks. The most common LAN router interfaces are Ethernet, Fast Ethernet, IBM Token Ring, and Fiber Distributed Data Interface (FDDI).

All these LAN protocols mentioned embrace the same Data Link layer physical addressing system (the MAC hardware address on a NIC or the MAC hardware address found on the controller of the router interface). These addresses are unique for each device.

Each LAN technology is discussed briefly in the list that follows:

- Ethernet provides a network throughput of up to 10Mbps. It is a passive network architecture that uses carrier sense multiple access with collision detection (CSMA/CD) as its network access strategy. A Cisco router can be used to segment an Ethernet network into logical subnets (such as IP subnets). Typically, a router is connected to an Ethernet network using UTP cable with an RJ-45 connector. Some routers, such as the Cisco 2505, provide direct hub connections that can be used to directly connect workstations to hub ports built into the router (see Figure 6.4).

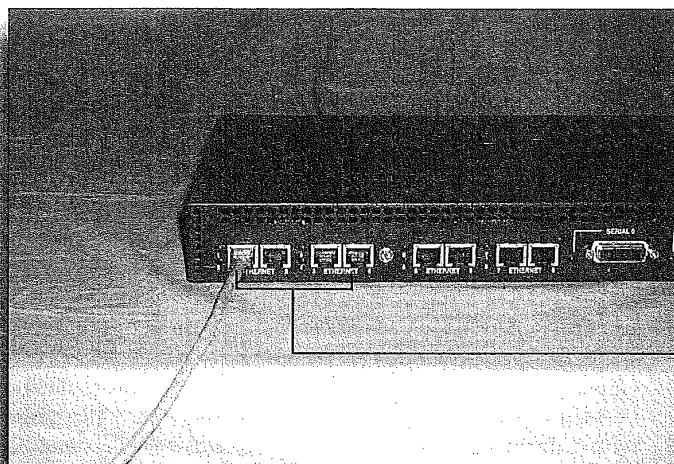


FIGURE 6.4

The 2505 router provides hub ports as the connection to the Ethernet interface built into the router.

① Hub ports

①

- Fast Ethernet operates at speeds of up to 100Mbps. It uses the same access strategy as regular Ethernet (CSMA/CD) and runs on UTP cable (as does regular Ethernet). Fast Ethernet does require special interfaces on routers (Fast Ethernet Interfaces) and Fast Ethernet NICs on nodes. Any hubs used as part of the network topology must also be Fast Ethernet hubs.
- Token Ring is a proprietary network architecture developed by IBM. Token Ring networks run on a logical ring (the ring is provided internally by the multi access units (MAUs) that are used to connect the various nodes on the network. Token Ring

Routers can handle more than one network protocol

Remember that routers can route more than one network protocol (such as IP and IPX/SPX) at the same time. They also can run more than one routing protocol at a time (which may be dictated by the network protocols that must be routed). Routed and routing protocols are discussed in Chapter 5, "How a Router Works."

Checking the MAC address of a router interface

Router LAN interfaces have unique MAC addresses just like network interface cards. To view the MAC address for your router interfaces, use the `show interface` command (refer to Figure 6.3).

networks use token passing as the network access strategy.

Routers used on Token Ring networks must contain a special Token Ring interface for connection to the network. Parameters such as the speed of the ring (either 4Mbps or 16Mbps) must be set on the router's Token Ring interface so the throughput speed matches that of the Token Ring network.

- FDDI is a token passing network that uses two redundant rings (passing tokens in opposite directions) as a fault tolerance method (*fault tolerance* is keeping the network up and running when one of the rings breaks down). FDDI, which is often employed as a fiber-optic backbone for larger networks or municipal area networks (MANs), can provide network throughput of up to 100Mbps. Routers used on FDDI networks must have an FDDI interface.

All the LAN protocols require a matching interface on the router that serves them. For example, a Token Ring network can only be attached to a router with the appropriate Token Ring interface. Specifications for some of the routers built by Cisco are discussed in Appendix C, "Selected Cisco Router Specifications." You can also view the various specifications of Cisco routers on Cisco's Web site at www.cisco.com. It is obviously important that when you plan your internetwork, you purchase a router that will provide you with all the necessary interfaces that your various LAN connections will require. Figure 6.5 shows the diagram of a network where several different LAN architectures have been connected using routers (the diagram is actually based on the network map of a real company's internetwork).

SEE ALSO

- MAC addresses are discussed on page 41.
- For more information on LAN architectures such as Ethernet or FDDI, see page 25.

Serial Interfaces

Serial router interfaces provide a way to connect LANs using WAN technologies. WAN protocols move data across asynchronous and synchronous serial interfaces (on routers), which are connected via leased lines and other third-party connectivity technologies.

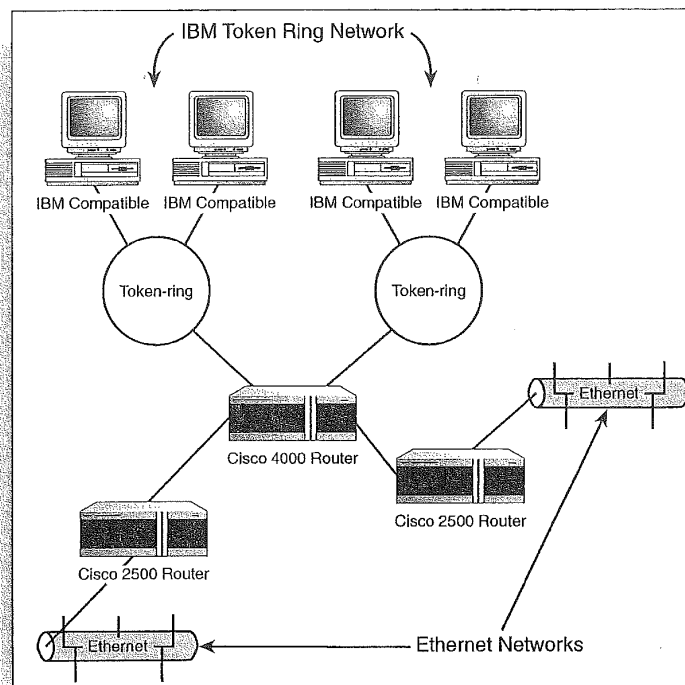


FIGURE 6.5
Routers can provide the connection between different LAN architectures such as the Token Ring and Ethernet networks shown here.

Some of the commonly used WAN Data Link layer technologies are High Level Data Link Control (*HDLC*), X.25, Frame Relay, Integrated Services Digital Network (ISDN), and Point to Point Protocol (PPP). All the WAN protocols discussed are configured on particular router interfaces (such as a serial interface or an ISDN interface) when the router is in the configuration mode. The actual command sets and the ins and outs of configuring WAN protocols on a Cisco router are discussed in Chapter 15, "Configuring WAN Protocols."

- HDLC is a Data Link layer protocol that provides the encapsulation of data transferred across synchronous data links. This means that a device such as a *DCE* (Data Communication Equipment) provides a connection to the network and provides a clocking signal that synchronizes the transfer of data between

the two ends of the serial link. Serial ports on a router are connected to a modem or other CSU/DSU device via special cables such as a V.35 cable. HDLC is the default WAN protocol for Cisco routers. Cisco's HDLC implementation is proprietary, however, and will not communicate with other vendor's HDLC (this is why trying to mix routers from different vendors such as Cisco and 3Com can be a real nightmare). HDLC is considered a point-to-point protocol and provides a direct connection between sending and receiving devices (such as two routers).

- Point to Point Protocol (PPP) is another Data Link layer point-to-point protocol supported by Cisco routers. It isn't proprietary, so it can be used to connect Cisco routers to internetworking devices from other vendors. PPP actually operates in both synchronous and asynchronous modes (meaning it can provide either encapsulation type). A flag (which is actually several bits inserted into the data stream) is used to signify the beginning or end of a frame or datagram of information flowing across the PPP connection. PPP can be used for connecting IP, AppleTalk, and IPX networks over WAN connections.

PPP is configured on the serial port of the router that provides the connection to a leased line or some other WAN connection. You may already be familiar with PPP because it is the protocol used to connect workstations to Internet service providers over analog phone lines via a modem.

- X.25 is a packet-switching protocol for use over public switched telephone networks. Data is passed along the switched network using virtual circuits (such as permanent virtual circuits). X.25 is a slow protocol when compared to newer WAN technologies like Frame Relay because it provides a great deal of error checking (which was a must when X.25 was first implemented several years ago over fairly low-grade telephone lines). X.25 is typically implemented between a DTE device and a DCE device. The DTE is typically a Cisco router, and the DCE is the X.25 switch owned by the public switched network. Figure 6.6 shows how two routers would be connected across an X.25 serial connection.

Synchronous versus asynchronous communications

Synchronous serial connections use a clocking device that provides the precise timing of the data as it moves from sending to receiving end across a serial connection.

Asynchronous connections rely on start and stop bits to make sure that the data is completely received by the destination interface.

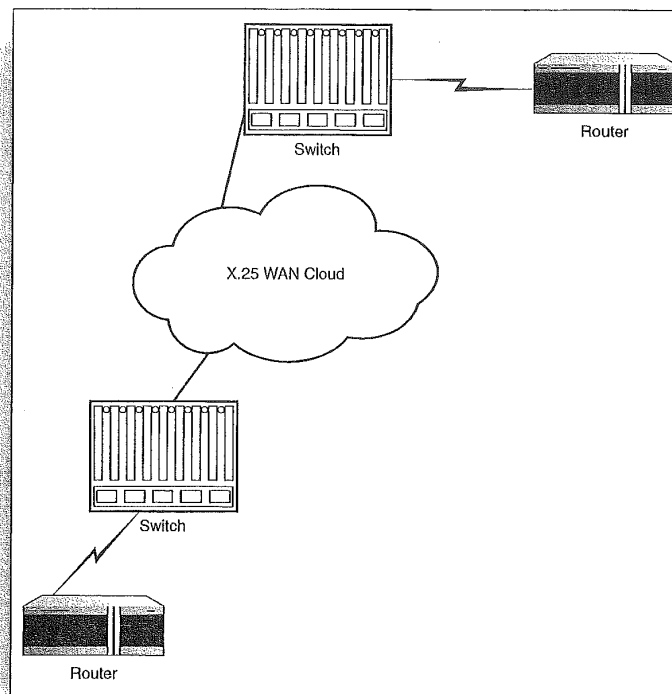


FIGURE 6.6

Routers can be connected to switching equipment and configured for WAN protocols such as X.25.

- Frame Relay is a packet switching Data Link layer protocol that was originally developed for use over ISDN connections. It has now replaced X.25 as the protocol of choice over switched networks, and it uses virtual circuits to define a route between two devices (such as two routers) communicating over the WAN. In a Frame Relay connection, a DTE such as a router is attached to a DCE such as a CSU/DSU (most CSU/DSUs can be connected to the router using a V.35 serial cable). Or the router can be connected directly to the phone company's switching equipment. Frame Relay-based WANs looked similar to the X.25 packet switching network depicted in Figure 6.6.
- Integrated Services Digital Network (ISDN) uses digital technology to move data, voice, and video over existing phone lines. It is an asynchronous WAN protocol. ISDN requires that the

X.25 and Frame Relay use serial port connections

Both of the packet switching WAN protocols, X.25 and Frame Relay, have their hardware components (DCEs such as CSU/DSUs or phone network switches) connected to the router via one of the router's serial ports. Even a low-end router such as the 2505 (as simple as it is) can have its serial interfaces configured for these WAN technologies, whereas in the case of ISDN, it is typically connected to an ISDN port that is provided by a specific model of router.

network be connected to the phone line using terminal equipment that is commonly referred to as an ISDN modem.

However, Cisco routers can be purchased that have a BRI interface (BRI stands for Basic Rate Interface) included on the router. The BRI interface is then connected directly to the phone lines. In cases where your router doesn't have the BRI port, you will have to connect one of the existing serial ports to an ISDN modem (or buy a new router).

SEE ALSO

➤ *WAN protocols and how they work are discussed in greater detail on page 53.*

Logical Interfaces

Before we conclude our discussion of router interfaces, we must take a look at logical interfaces. A *logical interface* is a software-only interface and is created using the router's IOS. Cisco's IOS is explored in Chapter 9, "Working with the Cisco IOS."

Logical interfaces don't exist as actual hardware interfaces on the router. You can think of logical interfaces as *virtual interfaces* that have been created with a series of router software commands.

These virtual interfaces can be viewed by devices on the network as real interfaces, just as a hardware interface such as a serial port is a real interface. You can configure different types of logical interfaces on a router including Loopback interfaces, Null interfaces, and Tunnel interfaces.

Loopback Interfaces

A *Loopback interface* is a software-only interface that emulates an actual physical interface on the router. Loopbacks are typically configured on a high-end router that serves as the core router between two corporate internetworks or between a corporate network and the Internet. Routers serving as core routers will be configured with an exterior gateway protocol such as Border Gateway Protocol that routes the packets between the two separate internetworks.

Logical interfaces on high-end routers

Logical interfaces are a little bit beyond the scope of this book but you should be familiar with their existence. They are sometimes configured on higher-end routers, such as the Cisco 4000 and 7500 series routers, which serve as central site access routers and core routers on very large internetworks. Logical interfaces can be used on higher-end routers as clever ways to either access or restrict traffic to a particular portion of the internetwork.

Because the router serves as such an important link between internetworks, you don't want it dumping data packets if a particular physical interface goes down on the router. So the Loopback virtual interface is created and configured as the termination address for the Border Gateway Protocol (BGP) sessions. In this way the traffic is processed locally on the router, which assures you that the packets get to their final destination.

Null Interfaces

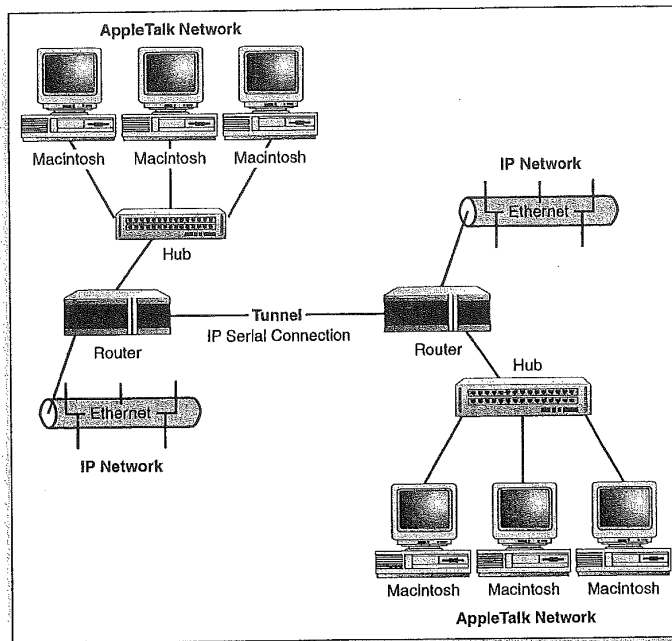
Another logical interface is the *Null interface*. It is set up on a router using the appropriate router commands and serves as a brick wall that can be used to keep out certain traffic. For example, if you don't want traffic from a particular network to move through a particular router (but move through the internetwork by other routes) you can configure the Null interface so that it receives and dumps any packets that the network sends to the router. Normally Access lists (discussed in Chapter 14, "Filtering Router Traffic with Access Lists") are used to filter traffic on an internetwork and define valid routes for certain networks. The Null interface is pretty much a sledgehammer approach to a process that is normally handled with jeweler's tools.

Tunnel Interfaces

A *Tunnel interface* is another logical interface that can be used to move packets of a particular type over a connection that doesn't typically support these types of packets. For example, a Tunnel interface can be set up on each of two routers that are responsible for routing AppleTalk packets from their LANs. These two routers are connected by a serial connection (see Figure 6.7). The Tunnel interface can be configured to route IP. And although AppleTalk would not be typically routed over an IP interface, the AppleTalk packets are encapsulated (stuffed in a generic envelope) and then moved across the Tunnel as if they were IP packets. Cisco routers provide the Generic Route Encapsulation Protocol (GRE), which handles the encapsulation of packets moved over a Tunnel interface.

FIGURE 6.7

AppleTalk packets are routed over a virtual IP Tunnel.



chapter

7

Setting Up a New Router

Becoming Familiar with Your Router

Cisco Router Design

Connecting the Console

Configuring the Router Console

Working with the Terminal Emulation
Software

Connecting the Router to the Network

A Final Word on Physical Router
Connections



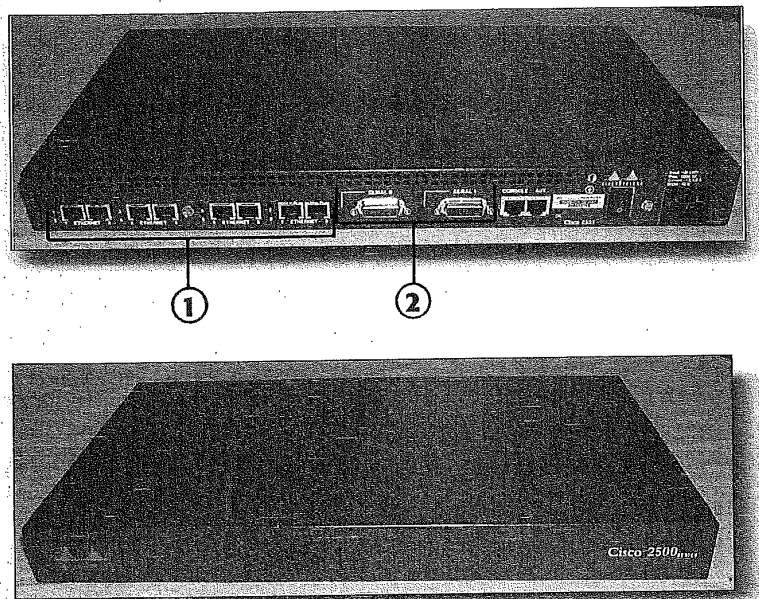
Becoming Familiar with Your Router

Routers provide the hardware and software necessary for routing. They are important internetworking devices for connecting LAN subnets and for making wide area connections between subnets. Chapter 5, "How a Router Works," provided the theory behind how a router works, and now we will take a look at the nuts and bolts of actually getting a router out of the box and ready for deployment on the network. Figure 7.1 shows the front and back of the Cisco 2505 router. The 2505 router provides only three interfaces, one LAN and two serial interfaces, and is typically used to connect subnets over serial connections such as ISDN, T1 leased lines, and other WAN alternatives.

FIGURE 7.1

The Cisco 2505 router is typically used to connect LANs over serial connections.

- ① Ethernet port/hub
- ② Serial ports



Several different Cisco Router models are available; each designed to satisfy a particular networking or set of networking needs. The number of ports and the type of ports on the different router models will vary, and rightly so because you will want to acquire a router

(or routers) with the appropriate connections to fill your internetworking requirements. (Many of the higher-end routers allow you to customize the type and number of interfaces found on the router.)

Cisco Router Design

Cisco routers must be able to build routing tables, execute commands, and route packets across network interfaces using routing protocols. This means that the router must have processing power, some sort of storage capacity, and available random access memory. Appropriate software such as an operating system that can be used to configure routed and routing protocols is also necessary (and is discussed in Chapter 9, "Working with the Cisco IOS").

Router CPUs

Routers aren't unlike PCs in that they contain a microprocessor. And just like PCs, different Cisco router models come with different processors. For example, the Cisco 2505 Router (which is the router that you will see in the various figures throughout this book) contains a 20MHz Motorola 68EC030 processor. A higher-end router like the Cisco 7010 Router contains a 25MHz Motorola MC68040 CPU. (Many of the lower-end routers use some of the same Motorola processors that are used in a variety of Apple Macintosh computers. Some of the very high-end routers use Risc processors that you would typically find on miniframe computers or very high-end servers.)

SEE ALSO

➤ For more information on specific Cisco routers, see page 337.

Router Memory Components

As already mentioned, routers not only need processing power, they also need a place to store configuration information, a place to boot the router operating system (IOS), and memory that can be used to hold dynamic information as the router does its job of moving packets on the internetwork. Cisco routers actually contain different types of memory components that provide the storage and dynamic

Getting the right router

Obviously, you will want to purchase the appropriate router or routers to fill your particular networking needs. The Cisco Web site at www.cisco.com provides a great deal of information on the various internetworking products that they sell. Also check out Appendix C, "Cisco Router Specifications List," which provides some descriptions and specifications for some of the Cisco routers available.

caching required. The following list provides information on the different memory components found in a Cisco router:

- **ROM**—Contains the Power-on Self-Test (POST) and the bootstrap program for the router. The ROM chips also contain either a subset or the complete router IOS (for example, the ROM on the 2505 router only contains a subset of the IOS, whereas the 7000 series contains the full IOS). Because the IOS is available on the ROM, you can recover from major disasters such as the wiping out of your Flash RAM. The ROM chips on Cisco routers are removable and can be upgraded or replaced.
- **NVRAM (nonvolatile RAM)**—Stores the startup configuration file for the router. NVRAM can be erased, and you can copy the running configuration on the router to NVRAM. The great thing about NVRAM is that it retains the information that it holds even if the router is powered down (which is extremely useful considering you won't want to have to reconfigure the router every time after the power goes down).
- **Flash RAM**—Flash is a special kind of ROM that you can actually erase and reprogram. Flash is used to store the Cisco IOS that runs on your router. You can also store alternative versions of the Cisco IOS on the Flash (such as an upgrade of your current IOS), which makes it very easy for you to upgrade the router. Flash RAM actually comes in the form of SIMMS (Single-Inline Memory Modules) and depending on the router you have, additional Flash RAM may be installed.
- **RAM**—Similar to the dynamic memory you use on your PC, RAM provides the temporary storage of information (packets are held in RAM when their addressing information is examined by the router) and holds information such as the current routing table. RAM also holds the currently running router configuration (changes that you make to the configuration are kept in RAM until you save them to NVRAM).

These various memory components all play an important role in what happens when you boot the router. The various possibilities revolving around the router system startup and where the router finds its IOS and start-up configuration files are discussed in the next chapter.

SEE ALSO

- » *The role that the different memory types play in the router boot up sequence are discussed in the next chapter, beginning on page 126.*

SEE ALSO

- » *The Cisco Router interfaces are another important hardware component of the router. They are discussed in Chapter 6, starting on page 99.*

Connecting the Console

With an overview of the internal components of the router and the router interfaces (in the previous chapter) taken care of, it's now time to walk through the steps of getting a new router out of its box and connecting it to the LANs that it will service (either by direct connection using a LAN port such as an Ethernet port or by connecting LANs using WAN connections). Configuring the router is discussed in Chapter 8, "Basic Router Configuration," with additional IOS configuration commands discussed in Chapters 9, 11, 12, 13, and 15.

Before you attempt to connect the router, it makes sense to take a look at the contents of the box that were shipped to you by Cisco or your Cisco reseller. Make sure you got what you paid for. Check the cable specifications (they are printed on the cable near the connectors), check the IOS that was shipped (the router won't work with the wrong IOS version), and make sure that the router contains the interfaces you ordered. If anything is missing or the router doesn't contain the correct interfaces (or interface cards used on the higher-end routers), get on the phone to Cisco (1-800-462-4726) or your local Cisco reseller.

After you have inventoried the router, cables, and software that you were shipped, you can start to put the router together. Connect the router's power cord to the router and a power source (make sure that the router is turned off); the next step is to connect a PC to the router to act as the router's console. The console can be pretty much any PC that has a serial port and can run some type of terminal emulation software. The PC, in effect, becomes a dumb terminal and provides you with the interface that you use to configure and monitor the router.

Getting the right IOS

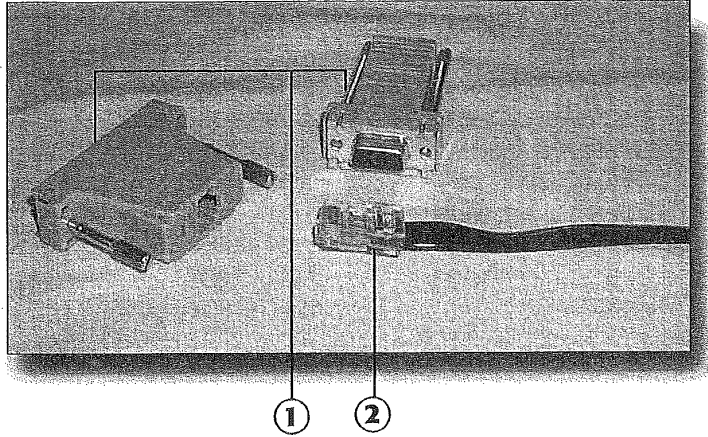
After you determine which router will work for a specific internetworking task, you also must decide which version of the Cisco IOS you will use. The Cisco site (www.cisco.com) also provides information on all the versions of the IOS available and provides a planner that helps you choose the appropriate IOS for your router (such as a 2505 router versus a 4500 router). The IOS that you select must also support the type of routing that you want to do. If you only want to route IP, you can choose a version of the IOS that only routes IP. If you must route IP, IPX, and AppleTalk, you must choose the correct version of the IOS. And be advised: The IOS is a separate purchase, so don't forget to order the appropriate IOS when you buy your router.

The console computer and the router are connected by the *roll-over cable* that ships with the router. The cable is terminated on both ends with an RJ-45 connector (see Figure 7.2).

FIGURE 7.2

The roll-over cable is used to connect the router to the PC console.

- ① Serial adapters
- ② Roll-over cable



Installing the router

You will want to position the router where it can be connected to the various LANs between which it will route information. This might mean that the router will be in a server closet or positioned where it can be connected to a leased line from your local telephone provider. Most Cisco routers come with mounting brackets that make it easy for you to install the router into hub racks and other server closet equipment racks. If the router will be placed in a very inaccessible spot, you can configure the router (discussed in Chapter 8) before you connect it to the various lines and LAN connections.

The router also comes with several different *serial adapters* that contain an RJ-45 port so that they can be connected to the roll-over cable and then to the serial port on the PC that you will use as the router's *console* (see Figure 7.2). After you've selected the appropriate serial adapter you are ready to connect the router and the console.

Connecting the router and the console

1. Place the RJ-45 male adapter on the roll-over cable in the port on the back of the router marked CONSOLE (see Figure 7.3).
2. Attach the serial adapter to the appropriate serial port on the PC that will serve as the console.

With the physical connection of the router to the PC taken care of, you now must set up some type of *terminal emulation software* on the PC. Terminal emulation software and the communication settings necessary to talk to the router are covered in the next section.

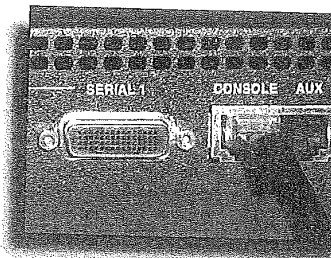


FIGURE 7.3

The roll-over cable is attached to the CONSOLE port on the router using the male RJ-45 connector.

Configuring the Router Console

The PC serving as the console communicates with the router using terminal emulation software. A number of these software packages exist, such as HyperTerminal (which ships as part of the Windows 95, 98, and Windows 2000 Professional operating systems) and ProComm Plus (a commercial communication program that offers faxing, terminal emulation, and other communication possibilities). A number of other possibilities are available on the Internet and can be downloaded as freeware or shareware (such as Tera Term Pro, an extremely easy-to-use and configure terminal emulator shown in Figure 7.4 and used throughout this book).

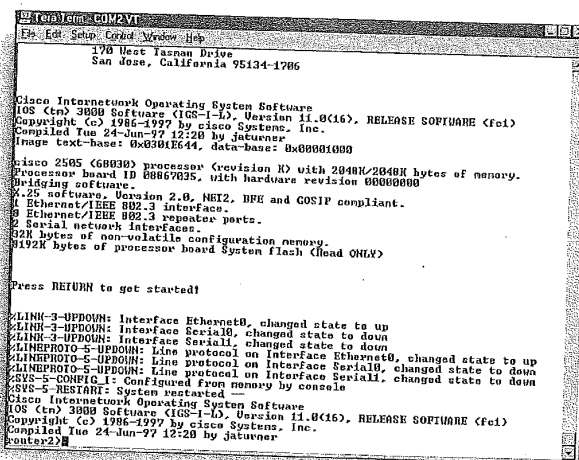


FIGURE 7.4

Terminal emulation software (such as Tera Term Pro) is used to communicate between the console and the router.

Make sure your terminal emulation software supports serial communication

Many terminal emulation software packages on the Internet are designed to telnet between computers connected to the Internet. This means that they don't support or allow you to configure the terminal software for communications via your serial ports. Before you spend a lot of time downloading and installing a particular package, make sure that it will allow serial connections. Windows HyperTerminal is available as part of your operating system and can be configured for serial communications (with the settings shown in Table 7.1).

After you have installed a particular terminal emulation software package, you must set up the communication parameters for the serial port that you will use to talk to the router. Table 7.1 shows the communication settings to be used by the software.

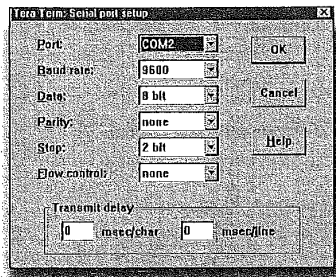
Table 7.1 Terminal Communication Settings

Parameter	Setting
Terminal Emulation	VT100
Baud rate	9600
Parity	none
Data bits	8
Stop bits	1 (2 stop bits for the 2500 series)

Working with the Terminal Emulation Software

Each terminal emulation package will operate a little differently, but each will provide some sort of menu/dialog box system that gives you access to the various settings for the software. Figure 7.5 shows the Serial port setup dialog box in Tera Term. Communication settings are configured using drop-down boxes.

FIGURE 7.5
Communications setting for the serial port will be available in a dialog box in most Windows-based terminal emulators.



After you've correctly configured the console's terminal emulator, it's really quite easy to establish communications with the router.

Establishing communications between the router and the console

1. Start your terminal emulator and make sure that you have selected the appropriate serial port for communications (and set the communication parameters shown in Table 7.1)
2. Power on the router (press the on/off switch on the router—it's on the back, left of the 2500 Series routers).

The banner for the router (as shown in Figure 7.4) should appear. If you seem to have a connection with the router, check your serial and console connections (on the roll-over cable) and make sure that you have specified the correct serial port for the communication session in the terminal emulator.

Routers right out of the box will not be configured. This means that none of the interfaces has been prepared for communications nor have the appropriate routed and routing protocols been set up on the new router. To configure a new router you'll need to follow the steps for router configuration found in Chapter 8.

SEE ALSO

» *Configuring a new router is discussed in the next chapter, starting on page 123.*

Connecting the Router to the Network

After the router is connected to the console you have a means to configure the various router parameters (other methods of configuring the router also are available, as outlined in the next chapter). The next step is connecting the router to the networks that it will service.

As discussed in Chapter 6, "Understanding Router Interfaces," several different interfaces can be available on your router (depending on the router model and the configuration that you chose for the router). For a basic walk through of some of the connection options, we will take a look at a 2505 Cisco Hub/Router.

LAN Connections

Depending on the type of router you have, LAN connections are typically made to an Ethernet or Token Ring interface port on the router and then to a hub or MAU (Multistation Access Unit, see

Serial communications trivia

Terminal emulation—your workstation is made to function as a dumb terminal that receives and sends information via its serial port. DEC (Digital Equipment Company) VT 100 was the standard mainframe and miniframe dumb terminal type and is used as a standard for many types of serial communications on the PC.

Baud rate—The speed of data transmission based on the signal elements sent per second (same as bps if each element is a bit).

Parity—An error-detection setting for serial communication; odd parity means that each data word must contain an odd number of bits; even parity means each data word transmitted must have an even number of bits. Any data words not following the parity setting (odd or even) must be retransmitted.

Data bits—The number of bits in each data packet that is sent and received.

Stop bits—The number of bits sent at the end of a data stream to signal the end of a particular packet.

Chapter 1 for more information) that supplies the connections for the various computers on the network. Let's assume that we are connecting an Ethernet LAN to our router. Typically a hub will be connected to the Ethernet port using CAT 5 twisted pair (the Ethernet interface provides an RJ-45 female port). The various computers on the network will then be connected to the hub.

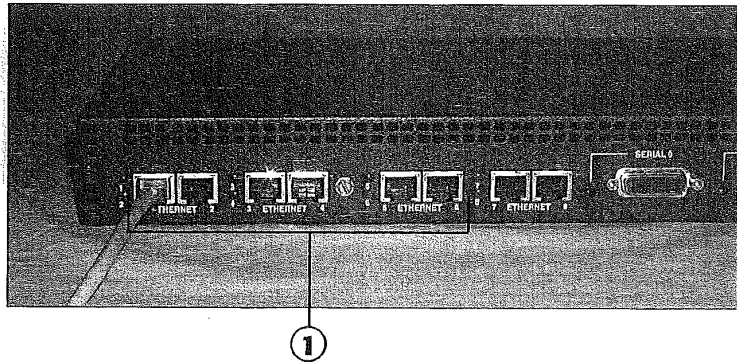
To use a straight-through CAT 5 twisted pair cable (the cable used for connecting PCs to hubs), you must switch the MDI/MDI-X switch on the router to the MDI-X position. For routers such as the Cisco 2505 and 2507 routers (which don't have the MDI/MDI-X switch), the router must be connected to a hub using a crossover cable (a cross-over cable is a modified straight-through twisted pair cable, where the pairs have been "reorganized" to reverse the transmit and receive electrical signals).

Some routers, such as the Cisco 2505 Router, actually provide the Ethernet interface in the form of a hub (see Figure 7.6). This negates the need for a separate hub, and PCs can be plugged directly into the hub ports available on the router. If more hub ports are required, a crossover cable can be used to connect one of the hub ports on the router to a port on an additional hub.

FIGURE 7.6

The Cisco 2505 provides one Ethernet interface in the form of an 8-port hub.

① Hub ports



SEE ALSO

► For more information on twisted pair cabling, see page 17.

Serial Connections

Serial connections on the router can be configured for several different WAN protocols. The actual physical serial connection on Cisco routers is a 60-pin female port (see Figure 7.7).

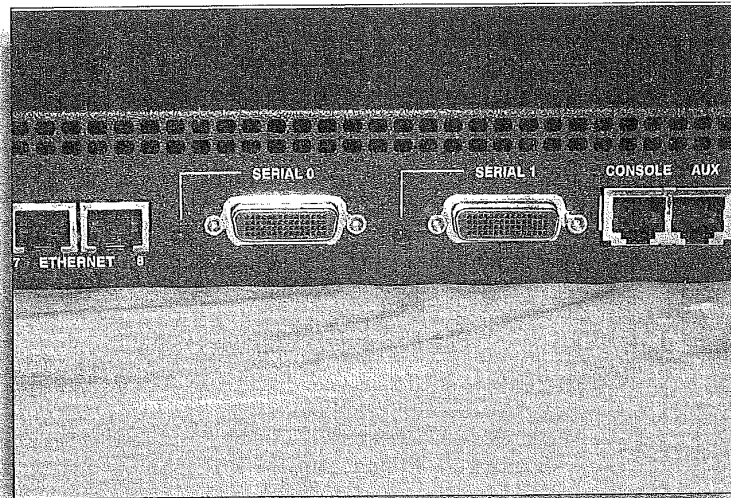


FIGURE 7.7

The Cisco router 60-pin serial port connector.

The Cisco 2505 Router (shown in Figure 7.6) supplies two serial ports. The serial port supports several different signaling standards including V.35, X.21bis, and EIA-530. Figure 7.8 shows a V.35 cable that supplies the male 60-pin connector for connection to the router's serial port. The other end of the V.35 cable would typically be placed in a CSU/DSU or other device in WAN connections. Table 7.2 lists some of the signaling standards supported by Cisco serial interfaces.

Table 7.2 Serial Signaling Standards

Standard	Specification
V.35	Synchronous communications between networks and packet-switching WANS
X.21bis	Defines communications between DTEs and DCEs in an X.25 WAN
EIA-530	RS232 standard for unbalanced serial communications

Daisy-chained hubs

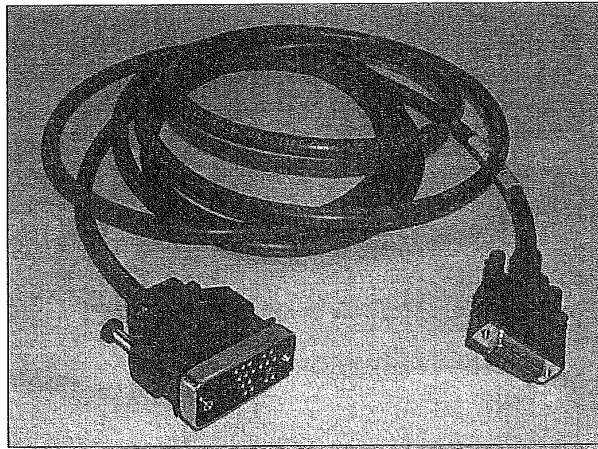
If you plan on daisy-chaining (connecting hub-to-hub) several hubs to an Ethernet port on a router, remember that you are limited to four hub devices in the data path between Ethernet devices.

Check your connections

If you've physically connected a particular interface correctly, you will typically find that the router acknowledges the connection. For example, connecting a serial connection from your router to the appropriate device will usually register on the router as the fact that the particular interface is up, meaning it is active (even if an appropriate protocol has not yet been configured for the interface).

FIGURE 7.8

A V.35 cable provides the connection between the router's serial port and another device such as a CSU/DSU.



Buyer beware!

When you acquire the cables that you need to connect your router interfaces to various serial connections, make sure that you purchase the appropriate pin configurations. All cables look alike to the vendors you call for your equipment; be very specific about your cable specifications.

A Final Word on Physical Router Connections

Whether you should configure the router before connecting it to the serial and LAN interfaces that it will service, or connect the router and then configure it, is pretty much a chicken-or-egg dilemma. Configuring the router with a very basic configuration so that it can be seen on the network can allow you to then connect the router to all its various physical connections and then complete the configuration of the router using a virtual terminal over the network (virtual terminals are discussed in the next chapter).

If the router can be connected to the various LAN and WAN devices before you configure the router, this allows you to fully configure and test the connections immediately. However, if the router is placed in an area that is somewhat difficult to access (such as a small closet on a hub rack), it might be difficult to directly connect a PC to the router for configuration purposes.

Whatever the case, the next chapter discusses how to configure a new router right out of the box.

chapter

8

Basic Router Configuration

Configuring a Router



Router Boot Sequence



Working with the System
Configuration Dialog Box



Using the Different Router Modes



Configuring a Router

Setting up a basic configuration for a router is a matter of enabling the various interfaces on the router and setting the software settings for the routed and routing protocols. For example, if you are routing IP, the interfaces must be assigned appropriate IP addresses. Routing protocols must also be configured (if you are going to use RIP or IGRP, you must configure these protocols). And any serial interfaces that you use must also be configured with an appropriate WAN layer 2 protocol (such as HDLC or Frame Relay). Basic configuration information may also include bandwidth information and timing information for WAN connections.

Bottom line—the configuration file for your router uses software settings that tell the router what to route and how to route it. All the commands that you use to configure the router are part of the Cisco IOS command set. You will also find that there are several different ways that you can configure the router, either directly by using the router console, or by loading a configuration file that has been placed on a Trivial File Transport Protocol (TFTP) server on your network. The following list shows some of the possibilities for loading configuration information onto a router:

- **Router Console**—You can configure the router directly from a PC—the *router console*—that is connected to the router console port using the rollover cable that comes with the router. The PC must be running terminal emulation software that allows you to connect to the router through the PC's serial port. You also can connect directly to the router using the router's auxiliary port, which is typically housed next to the console port on the back of the router.
- **Virtual Terminal**—If the router has already been provided a basic configuration that gets at least some of the interfaces up and running on the network (such as an Ethernet port), you can Telnet to the router via a *virtual terminal*. This simply means that a computer on the network that is running a Telnet program can connect to the router and configure the router (if the appropriate passwords are known—which will be discussed in more detail later in this chapter).

- **Network Management Workstation**—Routers can also be configured from a workstation on the network that runs special network management software, such as Cisco's CiscoWorks or a similar product from Hewlett Packard known as HP OpenView.
- **Cisco ConfigMaker**—This graphics-based program (see Figure 8.1) allows you to build a configuration for a router or routers on a network and then load the configuration to a router that is directly connected to a router console (the PC that is running ConfigMaker) or other routers that are connected to the network. Delivering router configurations from ConfigMaker to routers on the network requires that the network interfaces on these routers already be configured. ConfigMaker will be discussed in greater detail in Chapter 16, "Configuring the Router with Cisco ConfigMaker."
- **TFTP Server**—A configuration for a router can be loaded from a TFTP server on the network. Saving configurations to a TFTP server and then downloading them to a particular router is very straightforward. TFTP servers will be discussed in Chapter 17, "Using a TFTP Server for Router Configuration Storage."

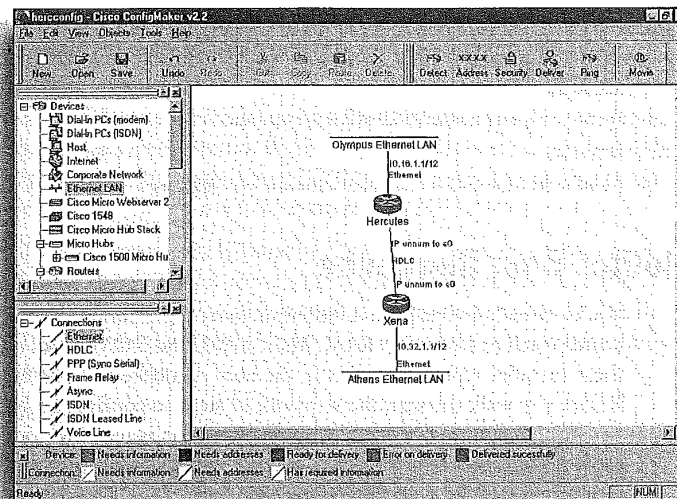


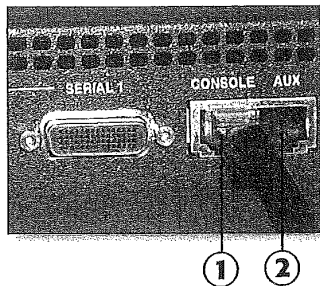
FIGURE 8.1
Software such as Cisco's ConfigMaker allows you to diagram your internetwork and then load your configurations to a router or routers.

Of all the configuration methods available, probably the easiest and the most directly hands-on is configuring the router by directly connecting a PC to the router console port (see Figure 8.2). This not only allows you to quickly set up a basic configuration on the router using the router System Configuration dialog, but it also allows you to fine-tune your configuration in the router Configuration mode. Both of these configuration methods will be discussed in the chapter.

FIGURE 8.2

A PC can be directly connected to a router using the console or auxiliary ports.

- ① Console port
- ② Auxiliary port

**As good as gold**

Configuring a router correctly and appropriately for the internetwork it serves is really the most important aspect of working with routers (of course, I'm downplaying internetwork design and troubleshooting for the moment). This is why Cisco Certified Internetworking Engineers are highly paid and respected internetworking professionals. A proper configuration really becomes as important as gold. You will look at different ways of saving (and protecting) your configuration files as you work through this chapter.

Before you take a look at how to set up a basic configuration using the System Configuration dialog on a new router, let's take a look at the router boot sequence. This will also give us some insight into where the router looks for a configuration file when it comes online.

SEE ALSO

» For more information about TFTP servers, see page 289.

SEE ALSO

» For more information about basic router commands and configuring a router, see page 141.

Router Boot Sequence

You've already learned the different memory types found in the router (such as RAM, NVRAM, Flash RAM, and ROM). And all these memory types play a part in the boot sequence of a router. Before you walk through the sequence of steps to configure a brand new router right out of the box, some discussion is required to explain the router boot sequence and the various places that the router will look for a configuration file.

When you power the router on, the ROM chip runs a *Power On Self Test (POST)* that checks the router's hardware such as the processor, interfaces, and memory. This test isn't unlike the power-on test that a PC runs when you power it on (RAM, CPU, and other hardware is checked).

The next step in the router boot-up sequence is the execution of a bootstrap program that is stored in the router's ROM. This bootstrap program searches for the CISCO IOS. The IOS can be loaded from the ROM itself (routers either have a partial or complete copy of the CISCO IOS in ROM), the router's FLASH RAM, or from a TFTP server on the network (commands for loading the IOS from various locations will be discussed in the next chapter). The IOS is typically stored in the router's Flash RAM.

After the router's IOS is loaded, the router searches for the configuration file. The configuration file is normally held in NVRAM (a copy command is used to copy a running configuration to NVRAM). As with the IOS, however, the configuration file can be loaded from a TFTP server (again, the location of the configuration file would be dictated by information held in the router's NVRAM).

After the router loads the configuration file, the information in the file enables the interfaces and provides parameters related to routed and routing protocols in force on the router. Figure 8.3 provides a summary of the router start-up process. Keep in mind that loading the IOS from a source other than Flash RAM requires a notation in the ROM's configuration Registry and that to load the configuration file from a source other than NVRAM, information pointing to the location of the file has to be contained in NVRAM.

If a configuration isn't found in NVRAM or in another place specified (such as a TFTP server), the Setup mode is entered and the System Configuration dialog appears on the router console screen. The next section discusses how to set up a basic router configuration using the dialog.

SEE ALSO

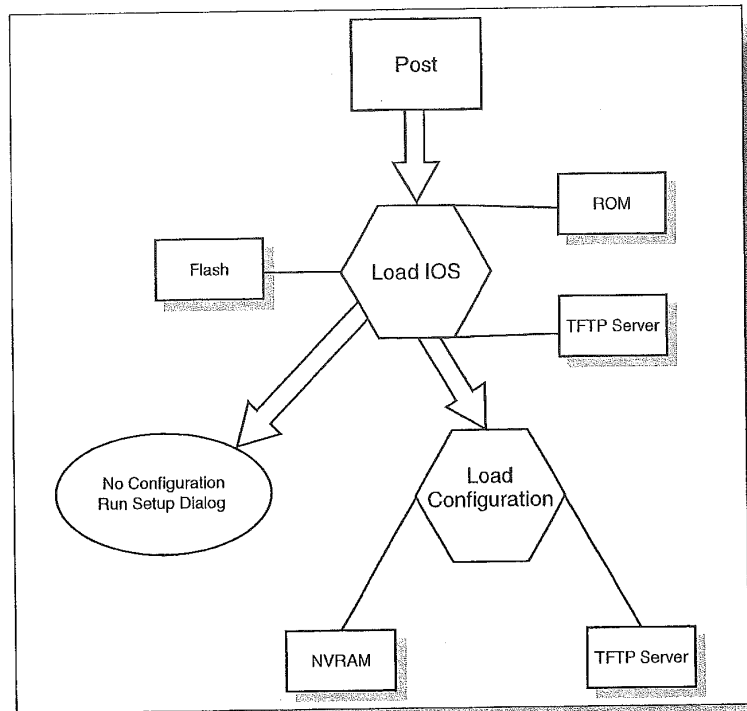
- To review the different memory components on a router, see page 113.

SEE ALSO

- For more about the Cisco IOS command set, see page 142.

FIGURE 8.3

The router boot sequence loads the router IOS and the router configuration file.



Configuring a router from scratch

You can erase the configuration file for a router and then start over, building a new basic configuration using the configuration dialog. At the enable prompt type `erase startup-config`, and then press **Enter**. This erases the configuration file from NVRAM. To restart the router type `reload`. Then press **Enter** to confirm the reload. The router will reboot and the System Configuration dialog will appear on the Router Console screen.

Working with the System Configuration Dialog Box

When you boot up a new router (or a router where the configuration file has been deleted), the System Configuration dialog is loaded (see Figure 8.4). This Setup mode asks you a series of questions; the answers to those questions provide a basic configuration for the router.

Working through the Setup dialog is very straightforward. You do need to know certain parameters related to the configuration of the router, however, such as which network protocols you will route (IP, IPX, AppleTalk) and the parameters related to the various interfaces. For example, if you route IP you will need to know the IP addresses of the router interfaces that you want to configure (the following steps provide sample addresses). If you have a router that you want to configure, follow the steps provided.


```

Cisco Internetwork Operating System Software
IOS (tm) 2500 Software (C2500-B-L), Version 11.3(3), RELEASE SOFTWARE (fc1)
Copyright (c) 1986-1998 by Cisco Systems, Inc.
Compiled Mon 24-Apr-98 18:46 by plannyp
Image text-base: 0x03031F7C, data-base: 0x00001000

Cisco 2505 (68030) processor (revision X) with 2048K/2048K bytes of memory.
Processor board ID 08867026, with hardware revision 00000000
Bringing software.
M.25 software, Version 3.0.0.
1 Ethernet/IEEE 802.3 interface(s)
1 Ethernet/IEEE 802.3 receiver port(s)
2 Serial network interface(s)
32K bytes of non-volatile configuration memory.
512K bytes of processor board System Flash (Read ONLY)

Notice: NVRAM invalid, possibly due to write erase.
      System Configuration Dialog

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Would you like to enter the initial configuration dialog? [yes]:

```

FIGURE 8.4

The Setup dialog helps you build a basic configuration for a new router by asking a series of questions.

SEE ALSO

➤ For more about IP addressing, see page 195.

Starting the Setup Dialog Box

The Setup dialog can ask you quite a few questions related to setting various passwords for the routers and configuring the interfaces on the router. The first part of the setup configuration relates to setting up enable and virtual terminal passwords for the router.

Starting the configuration process with the Setup dialog

1. You will be asked *Would you like to enter the initial configuration dialog?* (see Figure 8.4). Press **Enter** to answer yes (the default option) and continue.
2. You will then be asked if you want to see the current interface summary. This allows you to view the interfaces on the router. Press **Enter** to continue. A summary of the interfaces on the router will be provided as shown in Figure 8.5. Note that the Ethernet 0 interface is up, but that both the serial interfaces on this router are down. Also, no IP numbers have been assigned to the interfaces.
3. Next, you are asked to provide a name for the router. Type a name (such as *ciscokid*) and then press **Enter**.

IOS version and supported network protocols

The 2505 router configured in the figures in the following sections is running Cisco IOS 11.3. This version of the IOS supports IP, IPX, AppleTalk, and DECnet routing. This book will discuss the routing of IP, IPX, and AppleTalk, the most commonly routed network protocols.

FIGURE 8.5

The Setup dialog provides a summary of the physical interfaces on the router.

```

At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Would you like to enter the initial configuration dialog? [yes]:

First, would you like to see the current interface summary? [yes]:

Any interface listed with OK? value "NO" does not have a valid configuration

```

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0	unassigned	NO	unset	up	up
Serial0	unassigned	NO	unset	down	down
Serial1	unassigned	NO	unset	down	down

```

Configuring global parameters:
Enter host name [Router]:

```

4. The next Setup dialog question asks you to provide an enable secret password. This password is encrypted and will provide you with access to the router's Enable mode (the mode that allows you to make changes to the router's configuration). Type an appropriate password, and then press **Enter**.
5. You are then asked to provide an "enable" password, which seems redundant because you have already provided a secret password for the Enable mode. This second password is related to earlier versions of the Cisco IOS that didn't provide the capability to create an encrypted password for the Enable mode. Because you aren't allowed to leave this password blank (even though you won't use it), type a value (something you can remember but isn't apparent to someone trying to access the router who shouldn't). In this case I will use password. Press **Enter** to continue.
6. You will then be asked to provide a virtual terminal password for the router. This password is used by virtual terminals that Telnet to the router over the network. This enables you to monitor (and even configure a router) from a remote workstation on the network. Provide a virtual terminal password, and press **Enter** to continue.
7. The next Setup dialog question asks you if you want to enable *SNMP (Simple Network Management Protocol)*. This protocol provides baselines for network operations and provides a way to monitor changes in the network using a management station (which requires software such as CiscoWorks). If you won't use management software to manage the routers, there is no reason to enable SNMP). In this case you won't enable it. Type no at the prompt and press **Enter** to continue.

Configuring Routed Protocols

The next portion of the Setup dialog is related to the configuration of routed and routing protocols that will be used on the router. You will be asked if you want to enable each of the routed protocols supported by your version of the IOS and to choose which routing protocols you want to enable.

Configuring protocols with the Setup dialog

1. In the case of the 2505 router that you are configuring, the next prompt asks if DECnet should be enabled (*DECnet* is a protocol stack supported by the Digital Equipment Corporation). The default response is **No**. Press **Enter** to continue.
2. In the case of our 2505 router, the next dialog prompt asks if AppleTalk should be configured. For now, you will respond with **no** (the default). Chapter 13, "Routing AppleTalk," covers the ins and outs of AppleTalk routing and I'll defer AppleTalk until then. Press **Enter** to continue.
3. The next dialog prompt asks if IPX should be configured (IPX is covered in detail in Chapter 12, "Routing Novell IPX,"). To answer **no**, press **Enter**.
4. The next prompt asks if IP should be configured and the default answer is **Yes** (see Figure 8.6). Although IP will be covered in great detail in Chapters 10, "TCP/IP Primer," and 11, "Configuring IP Routing," it makes sense to enable IP at this point. This enables you to get the router up and running on the network, and then you can further configure the router using a virtual terminal or by loading a ready-made configuration file from Cisco ConfigMaker or a TFTP server. Press **Enter** (to say **yes**) and continue.
5. You will then be asked if you want to configure IGRP on the router. IGRP is one of the IP routing protocols. Configuring IGRP and RIP will be covered in Chapter 11, so for the moment you can say **no**. Type **no** and press **Enter** to continue.
6. You will then be asked to configure RIP. **No** is the default, so press **Enter** to continue.
7. The next dialog asks if bridging should be enabled on the router. Press **Enter** to continue (**No** is the default).

FIGURE 8.6

Enabling IP allows you to get the router up and running on the network for further configuration.

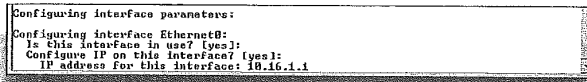
```
Configuring global parameters:
Enter host name [Router]: ciscokid
The enable secret is a one-way cryptographic secret used
instead of the enable password when it exists.
Enter enable secret: password
The enable password is used when there is no enable secret
and when using older software and some boot images.
Enter enable password:
No defaulting allowed
Enter enable password: cisco
Enter virtual terminal password: password
Configure SNMP Network Management? [yes]: no
Configure DECnet? [no]:
Configure AppleTalk? [no]:
Configure IP? [yes]:
```

Configuring Router Interfaces

The next part of the Setup dialog is related to the configuration of the router's interfaces. You will be asked which router interfaces will be in use on the router (such as Ethernet and serial interfaces). Also, because IP was enabled for routing, you will have to supply IP addresses for the various interfaces on the router. How these IP addresses were arrived at will be discussed in Chapter 10.

Configuring interfaces with the Setup dialog

1. The next prompt relates to the first interface on the router, which in the case of the 2505 router is the Ethernet 0 interface. You will be asked if this interface is in use. Yes is the default value, so to enable the interface, press **Enter**.
2. The next prompt asks if IP should be configured on the interface (E0). The default value is Yes; press **Enter** to continue.
3. The next prompt asks for the IP address of the interface (interfaces on the router use IP addresses just like any other node on the network). Type 10.16.1.1 as the address for the E0 interface (see Figure 8.7). Then press **Enter** to continue.
4. The next prompt asks how many bits are in the subnet field. This number relates to how many IP subnets have been created for your internetwork. This will be discussed in Chapter 10. For now, trust that I've divided the available network addresses (which are class A addresses) into 14 subnets, which requires 4 bits in the subnet field (this will make sense after you read Chapter 10). Type 4 and then press **Enter**.



```
Configuring interface parameters:
Configuring interface Ethernet0:
Is this interface in use? (yes):
Configure IP on this interface? (yes):
IP address for this interface: 10.16.1.1
```

FIGURE 8.7

An IP address is assigned to the Ethernet 0 port on the router.

5. Because the 2505 router's E0 interface is actually an eight-port hub, you are asked if you want to enable all ports on the hub. The default is Yes (and you want to say yes), so press **Enter** to continue.
6. You are then asked if you want to configure the next interface on the router, which in this case is serial 0. Yes is the default. Press **Enter** to continue.
7. You are then asked if you want to configure IP on the S0 interface. Press **Enter** and continue.
8. You are given the option of configuring the S0 interface as IP unnumbered (this means that the interface will route IP but doesn't require its own IP number). This is done to actually save your IP addresses (from the pool of IP addresses that you have available). Configuring serial interfaces with IP addresses will be handled in more detail in Chapter 11. For now, press **Enter** to say no.
9. You are then asked to provide an IP address for the S0 interface. Type 10.32.1.1. Then press **Enter**.
10. You will then be asked to provide the subnet field bits. This is defaulted to 4, which was entered in step 4. Press **Enter** to use the same bit count.
11. You are now asked to configure the Serial 1 interface. Press **Enter** to say yes.
12. Press **Enter** to say no to IP unnumbered.
13. Type the IP address 10.48.1.1 at the prompt (see Figure 8.8). Then press **Enter**.

FIGURE 8.8
IP addresses given to
each of the serial inter-
faces on the router.

```
Configuring interface parameters:
Configuring interface Ethernet0:
Is this interface in use? [yes]:
Configure IP on this interface? [yes]:
IP address for this interface: 10.16.1.1
Number of bits in subnet field [0]: 4
Class A network is 10.0.0.0, 4 subnet bits; mask is /12
Enable all hub ports on this interface? [yes]:

Configuring interface Serial0:
Is this interface in use? [yes]:
Configure IP on this interface? [yes]:
Configure IP unnumbered on this interface? [no]:
IP address for this interface: 10.32.1.1
Number of bits in subnet field [4]:
Class A network is 10.0.0.0, 4 subnet bits; mask is /12

Configuring interface Serial1:
Is this interface in use? [yes]:
Configure IP on this interface? [yes]:
Configure IP unnumbered on this interface? [no]:
IP address for this interface: 10.48.1.1
```

The next prompt is where you enter the subnet bits (4 is supplied as the default number of subnet bits). Then press **Enter**.

After you press **Enter**, the screen will scroll rapidly, showing link tests for the interfaces that you have configured. You will be asked if you want to use the current configuration. Type **yes** and then press **Enter** to save the configuration file that you created using the System dialog. The router will build the configuration and save it to NVRAM.

The next time you press **Enter**, the router will take you to the router's User mode prompt. You are now ready to view the configuration parameters on the router or edit the configuration of the router.

Using the Different Router Modes

After the router contains a basic configuration, you can begin to examine the different router modes available. The router supplies you with three basic levels of access: *User mode*, *Privileged mode*, and *Configuration mode*.

Each of the basic router modes provides a higher degree of access to the router's configuration and also gives you greater capabilities to edit the configuration of the router. The list that follows briefly describes the three router modes:

- **User mode**—This mode provides limited access to the router. You are provided with a set of nondestructive commands that allow examination of certain router configuration parameters. You cannot, however, make any changes to the router configuration.

Other router modes

Other router modes exist that enable you to configure a router that cannot find a valid IOS image in Flash RAM or in cases where you want the router to load the IOS from a source other than Flash RAM. The ROM Monitor mode is entered when the router doesn't find a valid IOS image. You can configure the router from the ROM Monitor prompt. The RXBoot mode is used to actually help the router boot when it doesn't find a valid IOS image. An important use of the ROM Monitor is changing forgotten passwords. See "Getting Around Lost Passwords," later in this chapter, for information about getting around lost passwords.

- Privileged mode—Also known as the Enabled mode, this mode allows greater examination of the router and provides a more robust command set than the User mode. After you enter the Privileged mode using the secret or enable password (if a secret encrypted password was not set), you have access to the configuration commands supplied in the Configuration mode, meaning you can edit the configuration for the router.
- Configuration mode—Also called the Global Configuration mode, this mode is entered from the Privileged mode and supplies the complete command set for configuring the router. Subsets of the Configuration mode exist for protocols, interfaces, and other aspects of the router operation.

User (Unprivileged) Mode

As I've already noted, the User mode enables you to do a limited survey of the router's configuration. The User mode is the default mode when you reboot a router. Even access to the User mode can be protected by a console password (see the "Configuration Mode" section that follows for information on the various password commands).

Figure 8.9 shows the user prompt on the router you configured using the System dialog. The prompt is the router's name followed by > (the greater than sign). This figure also shows a portion of the results from the show interfaces command.

```

Cisco642>show interfaces
Ethernet0 is up, line protocol is up, using hub 0
Hardware is Lance, address is 0010.7b3a.50c3 (bia 0010.7b3a.50c3)
Internet address is 10.16.1.1/22
MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
Encapsulation ARPA, loopback not set, keepalive set (10 sec)
ARP type: ARPA, ARP Timeout 04:00:00
Last input never, output 00:00:02, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runs, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 input packets with dribble condition detected
445 packets output, 47161 bytes, 0 underruns
0 output errors, 0 collisions, 2 interface resets
0 babble, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
--More--

```

FIGURE 8.9

The User mode allows you to view router configuration information using a limited command set.

The User mode is pretty much a “you can look but don’t touch” environment. It can, however, provide a wealth of information about the router and its current status. More about the commands available in the User mode are discussed in Chapter 9, “Working with the Cisco IOS.”

SEE ALSO

- For more information about router examination commands available in the User mode, see page 141.

Privileged Mode

The Privileged mode provides all the commands found in the User mode, but also includes an extended set of commands for examination of the router status (such as the `show running-config` command for examining the current running configuration on the router). The Privileged mode also supplies the `config` command, which enables you to enter the Configuration mode for the router.

The Privileged mode really controls the router. So, it’s important that the enable password be considered a thing of great value. You don’t want just anyone messing with the router’s configuration (if you just want to let someone take a look at some of the router parameters, they can use the User mode).

To enter the Privileged mode on a router, type `enable` at the User mode prompt and then press **Enter**. Provide the enable password (which will be the secret encrypted password that you set for the router) and press **Enter**. Figure 8.10 shows the router in the Privileged mode after the `show running-config` command has been invoked. The Privileged prompt is the router’s name followed by the # (number) symbol.

When you have finished working in the Privileged mode, it makes sense to return to the User mode. Otherwise, you leave the router wide open to be configured by anyone who happens by the terminal. To return to the User mode, type `disable` and press **Enter**. If you want to totally log off the router, type `logout` and press **Enter**. This means that the next person to use the console will have to enter the router password (if one exists) to enter the User mode.

3. To change the name that has been given to the router, type `host-
name [name]`, where *name* is the name you want to give to the
router. After entering the command, press **Enter**. The new
router name will appear at the Config prompt (see Figure 8.11).

FIGURE 8.11

In the Configuration mode you can change the router name and set the enable and login passwords.

```

Term - COM2VT
File Edit Setup Config Window Help
Popeye#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CTRL/Z.
Popeye(config)#hostname Winpy
Winpy(config)#enable secret hamburger
Winpy(config)#line console 0
Winpy(config-line)#login
Winpy(config-line)#password cisco
Winpy(config-line)#
  
```

4. To set the enable password, type `enable secret [password]` at the Config prompt, where *password* is the word that you will use as your secret password to get into the Privileged mode. Then press **Enter**. I've set my password as hamburger (see Figure 8.11).
5. Now you can set a password for the router. This means that anyone logging in to the router will have to provide this password to even access the User mode. To set the password you must get into the Line Console mode. Type `line console 0`, and then press **Enter**.
6. You are now in the line console Configuration mode; type `login`, and then press **Enter**.
7. To specify the login password, type `password [password]`, where the *password* is the word you want to use to log in to the router. For example, I've made my login password cisco (see Figure 8.11).
8. When you have completed your configuration changes, press **Ctrl+Z**. This will save your changes to the router's running configuration and return you to the Privileged prompt.

After you have made changes to the running-config, you may want to save them to your startup-configuration in NVRAM. This is the file that is loaded when the router is rebooted or restarted. At the Privileged prompt, type `copy running-config startup-config`, and then press **Enter**. The new startup-configuration will be displayed on the console screen.

Getting Around Lost Passwords

Sometimes you just forget those passwords, which can be bad news if you need to enter the Privileged mode and change a router's configuration.

Replacing a lost password

1. Turn the router off and after waiting five seconds turn it back on. As the router reboots, press **Ctrl+Break**.
2. You will enter the ROM Monitor mode. Type `e/s2000002`, and then press **Enter**. Write down the virtual configuration number that appears.
3. Now at the prompt type `o/r0x2142` and press **Enter** (this makes the router ignore the configuration file in NVRAM). Type `i` at the prompt, and press **Enter**. The router will reboot and enter the configuration dialog. Click **No** at the dialog prompt, and then press **Enter**.
4. At the router prompt type `enable` to enter the Privileged mode. Type `copy startup-config running-config`, and then press **Enter** to get your original configuration into the router's RAM.
5. At the enabled prompt type `config`. You are now in the Configuration mode. Type `enable secret new password`, where *new password* is your new secret password. Now you must set the register contents back to the original contents.
6. At the config prompt, type the `config-register 0x` virtual configuration number (which is the virtual configuration number that you wrote down). Press **Enter**.
7. Now Type `end` and press **Enter** to get out of the Configuration mode. Reboot the router. Now you should have a new secret password, and the router should be back to its normal configuration.

Virtual terminal password

One password that you didn't work with is the virtual terminal password. This is the password that is used by anyone who wants to Telnet into your router. Telnet is discussed in Chapter 11 in the "Using Telnet" section. To change your virtual terminal password, at the Config prompt type `line vty 0 4`, and then press **Enter**. This puts you in the virtual terminal Configuration mode. Type `login` and press **Enter**. Then type `password [password]`, where *[password]* is the password you want to use as the Telnet password. Press **Ctrl+Z** to end the configuration session.

PART II Router Design and Basic Configuration

CHAPTER 8 Basic Router Configuration

Becoming familiar with the various modes of the router and the commands that they offer is an extremely important aspect of overall router management. In the next chapter you will become more familiar with the Cisco IOS and the commands and command structure that it offers. Each of the modes discussed in this chapter will be covered in the context of the IOS commands available in a particular mode.

chapter 9

Working with the Cisco IOS

Introducing the Internetworking
Operating System

Command Structure

The IOS Help System

Router Examination Commands

Using the Privileged Mode

Checking Router Memory

Checking Out the Internetwork
Neighborhood

Viewing CDP Neighbors

Creating a Router Banner



Introducing the Internetworking Operating System

The Cisco *Internetworking Operating System (IOS)* is the software that provides the router hardware with the capability to route packets on an internetwork. The IOS, like any operating system, provides the command sets and software functionality that you use to monitor and configure the router, and it also provides the functionality for the various protocols—both routed and routing—that make internetworking a reality.

Configuring the router means that you enable the various interfaces and protocols on the router. You must use commands that bring your various hardware interfaces such as Ethernet or serial interfaces to life. You must also provide configuration information for the protocols that are routed, such as IP or IPX/SPX. And you must also configure routing protocols such as RIP and IGRP. After the router is configured, you must manage your configuration files. The list that follows details some of the things you would do with the IOS command set:

- **Configure the router LAN interfaces**—Configuring the router LAN interfaces should be done after you have made the physical connections, assembling the router hardware and connecting the various cables to LAN or WAN networks. The router interfaces must be configured for use on these networks. For example, on a network that routes IP, each Ethernet interface involved must be configured with an appropriate IP address and subnet mask.
- **Configure Serial Connections and WAN protocols**—In cases where your router is connected to a WAN by a leased line or some other WAN technology, you must configure the WAN protocol used on the serial interfaces of the router.
- **Manage router configuration files**—After the router is configured, you will want to maintain copies of the configuration file. You will save the running configuration to NV RAM where it is stored as the startup configuration. You may also want to save a configuration file or load a configuration file from a TFTP server (this is covered in Chapter 17, “Using a TFTP Server for Router Configuration Storage”).

- Monitor and maintain the router—You will also use the IOS command set to monitor and troubleshoot problems with the router. A time may also come when you need to update the router IOS in Flash RAM. The command set provides all the tools necessary to keep an eye on the router and update its IOS and feature set if required.

Although this list may seem exhausting in terms of what you must do to maintain routing on your network, it is by no means exhaustive. The Cisco IOS command set is huge and the subject of a number of books. Cisco publishes a software command summary for each of the IOS versions, and these books are as thick as the New York City telephone directory. The command reference for IOS 11.3 is in excess of 1,000 pages. You will find, however, that you will use only a fairly small percentage of all the IOS commands available, even if you become a routing maniac and have an opportunity to work with some of the higher end routers on a large internetwork.

Cisco provides a *Command-Line Interface (CLI)* that you can use to configure and maintain your router. You can access the CLI using a router console or by Telnetting to a router using a virtual terminal.

This chapter will provide an overview of the IOS and the CLI and let you get your feet wet with a very complex and robust operating system. Commands related to configuring IP, IPX, AppleTalk, and router serial interfaces (and WAN protocols) are discussed in subsequent chapters.

If you are a DOS or UNIX aficionado, you will find the CLI familiar. It is a very typical command-line interface. If you aren't familiar with command-line interfaces, figures are provided to keep you on track with the commands discussed. You will find that the command structure is fairly straightforward.

SEE ALSO

- For more information about the routing protocols such as RIP and IGRP, see page 131.
- For a summary of the IOS commands discussed in this book, see page 324.

Command Structure

You already worked with the IOS command set briefly in Chapter 8, “Basic Router Configuration,” when you explored the different router modes: User, Privileged, and Configuration. Each of these modes provides a different set of commands:

- The User mode provides only basic commands that enable you to view system information and perform basic tests.
- The Privileged mode provides a larger set of commands for viewing router information and also provides access to the Configuration mode.
- The Configuration mode provides the command set that enables you to configure the interfaces and protocols used on the router.

SEE ALSO

➤ For more information about the different router modes, see page 134.

Exec Commands

The Cisco IOS uses a command interpreter to execute your commands (it interprets the command and then executes it) called the *Exec*, and the User mode and the Privileged mode are considered different levels of the Exec. So, when you are in the User mode or the Privileged mode, the commands available take on a particular basic structure: the command followed by the router parameter. The command will be one of the IOS commands, such as `show`, and the router parameter is the item on which you want the command to act.

So, for example, the command `show Ethernet 0` will display the parameters related to the first Ethernet interface on the router. Figure 9.1 shows this command and its results on a 2505 router running IOS 11.2.

To actually execute the commands that you use in the various router modes, always press **Enter** after typing the command. The results of the command are then displayed on the router console or virtual terminal screen.

Configuration mode has its own command structure

Although the Configuration mode is kind of an extension of the Privileged mode (you have to be in Privileged mode to get to the Configuration mode), you will find that the configuration commands have a slightly different structure than the Exec commands used in User and Privileged modes. Router configuration commands will be discussed in a number of different chapters in the context of particular configurations such as different LAN protocols in Chapters 11, 12, and 13 and WAN protocols in Chapter 15.


```

Tera Term - COM2.V1
File Edit Setup Control Window Help
show interface ethernet 0
Ethernet0 is up, line protocol is up, using hub 0
Hardware is Lance, address is 0010.7b3a.50b3 (bia 0010.7b3a.50b3)
Internet address is 138.10.64.1/19
MTU 1500 bytes, BW 100000 Kbit, DLY 10000 usec, rely 255/255, load 1/255
Encapsulation ARPA, loopback not set, keepalive set (10 sec)
ARP type: ARPA, ARP Timeout 04:00:00
Last input never, output 00:00:04, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 runs, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 input packets with dribble condition detected
6 packets output, 840 bytes, 0 underruns
0 output errors, 0 collisions, 3 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
nopeye0

```

FIGURE 9.1

Exec commands typically take on the form of a command followed by what you want them to act on, such as show Ethernet 0.

Configuration Mode

The Configuration mode handles its commands in a slightly different way. Whereas the Exec commands are two-part requests where you specify the command and what you want it to act on, the *configuration commands* require that you execute several commands that, layered together, actually change the parameters related to a particular interface or a particular protocol. For example, let's say that you want to select the WAN protocol that you will use on a particular serial interface on the router (in this example, serial 1). Remember that you have to be in the Privileged mode to enter the Configuration mode.

Configuring a WAN protocol for a serial interface

1. At the Privileged prompt type `config`, and then press **Enter**.
2. You will be asked if you want to configure from the terminal, memory, or the network. The default is the console terminal, so press **Enter** to continue.
3. After you are in the Configuration mode (at the Config prompt), you then have to specify the IOS device that you want to configure. So to configure the serial 1 interface, type `interface serial 1` (see Figure 9.2).

FIGURE 9.2

Configuration commands are often done in steps, where you specify the item to configure and then provide the configuration information.

```

popeye#conf t
Configuring from terminal, memory, or network (terminal)?
Enter configuration commands, one per line. End with CNTL/Z.
popeye(config)#interface serial 1
popeye(config-if)#encapsulation ppp
popeye(config-if)#end
popeye#
04:39:56: %SYS-5-CONFIG_I: Configured from console by console
popeye#

```

Encapsulation isn't just a fancy word

When you are configuring serial interfaces with particular WAN protocols, you use the encapsulation command followed by the name of the protocol.

Encapsulation is the packaging of data in a particular protocol header. For example, Ethernet data is encapsulated in an Ethernet header before being placed on the network. In cases where Ethernet frames are moved across a WAN connection, the entire frame is placed in (or encapsulated in) a frame type dictated by the WAN protocol used, such as HDLC or PPP. Encapsulation will be discussed in Chapter 10, "TCP/IP Primer," and Chapter 15, "Configuring WAN Protocols," as it relates to specific protocols.

4. When you press **Enter**, the prompt changes to **config-if**, meaning that you have specified a particular interface (in this case, serial 1).
5. Now you can type in the command that will actually change the configuration of the specified interface. To enable point-to-point protocol for example, type **encapsulation ppp**, and then press **Enter**.
6. You can now type additional commands that will configure parameters related to the serial 1 interface. When you have completed the configuration, type **end** (or press **Ctrl+Z**), and then press **Enter**. This ends the configuration session for the serial 1 interface.

As you can see, configuration commands move from the general to the specific. First, you let the IOS know that you want to configure something, then you let it know what you want to configure, and then you provide it with the specific configuration parameters. Much of your router configuration takes this format. There are, however, some configuration commands that can be fired off as one-liners such as the **hostname** command, which enables you to change the name of your router. In fact, router configuration commands can be broken down into three categories:

- **Global commands**—*Global commands* are self contained, one-line commands that affect the overall global configuration of the router. Examples are **hostname** and **enable secret** (which sets the secret password for the Privileged mode). These types of commands are global because they affect a parameter that affects the overall functionality of the router, such as the router's name or the password you type to get in the Privileged mode.

- **Port Commands**—*Port Commands* are a set of commands that enable you to specify a particular interface or controller for configuration; these commands must be followed by subcommands that provide additional configuration information related to a particular interface or controller. For example, a port command to specify that serial 0 should be configured would be `interface serial 0`.
- **Subcommands**—*Subcommands* provide specific configuration information for the interface or controller that you specify with a particular port command. For example, to provide an IP address for a serial interface, you would type `IP Address` followed by a specific IP address and subnet mask.

You will have an opportunity to work in the Configuration mode later in this chapter and subsequent chapters where configuration information is given for specific network protocols such as IP, IPX, and AppleTalk.

The IOS Help System

No matter what mode you are in, the Cisco IOS can provide help. Now, I'm not talking about the handholding type of help you are used to getting with the various Windows-based programs that you probably use, but a more subtle type of help that is pretty decent for a command-line interface.

Suppose you are in the User mode and would like to see a complete list of the commands available. Type `?` and then press **Enter**. The commands are listed on the console screen as shown in Figure 9.3.

Okay, so now after checking out the commands available in the User mode, you find that you would like to use a particular command, but would like additional help on how that command should be entered at the prompt. For example, let's say you would like to see how to use the `show` command. Type `show` (or the command that you want to get help with) at the prompt followed by `?` (place a space between `show` and the question mark or you will get a "bad command" notification), and then press **Enter**. You will be provided with help specific to the chosen command, as shown in Figure 9.4.

FIGURE 9.3

You can get help in any of the router modes; type ? and then press Enter.

```

Tera Term - COM2VT
File Edit Setup Control Window Help

popeye>?
EXEC commands:
access-enable  Create a temporary Access-List entry
clear          Reset functions
connect        Open a terminal connection
disable        Turn off privileged commands
disconnect     Disconnect an existing network connection
enable         Turn on privileged commands
exit           Exit from the EXEC
help           Description of the interactive help system
lock           Lock the terminal
login          Log in as a particular user
logout         Exit from the EXEC
ninfo          Request neighbor and version information from a multicast
               router
netstat        Show statistics after multiple multicast traceroutes
ntrace         Trace reverse multicast path from destination to source
name-connection Name an existing network connection
pad            Open a X.29 PAD connection
ping           Send echo messages
ppp            Start IETF Point-to-Point Protocol (PPP)
resume         Resume an active network connection
vlogin         Open an vlogin connection
show           Show running system information
slip           Start Serial-line IP (SLIP)
sysstat        Display information about terminal lines
telnet         Open a telnet connection
terminal       Set terminal line parameters
tracroute      Trace route to destination
tunnel         Open a tunnel connection
where          List active connections
x2             Set X.3 parameters on PAD

popeye>

```

FIGURE 9.4

You can get help on specific commands.

```

Tera Term - COM2VT
File Edit Setup Control Window Help

popeye>show ?
clock          Display the system clock
dialer         Dialer parameters and statistics
history        Display the session command history
hosts          IP domain-name, lookup style, nameservers, and host table
hub            Hub status and configuration
location       Display the system location
modencap       Show Modem Capabilities database
ppp            PPP parameters and statistics
rmon           RMON statistics
sessions       Information about Telnet connections
snmp           SNMP statistics
tacacs         Show tacacs+ server statistics
terminal       Display terminal configuration parameters
traffic-shape  Display traffic shaping configuration
users          Display information about terminal lines
version        System hardware and software status

popeye>show

```

After providing help on the specific command, the command itself is automatically retyped for you at the command prompt (see Figure 9.4). You can then add specific parameters to the command and press Enter to execute it. For example, in the case of the show command, you can add version to the command and then press Enter. Parameters related to IOS currently installed on the router will be displayed on the screen (see Figure 9.5).

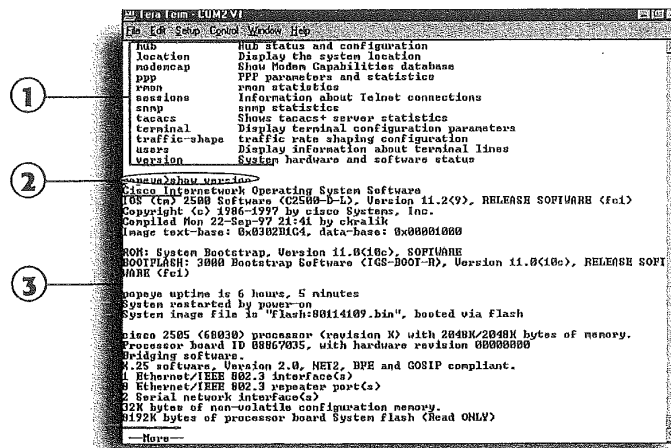


FIGURE 9.5

Use the Help system to correctly enter a particular command.

- 1 Help information
- 2 Completed command
- 3 Results of command

As stated before, the help system is also available in the Privileged and Configuration modes. The Privileged mode help is similar to that found in the User mode. You can receive general help by typing `?` or more specific help by typing a command followed by `?`.

Figure 9.6 shows the Help screen for the Privileged mode. Notice that it provides a larger number of commands than the User mode (which makes sense because the Privileged mode is a password-protected mode that provides greater access to the router).

You can also get help in the Configuration mode. For example, you may be in the middle of configuring a particular router interface and would like to see a list of subcommands available. Type `?` at the configure interface prompt and you will receive a list of available commands, as shown in Figure 9.7.

How to get more

When the information provided by a particular command (such as `?`) doesn't fit on one console screen, more will appear at the bottom of the displayed information. To move down through the additional information, press **Enter** to advance one line and press the **Spacebar** to advance one screen. In cases where you don't want to view more information, and want to return to the console prompt, press **Escape** (Esc).

Router Examination Commands

When you work in the Exec modes (User and Privileged) a number of the commands you use center around examining the various configuration settings and hardware parameters of the router. One of the most useful commands is the `show` command. You can use this command to view the status of all the interfaces on the router and view the statistics for such items as Flash RAM and the network protocols

currently being routed. You will find the show command invaluable in both the User and Privileged modes.

FIGURE 9.6

The Privileged mode provides a larger set of commands than the User mode does.

```

C:\Term - COM2.VI
File Edit Setup Control Window Help

popeye#?
Exec commands:
access-enable      Create a temporary Access-List entry
access-template    Create a temporary Access-List entry
bps                For manual emergency nodes setting
clear              Reset Functions
clock              Manage the system clock
configure          Enter configuration mode
connect            Open a terminal connection
copy               Copy configuration or image data
debug             Debugging functions (see also 'undebug')
disable            Turn off privileged commands
disconnect         Disconnect an existing network connection
enable             Turn on privileged commands
erase              Erase flash or configuration memory
exit              Exit from the EXEC
help              Description of the interactive help system
lock              Lock the terminal
login             Log in as a particular user
logout            Exit from the EXEC
nriinfo            Request neighbor and version information from a multicast
router
nstat             Show statistics after multiple multicast traceroutes
ntrace            Trace reverse multicast path from destination to source
name-connection   Name an existing network connection
no                Disable debugging functions
pad               Open a K.23 PAD connection
ping              Send echo messages
ppp               Start IETF Point-to-Point Protocol (PPP)
reload            Halt and perform a cold restart
resume            Resume an active network connection
rlogin            Open an rlogin connection
reload            Execute a remote command
send              Send a message to other tty lines
setup             Run the SETUP command facility
show              Show running system information
slip              Start Serial-line IP (SLIP)
--More--
  
```

FIGURE 9.7

Help is available even in the Configuration mode.

```

C:\Term - COM2.VI
File Edit Setup Control Window Help

popeye(config-if)#?
Configuring from terminal, memory, or network (terminal)?
Enter configuration commands, one per line. End with CNTRL/Z.
popeye(config-if)#?
Interface configuration commands:
access-expression  Build a bridge boolean access expression
appletalk          Appletalk interface subcommands
arp                Set arp type (arpa, probe, snap) or timeout
autotest           Autotest Encapsulations on Serial interface
backup            Modify dial-backup parameters
bandwidth          Set bandwidth informational parameter
bridge-group       Transparent bridging interface parameters
carrier-delay      Specify delay for interface transitions
cdp                CDP interface subcommands
clock              Configure serial interface clock
compress           Set serial interface for compression
custom-queue-list  Assign a custom queue list to an interface
dca-terminal-timing-enable Enable DCE terminal timing
deconf             Interface DEConf config commands
default            Set a command to its defaults
delay             Specify interface throughput delay
description        Interface specific description
dialer             Dial-on-demand routing (DDR) commands
dialer-group       Assign interface to dialer-list
down-when-looped   Force looped serial interface down
dci                ATM-BKI configuration commands
encapsulation      Set encapsulation type for an interface
exit              Exit from interface configuration mode
fair-queue         Enable Fair Queuing on an interface
full-duplex        Configure full-duplex operational mode
half-duplex        Configure half-duplex and related commands
help              Description of the interactive help system
hold-queue         Set hold queue depth
idle-character     Set idle character type
ignore-dcd         Ignore dcd
invert-txc         Invert transmit clock
ip                Interface Internet Protocol config commands
lpx               Novell/IPX interface subcommands
keepalive          Enable keepalive
--More--
  
```


You've already seen in the preceding section that the User mode provides you with a set of commands that you can use to examine the router status, and it is actually a subset of commands that are available to you in the Privileged mode. And even though you are working with a subset of types of items you can view with the `show` command, you can actually learn quite a lot about how the router has been configured in the User mode.

So, suppose you are stuck in the User mode on a router (you don't have the Privileged mode password) and want to examine the router. The first thing you would like to view is the interfaces available on the router.

Using the `show interfaces` command

1. At the User prompt, type `show interface`.
2. Press `Enter` to execute the command.

The results of the command will appear on the router console screen. Figure 9.8 shows the results of the `show interfaces` command on a 2505 router that has one Ethernet and two serial interfaces. It shows one screen-full of information; to see the rest of the output, you would have to press the `Spacebar`.

Quite a lot of information is provided by this one command. The hardware address (MAC) and the IP address are shown for Ethernet 0. The status of the interface (such as up or down) and the status of the protocol (or protocols) configured on that interface also appear. Additional information relates to the number of packets that have been input and output by the interface. Because this is an Ethernet interface (which uses CSMA/CD as the network access strategy), the number of collisions and illegal frames (giants and runs) are also provided.

Information on the other interfaces on the router will also be provided by this command. Note the Serial 0 interface information shown in Figure 9.8. The IP address for the interface is shown and the encapsulation type, PPP (which is the WAN protocol being used on this interface).

Command-line savvy

When you are working with the CLI there are some keystrokes that will help you if you make a mistake in a command and want to edit it before you execute it. Press **Backspace** to delete characters to the left of the cursor and then retype them. If you need to move to the beginning of the command line, press **Ctrl+A**. To move to the end of the line press **Ctrl+E**. Remember that you must press the **Enter** key to execute your commands.

FIGURE 9.8

The `show interfaces` command gives you information related to the interfaces installed on the router.

- ① Ethernet interface hardware address (0010.7b3a.50b3)
- ② Ethernet interface IP address (130.10.64.1/19)
- ③ Ethernet encapsulation type (Encapsulation ARPA)
- ④ Serial 0 IP address (130.10.32.1/19)
- ⑤ Serial 0 encapsulation type (Encapsulation PPP)

```

Termin-20M2V1
show interfaces
Ethernet0 is up, line protocol is up
Hardware is Lance, address is 0010.7b3a.50b3
Internet address is 130.10.64.1/19
MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load 1/255
Encapsulation ARPA, loopback not set, keepalive set (10 sec)
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:30, output 00:00:07, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
287 packets input, 35292 bytes, 0 no buffer
Received 202 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 input packets with dribble condition detected
7051 packets output, 663296 bytes, 0 underruns
0 output errors, 0 collisions, 3 interface resets
0 babbles, 0 late collision, 0 deferred
0 lost carrier, 0 no carrier
0 output buffer failures, 0 output buffers swapped out
Serial0 is up, line protocol is up
Hardware is HD64570
Internet address is 130.10.32.1/19
MTU 1500 bytes, BW 2000 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation PPP, loopback not set, keepalive set (10 sec)
LCP Open
Open: LCP, CDP
Last input 00:00:06, output 00:00:04, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/64/0 (size/threshold/drops)
Conversations 0/2 (active/max active)
Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
16075 packets input, 586464 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
More--
  
```

The `show interfaces` command will give you information on all the interfaces on a particular router. In the case of the 2505 router, I would have to press the **Spacebar** to show the next screen so that I can see the parameters related to the Serial 1 interface on the router. If you are using a higher-end router with several interfaces, you will have to continue to press **Enter** or the **Spacebar** to view the information. When you have come to the end of the information provided by the command, you will be returned to the user prompt.

If you find that `show interfaces` provides you with more information than you need and you just want to hone in on a particular interface on the router, you can use the `show` command to view the parameters related to just one interface.

Narrowing the focus of the `show` command

1. At the user prompt, type `show interface Ethernet 0`.
2. Press **Enter** to execute the command.

You will see results similar to those shown in Figure 9.8, but only the information for the Ethernet 0 interface will be provided.

The show command can also be used to gather other information related to the router. Table 9.1 lists some of the additional show-related commands that you can use in the User mode (all these show derivations will also work in the Privileged mode).

Table 9.1 The *show* Command in the User Mode

Command	Provides
Show clock	The time and date settings for the router
Show version	The version of the IOS currently running on the router
Show protocols	Lists the network protocols configured on the router
Show processes	CPU utilization information
Show history	A list of your last 10 commands
Show hub	Information on the status of the hub ports of a 2505 router

A number of other show-related commands exist. I will discuss several more show commands in the context of the particular network or routing protocol that they are used to monitor.

SEE ALSO

- For more information on using show to view IP-related parameters, see page 195.
- For more information on using show to view IPX-related parameters, see page 211.
- For more information on using show to view AppleTalk-related parameters, see page 227.

Abbreviate your commands

You will find that the Cisco IOS commands can be abbreviated in many cases. For example, rather than typing the show command, you can get away with the abbreviation sh. The abbreviated form of interface Ethernet 0 would be int E0. So the entire command to show interface Ethernet 0 would be sh int E0. Try your own abbreviated forms of commands as you work with your router. The worst thing that will happen is that the command interpreter won't recognize the command and let you know that there was invalid input or an incomplete command.

Using the Privileged Mode

The Privileged mode also allows you take advantage of all the show commands discussed in the previous section and several others that aren't available in the User mode. You will learn some of these "privileged" show commands, such as show running-config, in the "Checking Router Memory" section of this chapter.

More importantly, the Privileged mode provides you with the capability to access more complete information on the router's configuration and set operating system parameters (and you already know that you must be in the Privileged mode to enter the router's

Configuration mode). Let's say that you would like to set the system clock for the router; you must do it in the Privileged mode.

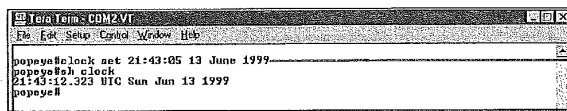
Setting the time and date

1. At the User prompt, type `enable`, and then press **Enter**.
2. Type the Privileged mode password and press **Enter**. You are now in the Privileged mode.
3. Type `clock set` followed by the time, day, month, and year; a correct entry for the time would be `clock set 21:43:05` (hour, minutes, seconds); a correct entry for the date would be `13 June 1999`. Using the example data shown, the complete command would read `clock set 21:43:05 13 June 1999`, as shown in Figure 9.9.
4. Press **Enter** to execute the command.
5. To check the new settings type `show clock`, and then press **Enter** (see Figure 9.9).

FIGURE 9.9

You can set the time and date on the router using the `clock set` command.

- ① The `clock set` command



Several other Privileged commands exist that you will use on a regular basis. For example, `show cdp neighbors` is an internetwork exploratory tool that I will discuss in the "Checking Out the Internetwork Neighborhood" section found later in this chapter. Other Privileged commands are discussed in the next section.

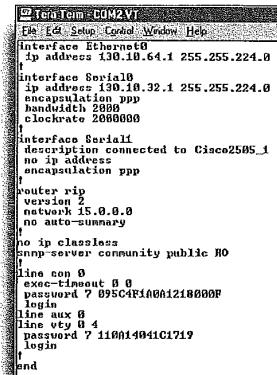
Checking Router Memory

When you configure the various interface and protocol parameters for a router, this information is stored in the router's RAM. It's important that you store this information somewhere, in case the router loses power. In the Privileged mode you can save your running configuration to NVRAM where it becomes the router's startup configuration (and is loaded if the router is rebooted).

The Privileged mode also allows you to examine the contents of RAM and NVRAM using the show command. These commands aren't available in the User mode.

Viewing the running configuration

1. At the User prompt, type **enable**, and then press **Enter** (if you aren't in the Privileged mode).
2. Type the Privileged mode password and press **Enter**. You are now in the Privileged mode.
3. Type **show running-config**, and then press **Enter** to execute the command. The command results will appear on the router (see Figure 9.10).
4. To advance through the information on the screen, press **Spacebar** for an entire screen or **Enter** to advance line by line.



```

Termin - COM2VT
File Edit Setup Control Window Help
Interface Ethernet0
  ip address 130.10.64.1 255.255.224.0
Interface Serial0
  ip address 130.10.32.1 255.255.224.0
  encapsulation ppp
  bandwidth 2000
  clockrate 2000000
Interface Serial1
  description connected to Cisco2505.1
  no ip address
  encapsulation ppp
router rip
  version 2
  network 15.0.0.0
  no auto-summary
no ip classless
snmp-server community public RO
line con 0
  exec-timeout 0 0
  password 7 095C4F1A001218000F
  login
line aux 0
line vty 0 4
  password 7 110014041C1719
  login
end
  
```

FIGURE 9.10
Show running-config displays the entire running configuration for the router.

The *running configuration* provides information on how the different interfaces are currently configured and which routing protocols have been enabled. It also shows the passwords that have been set on the router (however, remember that the Privileged mode secret password is encrypted, so you can't tell what it is). The running-config command provides a complete picture of the parameters running on the router, and this is why it is a Privileged mode command; it's information important to the router's administrator, so it should be protected.

Scroll through a list of recent commands

You can use the Up Arrow key on the keyboard to cycle through the commands that you recently used. Press the Up Arrow and you will see the last command used (it is placed at the router prompt); continue to press the Up Arrow and your commands (the last 10 from most to least recent) will appear one by one. To fire off a recycled command, just use the Up Arrow key to place the appropriate command at the prompt, and then press **Enter**.

Remember to exit the Privileged mode

When you finish working in the Privileged mode type **disable**, and then press **Enter** to return to the User mode. This will protect your router from being reconfigured by an overly zealous coworker or corporate terrorist who is trying to bring down your Silly Putty manufacturing empire.

FIGURE 9.11 **show flash** displays the IOS file in flash and the amount of flash available.

① OS filename

As you fine-tune your running configuration, a time will come when you would want to save it to NVRAM as the startup configuration. The great thing about the **copy** command is that you can copy information from RAM to NVRAM (running to startup). Or if you mess up your running configuration, you can copy information from NVRAM to RAM (startup to running). The command you use to copy information from one type of memory to another is **copy**.

Copying the running configuration

1. In Privileged mode, type **copy running-config startup-config**.
2. Press **Enter** to execute the command.

The router will pause for a moment. Building configuration will be displayed on the screen. Then “[OK]” will appear. The running configuration has been copied to the startup configuration. You can quickly check your new startup configuration with the **show startup-config** command (the output will be similar to the **running-config** shown in Figure 9.10). The results of this command also show you how much NVRAM is being used on the system to store the configuration file.

Another memory type on the router is *Flash RAM*. This is where the router's IOS is stored. You can view the contents of Flash in both the User and Privileged mode.

Viewing Flash contents

1. In the Privileged or User mode, type **show flash**.
2. Press **Enter** to execute the command.

The results of the command will appear on the console screen (see Figure 9.11). The IOS filename is given and the amount of free and used Flash RAM is displayed.

```

C:\Term: COM2.VT
File Edit Setup Control Window Help
popeye@show flash
System Flash directory:
File Length Name/status
1 6334792 801/489.kim
(5334956 bytes used, 3853752 available, 8388608 total)
8192K bytes of processor board System Flash (Read ONLY)
popeye@
  
```


Checking Out the Internetwork Neighborhood

When you work with internetworks, it's important to be able to gather information related to routers that are directly connected to your router. These routers are typically referred to as *neighbors*. Cisco routers have a proprietary protocol, *Cisco Discovery Protocol (CDP)*, that provides you with the capability to access information related to neighboring routers. CDP uses Data Link broadcasts to discover neighboring routers that are also running CDP (CDP is turned on automatically on routers running IOS 10.3 or newer).

Working with CDP

Before you use CDP to view information about other routers, you may want to check your router interfaces to make sure that CDP is enabled. This is done using the `show cdp interface` command.

Viewing CDP interfaces

1. At the User or Privileged prompt type `show cdp interface`.
2. Press **Enter** to execute the command.

The results of the command will appear on the router console screen (see Figure 9.12). The CDP information for all the interfaces on the router will appear.

Make sure your running configuration works

You will want to put a new running configuration through its paces (let it run for a while and monitor router parameters using the `show` command and a command I haven't discussed yet called `debug`) before you save it as the router's startup configuration. You may also want to back up the original startup configuration to a TFTP server before you save a new running configuration as the startup configuration (covered in Chapter 17).

```

Tera Term - COM2VT
File Edit Setup Control Window Help
popes@shou cdp interface
Ethernet0 is up, line protocol is up
  Encapsulation ARPA
  Sending CDP packets every 60 seconds
  Holdtime is 180 seconds
Serial0 is up, line protocol is up
  Encapsulation FDDI
  Sending CDP packets every 60 seconds
  Holdtime is 180 seconds
Serial0.1 is deleted, line protocol is down
  Encapsulation FDDI
  Sending CDP packets every 60 seconds
  Holdtime is 180 seconds
Serial1 is down, line protocol is down
  Encapsulation FDDI
  Sending CDP packets every 60 seconds
  Holdtime is 180 seconds
popes#
popes@shou cdp interface serial 0
Serial0 is up, line protocol is up
  Encapsulation FDDI
  Sending CDP packets every 60 seconds
  Holdtime is 180 seconds
popes#
  
```

FIGURE 9.12
The `show cdp interface` command shows which interfaces are enabled for CDP.

Working with flash

You erase the contents of Flash in the Privileged mode (not generally a good idea) using the `erase` command; you can also load a new version of the IOS into Flash using a TFTP server and the `copy` command, which is discussed in Chapter 17.

CDP doesn't care about network protocols

CDP is platform-independent, so it will accumulate information about neighbor routers no matter which network protocol stack they might be running (such as TCP/IP, IPX/SPX, and so on).

Changing CDP holdtime

You can manually set the holdtime for CDP in the configuration mode. At the configuration prompt type `cdp holdtime seconds`, where *seconds* is the time interval for the holdtime.

You can also view the CDP information for a particular interface. For example, in Figure 9.12, the command that follows the initial `show cdp interface` command is `show cdp interface s0`. This provides the CDP information for just interface serial 0.

In Figure 9.12, you will see two pieces of information that warrant further discussion: the CDP packet send interval and the CDP holdtime. Notice that CDP packets are sent by CDP-enabled interfaces every 60 seconds. This means that they are broadcasting information to their CDP neighbors every minute.

The *holdtime* refers to the amount of time a router should hold the CDP information that it has received from a neighboring router. If a router doesn't receive an update message from a neighbor within three minutes (180 seconds), it must discard the old CDP information that it holds.

Remember that the purpose of CDP is to stay up to date on the status of your neighboring routers. So, if a line is down or some other problem causes you to lose contact with a neighbor, you don't want your router relying on old information when it makes routing decisions.

If a particular interface isn't enabled for CDP, you can enable it in the configuration mode.

Enabling CDP on an interface

1. At Privileged prompt type `config terminal`. You are placed in the configuration mode with the console (terminal is the source for the configuration information).
2. At the Config prompt type the interface you want to enable for CDP, such as `interface serial 0`. Then press **Enter**. The prompt changes to the Config-If prompt, letting you know that you can now enter information for the configuration of the designated interface.
3. Type `cdp enable`, and then press **Enter**.
4. To end the configuration of the serial interface, press **Ctrl+Z**. You will be returned to the Privileged prompt (see Figure 9.13).


```

TeraTerm - COM2-VT
File Edit Setup Control Window Help
popaya#config terminal
Enter configuration commands, one per line. End with CNTL/Z.
popaya(config)#interface serial10
popaya(config-if)#cdp enable
popaya(config-if)#^Z
popaya#
1022h: x86S-5-CONFIG-I: Configured from console by console
popaya#

```

FIGURE 9.13

You can easily enable an interface for CDP if it has been previously disabled.

Viewing CDP Neighbors

After you have viewed the status of CDP on your various interfaces, you can use CDP to take a look at platform and protocol information on a neighboring router or routers.

Viewing CDP neighbors

1. At the User or Privileged prompt type `show cdp neighbors`.
2. Press **Enter** to execute the command.

Figure 9.14 shows the result of this command for a 2505 router that only has one neighbor, which is connected via a serial interface. Table 9.2 describes the information shown in Figure 9.14.

```

TeraTerm - COM2-VT
File Edit Setup Control Window Help
popaya#show cdp neighbor
Capability Codes: R - Router, I - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, F - FIBR, r - Repeater
Device ID        Local Interface  Holdtime  Capability  Platform  Port ID
olive            Ser 0          126       R           2505      Ser 0
popaya#

```

FIGURE 9.14

The `show cdp neighbor` command lets you check your network neighborhood and view directly connected routers.

Table 9.2 The `show` Command in the User Mode

Parameter	Meaning	Example from Figure 9.14
Device ID	The neighbor's or neighbors' hostname(s)	Olive
Local Interface	The interface on the local router that provides the connection to the neighbor	Serial 0
Capability	Whether the router is configured to serve multiple functions such as routing (R), Bridging (B), and switching (S).	R (this router is only configured to route)

continues...

Disabling and enabling CDP

CDP can be disabled globally or on an interface-by-interface basis. To disable CDP globally, enter the Configuration mode and type `no cdp run`. This shuts it off on all interfaces. For a particular interface, enter the Configuration mode, specify the interface you want to disable, and then use the command `no cdp enable`. The global command for turning CDP on is `cdp run` and is used at the Privileged prompt.

Table 9.2 Continued

Parameter	Meaning	Example from Figure 9.14
Platform	The type of Cisco router.	2505 (the neighbor is a 2505 router)
Port ID	The interface used on the neighbor to connect to your local router	Serial 0

Obviously, if you are using a higher-end router that is connected to many different neighbors via its various interface ports, the number of neighbors shown using the `show cdp neighbors` command would be greater than that shown in Figure 9.14.

If you want to see more details concerning your CDP neighbors, you can use the `show cdp neighbor details` command. You can enter this command at the User or Privileged prompt. Figure 9.15 shows the results of this command. Notice that this command provides the IP address of the neighbor's interface and the version of the IOS that the neighbor is running.

Using Ping

A command that can be very useful when you are working with routers is ping. And if you use the Internet a great deal you may have already used this command to test the lag time between you and another computer on the Net. *Ping* (which is short for *Packet InterNet Groper*) is used to test the connection between two or more nodes on a network. These nodes can be host computers, servers, or routers.

Ping can be used with a number of Layer 3 protocols such as IP, IPX, and AppleTalk, and uses the logical address assigned to the node on the network. On routers, you can Ping different interfaces because in most cases they will each be assigned a logical address. For example, if you are routing IP, each interface on your router will probably be assigned an IP address.

For example, let's say you want to see whether your connection to another router is up and running. All you have to do is ping the interface on the other router that your router is connected to.

Pinging a neighbor

1. At the User or Privileged prompt type `ping ip address`. In this case you are trying to ping the Olive router that is connected to your router via a serial interface. So, the command reads `ping 130.10.32.2`.
2. Press **Enter** to execute the command.

The results of the Ping command appear in Figure 9.15. Notice that the success rate is 100%. In cases where you can't reach the node that you've pinged, the success rate will be 0%.

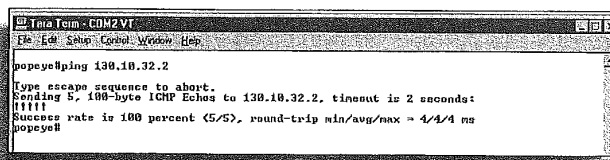


FIGURE 9.15
ping can be used to check your connection to a particular router on the internetwork.

Ping will be discussed in more detail later in this book (as will the Trace and Extended Ping commands in Chapter 18, "Basic Troubleshooting").

SEE ALSO

- For more about ping and extended ping, see page 314.
- Another TCP/IP protocol stack member, Telnet, can also be used to connect to other routers on the internetwork. For more information, see page 209.

Creating a Router Banner

You have explored the Cisco IOS in the User and Privileged mode (and worked with a number of different and useful IOS commands) in this chapter, and you should also spend some time working in the Configuration mode. Because several chapters are devoted to configuring specific LAN, WAN, and routing protocols on the router, let's work on something fun in the Configuration mode—the creation of a banner. This banner will appear on your console screen when the router is booted (or rebooted) and will also appear on the screen of virtual terminals that are used to log in to your router (using Telnet, which is discussed in Chapter 11, "Configuring IP Routing").

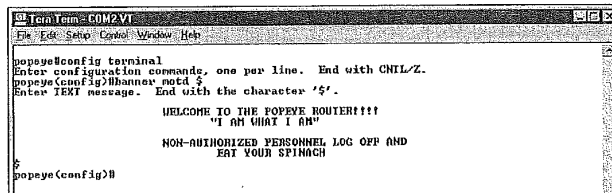
The router banner is created in the Configuration mode. The command is `banner motd end character`; where the end character is a keyboard character of your choice that tells the configuration mode when you have completed your banner text (`motd` actually stands for message of the day). For example, you will want to choose a character such as the number sign (#), dollar sign (\$), or other character that will not appear in the body of your banner (such as most letters of the alphabet).

Creating a router banner

1. At the Privileged prompt type `config terminal`. You are placed in Configuration mode with the console (terminal is the source for the configuration information).
2. I will use the dollar sign (\$) as our end character. Type `banner motd $`. Then press **Enter**. You will be told to type your banner text and end the banner with the \$ character.
3. Type the text for your banner. Use the **Enter** key to place blank lines in the banner text. Use the **Spacebar** to position items from left to right in the banner. Figure 9.16 shows a sample router banner.
4. Type your selected end character (\$) in this case) and press **Enter**. You will be returned to Configuration mode.
5. Press **Ctrl+Z** to save your banner and exit Configuration mode.

FIGURE 9.16

You can create a banner for your router in the configuration mode.



After exiting the Configuration mode, you may have to press **Enter** once to return to the Privileged prompt. To view your router banner, type `quit` and press **Enter**.

This exits you from the router. When you press **Enter** on the initial router screen, your router banner will appear (see Figure 9.17). If you have set up the router with a login password, you will be asked to provide the password to enter the router.

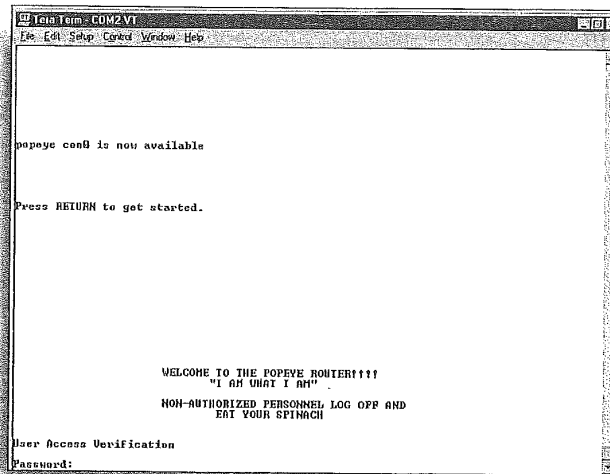
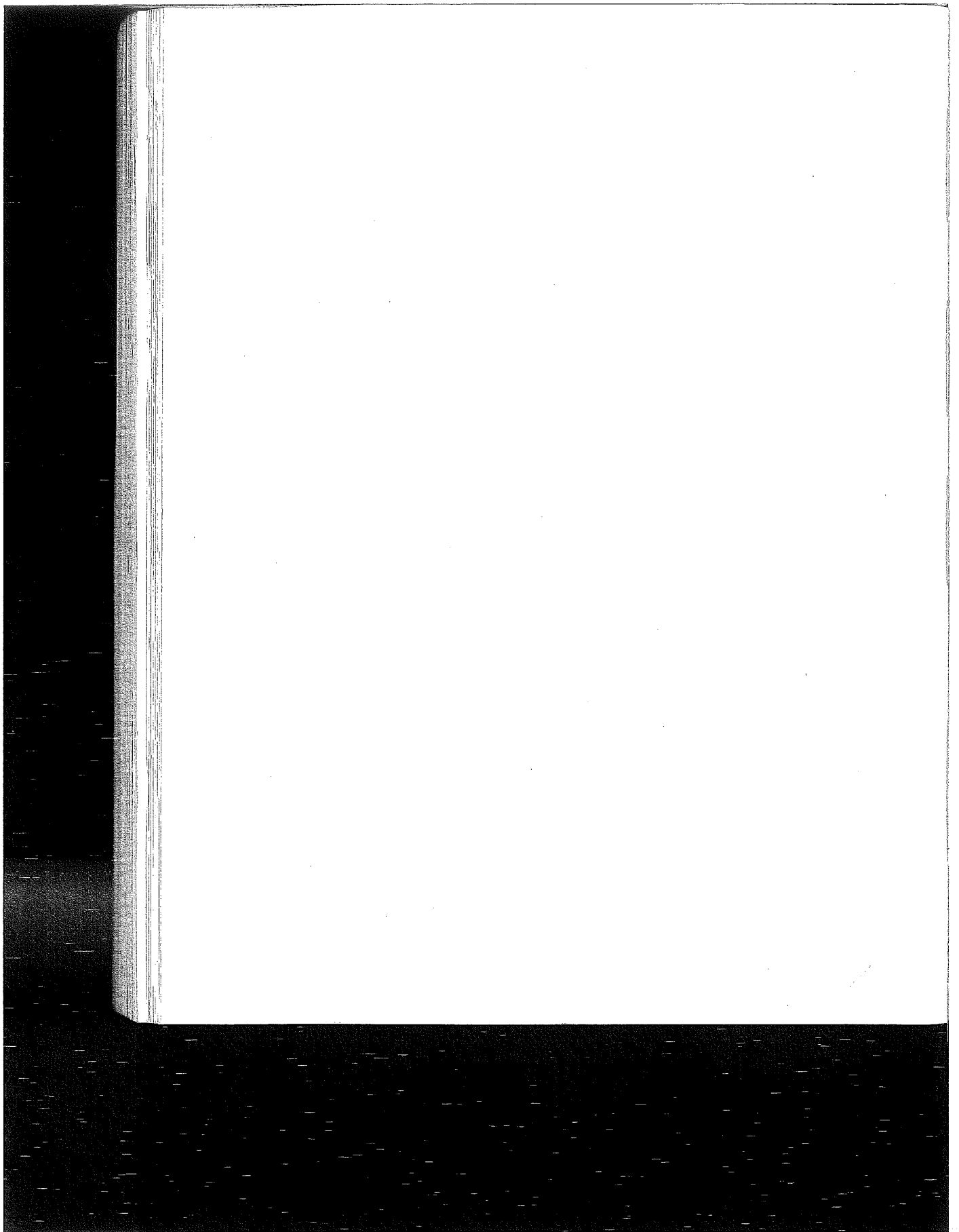


FIGURE 9.17
Your banner will appear
when you attempt to log
on to the router.

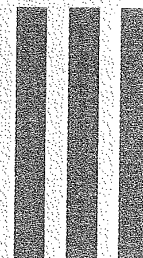
As you can see from this chapter, the Cisco IOS provides a large and robust command set. You will remember the commands that you use often and probably have to look up the commands that you don't. A summary of the basic commands covered in this book is available in Appendix A, "Basic Router Command Summary," as a resource.

SEE ALSO

- » For more information on setting passwords on the router (in the Configuration mode), see page 137.
- » The password commands also appear in the command reference in Appendix B; see page 323.

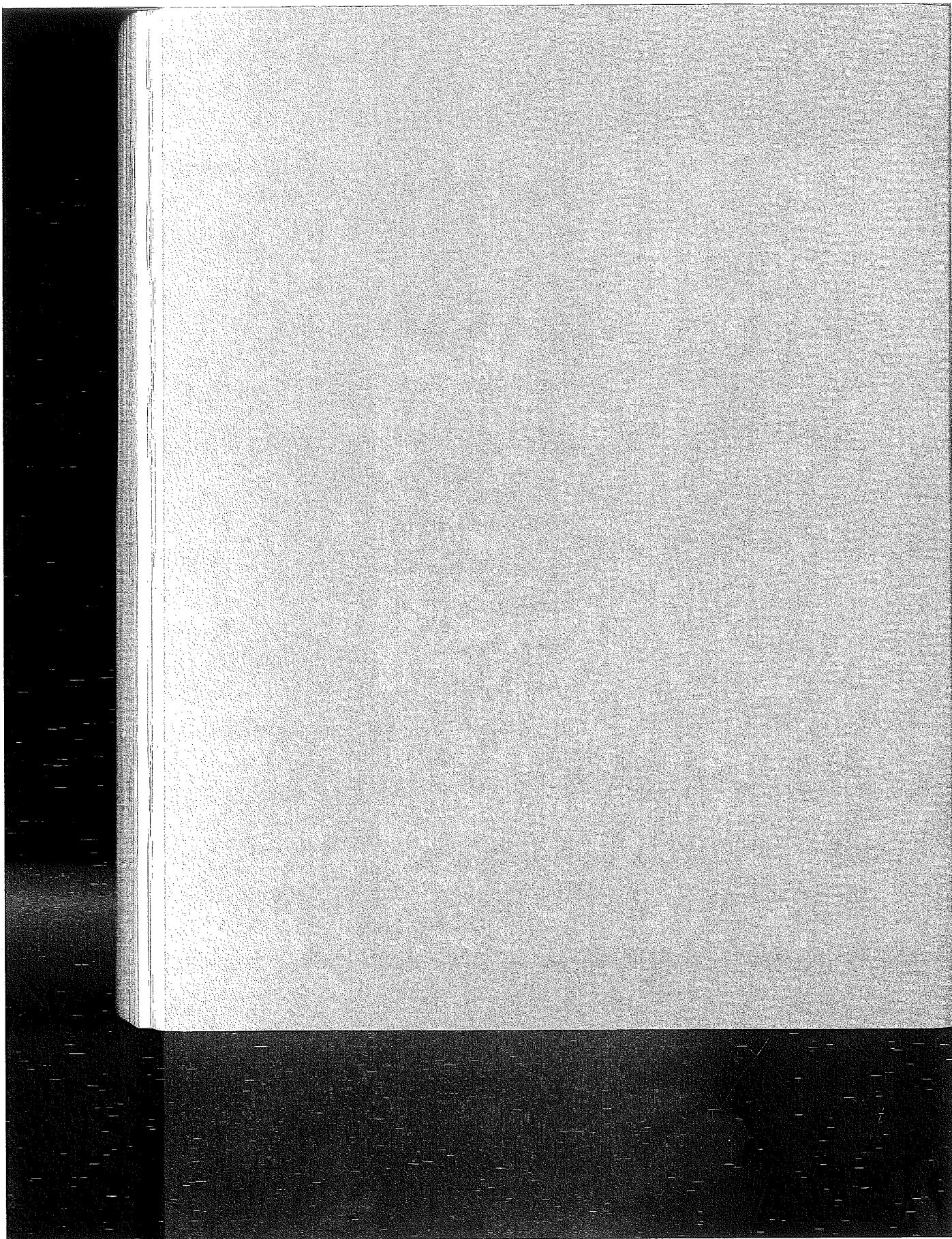


part



ROUTING LAN PROTOCOLS

TCP/IP Primer	167	10
Configuring IP Routing	195	11
Routing Novell IPX	211	12
Routing AppleTalk	227	13



chapter

10

TCP/IP Primer

The TCP/IP Protocol Stack

TCP/IP and the OSI Model

Working with IP Addresses

Subnetting IP Addresses

Creating Class B and Class C Subnets

A Final Word on Subnetting

The TCP/IP Protocol Stack

TCP/IP (Transmission Control Protocol/Internet Protocol) has become the common language for the networking world and is a commonly deployed protocol suite on enterprise networks. It is also the foundation for the worldwide Internet—the mega network of networks. Many network operating systems (NOS), such as Windows NT 4.0 Server, Windows 2000 Server, and Novell Netware 5.0, embrace TCP/IP as their default networking protocol.

I discussed TCP/IP briefly in the Chapter 2, “The OSI Model and Network Protocols.” And as you already know, TCP/IP was developed originally as a set of WAN protocols that could be used to maintain communication links between sites even if certain sites became inoperable during a worldwide nuclear war. In light of the kind of fun people have on the Internet today using the TCP/IP stack, it is somewhat ironic (and somewhat depressing) that the suite was originally developed as a sort of wartime network failsafe system by the Department of Defense.

Another point that must be made about TCP/IP is that it has become an integral part of operating and supporting routers on an internetwork. Cisco router administrators use Telnet (a member of the TCP/IP stack) to communicate with remote routers and use TFTP (another TCP/IP protocol) as a mechanism for copying and saving configuration files and loading new IOS software on the router. Most big networks use TCP/IP as their network protocol, so a lack of understanding of the TCP/IP stack will make it pretty hard for you to work with routers and internetworks. TFTP is discussed in more detail in Chapter 17, “Using a TFTP Server for Router Configuration Storage.”

SEE ALSO

➤ To check out some of the other overview information on TCP/IP, see page 45.

TCP/IP and the OSI Model

TCP/IP was developed in the 1970s and so preceded the completion of the OSI model (in the 1980s). This means that the different protocols in the TCP/IP stack don't map directly to a single layer in

the OSI model (although the lower-layer Network and Data Link protocols, such as IP and ARP, do map somewhat closely to their conceptual equivalent in the OSI model).

When TCP/IP was developed, the Department of Defense (DOD) developed its own conceptual model—the *DOD model*—(also known as the DARPA model) for how the various protocols in the TCP/IP stack operate. This reference model divides the movement of data from a sending node to a receiving node into four layers (compared to the seven layers of the OSI model). Figure 10.1 shows how the DOD model maps to the OSI model.

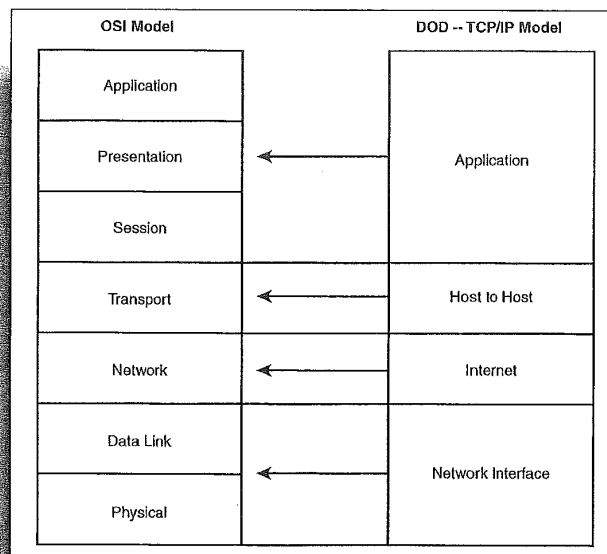


FIGURE 10.1
The DOD four layer model mapped to the seven layers of the OSI model.

Each layer in the DOD-TCP/IP conceptual stack defines the job that TCP/IP protocols do that operate at that particular level (just as the OSI model does). In the next four sections you will take a look at what happens at each layer of the DOD-TCP/IP conceptual stack and the actual TCP/IP stack protocols that operate at these levels. Figure 10.2 shows the TCP/IP stack mapped to the DOD model.

SEE ALSO

» To review the OSI model, see page 34.

Application Layer

The Application layer protocols provide the user interface for the various protocols and applications that access the network. Application layer protocols in the TCP/IP stack handle file transfer, remote login to other nodes, email functionality, and network monitoring. A number of different protocols reside at this level:

- *FTP (File Transfer Protocol)* is a protocol that provides the capability to transfer files between two computers. FTP is actually a full-blown application (FTP clients can be downloaded from the Internet and used to move files between computers) and a protocol that is supported by other applications such as Web browsers.
- *TFTP (Trivial File Transfer Protocol)* is a stripped down version of FTP that provides a way to move files without any type of authentication (meaning no username or password). TFTP is used in the router world as a way to save router configuration files or update the IOS of a router (this protocol is described extensively in Chapter 17).
- *SMTP (Simple Mail Transport Protocol)* is a protocol that provides mail delivery between two computers. It is a protocol supported by email clients and used for sending and receiving email on the Internet.
- *SNMP (Simple Network Management Protocol)* is a protocol that provides the capability to collect network information. SNMP uses *agents* (software watchdogs that keep an eye on network processes) that collect data on network performance. The collected data can then be compared to baseline information. Software packages like CiscoWorks use SNMP to help network administrators monitor the relative health of a network.
- *Telnet* is a terminal emulation protocol that allows you to connect a local computer with a remote computer (or other device such as a router). The local computer becomes a virtual terminal that has access to applications and other resources on the remote computer. Telnet will be used to log on to a remote router from a local router in Chapter 11, "Configuring IP Routing."

Host-to-Host Layer

The Host-to-Host layer protocols provide flow control and connection reliability as data moves from a sending to a receiving computer. This layer takes the data from the Application layer protocols and begins the process of readying the data for movement out over the network. Two TCP/IP suite protocols inhabit the Host-to-Host layer: TCP and UDP.

- *TCP (Transport Control Protocol)* is a connection-oriented protocol that provides a *virtual circuit* (not unlike establishing a phone call between the sending and receiving nodes) between user applications on the sending and receiving machines. TCP takes the data from the Application layer protocols and breaks it into segments and then makes sure that they are reassembled on the receiving end. TCP requires that the sending and receiving computer establish a synchronized connection, which is done by the exchange of packets carrying sequencing numbers and a synch control bit. TCP requires a lot of network overhead.
- *UDP (User Datagram Protocol)* is a connectionless transport protocol that provides a connection between Application layer protocols that don't require the acknowledgements and synchronization provided by TCP. UDP is like sending a postcard through the mail system. The packet is addressed for the receiving node and sent on its way. UDP is much more passive than TCP. Application layer protocols that use UDP include TFTP and SNMP.

Internet Layer

The Internet layer (corresponding to the OSI Network layer) is responsible for the routing of data across logical network paths and provides an addressing system to the upper layers of the conceptual model. This layer also defines the packet format used for the data as it moves onto the internetwork. The Internet layer really revolves around one protocol—IP. Other protocols at this layer basically provide support for the IP addressing system and packet format. An important job of the Internet layer is resolving logical addresses (such as IP addresses) to the actual hardware (MAC) addresses of the nodes on the network.

IP datagrams are surrounded by MAC layer information

IP datagrams consist of an IP header, which contains the source IP address, the destination IP address (and some other IP related items), and the data provided by the upper-layer protocols. This datagram is sandwiched inside MAC layer header (containing information regarding the media access type, such as Ethernet or Token Ring) and MAC layer trailer, which contains the CRC check for the packet. In our DOD diagram the MAC layer protocols operated at the Network Access layer (described in the next section) and at the Data Link layer of the OSI model. This IP datagram is a good example of how the layers work together to get data to its destination.

Everything you ever wanted to know about IP

The entire TCP/IP stack and IP in particular (RFC 791) have been documented in RFC (Request For Comments) documents. These documents are available at a number of sites on the World Wide Web. Two locations that are good bets for finding a particular RFC are Ohio State's RFC repository at <http://www.cis.ohio-state.edu/hypertext/information/rfc.html>, and the Hyper-RFC site at <http://www.csl.sony.co.jp/rfc/>. Or you can just search the Web with RFC as your keyword.

Ping and traceroute use ICMP

Both ping and traceroute, a router command, use ICMP messages. Ping is introduced in Chapter 9, "Working with the Cisco IOS," and traceroute will be looked at in Chapter 18, "Basic Router Troubleshooting."

- **IP (Internet Protocol)**—IP takes the data from the Host-to-Host layer and fragments the information into packets or *datagrams*. It labels each packet with the IP address of the sending device and the IP address of the receiving device. IP also reassembles datagrams on the receiving machine into segments for the upper-layer protocols. IP is a connectionless protocol that has no interest in the contents of the datagrams. Its only desire is to address and move the datagrams toward their destination.
- **ARP (Address Resolution Protocol)**—When IP prepares a datagram, it knows the IP address of the sending and receiving computers (it receives this information from the upper layer protocols such as Telnet or SMTP). IP also needs the MAC hardware address for the receiving computer because it must provide this information to the Network Access layer protocol used on the network (such as Ethernet). ARP provides the mechanism for resolving the IP address to an actual hardware address. ARP sends out broadcasts with the receiving computer's IP address and asks the computer to reply with its hardware address.
- **ICMP (Internet Control Message Protocol)**—This protocol is a message service provider and management protocol that is used by routers to send messages to host computers that are sending data that must be routed. Routers can let the sending host know when a destination is unreachable or when the router's memory buffer is full of data. Again, ICMP is basically used as a support protocol for IP addressing as ARP is.

➤ The logical addressing system provided by IP is discussed in greater detail later in this chapter on page 180.

Network Access Layer

The Network Access layer consists of the protocols that take the datagrams from the Internet layer and envelope them in a specific frame type that is then placed on the network's physical medium as a bit stream. You are already familiar with these protocols, which were previously described as the Data Link layer protocols of the OSI model and include such network architectures as Ethernet, Token Ring, and FDDI. The IEEE specifications described in Chapter 2 provide the specifications for the different frame types used by these network architectures.

Because these protocols reside at the MAC layer (a part of the Network Access Layer of the DOD model and the Data link layer of the OSI model), they are integrally involved in the physical addressing of the data packets. The physical address for a computer is actually burned on the network interface card that is placed in the computer. Router Ethernet, Token Ring, and FDDI interfaces also have MAC addresses burned into the ROM chip of the controller for the interface (serial interfaces on routers don't have MAC addresses).

Figure 10.2 provides a summary of how the OSI model maps to the DOD model and provides the TCP/IP stack mapped to the DOD model. The protocols shown in the TCP/IP stack will be discussed further in respect to how they relate to routers and routing.

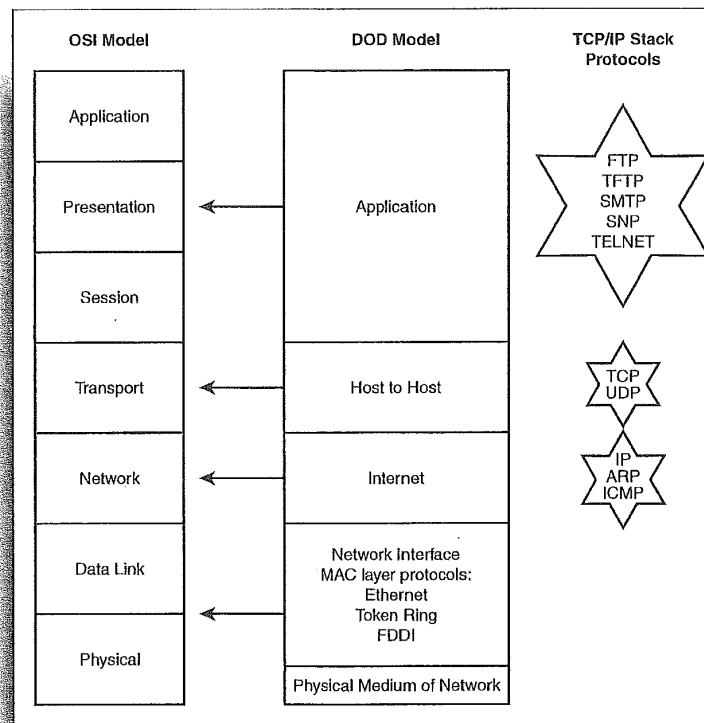


FIGURE 10.2
The DOD model and
TCP/IP stack mapped to
the OSI model.

An important question arises as to how the logical addressing system provided by IP is resolved to the MAC hardware addresses found on the various network nodes. This subject will be discussed in the next section, which provides a basic overview of IP addressing.

SEE ALSO

- *For an overview of network architectures, see page 25.*
- *For more information on the IEEE specifications, see page 47.*

Working with IP Addresses

IP addresses are 32 bits long and consist of four 8-bit octets (each octet is one byte). A typical IP address would be 200.1.25.7 (as shown in dotted decimal format). The IP address actually exists as a binary number (1s and 0s), which, as you will see when you get to subnetting, becomes very important in calculating subnets.

An IP address can be written in three different forms:

- Dotted decimal: 200.1.25.7
- Binary: 11001000 00000001 00011001 00000111
- Hexadecimal: C8 1 19 7

IP addresses are hierarchical addresses in that they provide different levels of information; they can tell you the network that the node resides on, the subnet it belongs to, and the actual node address. The IP addressing system isn't unlike the system used to designate your home address by the U.S. Post Office. A letter to you provides your street address, city, and state (and of course zip code). A number of people can live in your city or state, but only you live at the particular postal address.

IP addresses use this same strategy, so part of the IP address tells you the network the node is on, part of it tells you the subnet, and most importantly part of the IP address tells you the node designation. This type of addressing system makes routing practical because data can be forwarded by routers using the network and subnet information (they don't have to actually know the physical MAC address of the receiving node) to the router that serves that particular subnet (meaning it is connected to that subnet).

After the router that serves a particular subnet has the packets for a node that resides on that subnet, it can make sure the packets get to their final destination by resolving the IP address in the packet to the MAC address on the receiving computer. Again, this is like your home address; if you reside in California, mail from the East Coast is forwarded to intermediary post offices in the Midwest using the Zip Code and eventually arrives at your local post office. From your local post office your mail is “resolved” to your home address and delivered by a mail carrier.

Having a network portion to the address means that a router only has to know how to get the packets to that network address (through a maze of routers). And it gets help from the other routers on the internetwork as the packets hop from router to router on the way to their final destination.

Understanding which part of the IP address refers to the network subnet, and which part refers to the node, is a very important aspect of working with IP addresses. The next section explores the different IP classes and the subnet masks they use to make this whole IP addressing thing work.

IP Classes

IP addresses have been broken down into 3 classes based on the size of the network that they serve. There are Class A, Class B, and Class C IP internetworks.

- *Class A* is used for very large networks and supplies over 16 million node addresses for the network. Because of the way IP addresses are structured, a Class A network can serve a huge number of host computers (nodes), but there can only be 127 class A networks. The reason for this will become apparent shortly—read on! ARPAnet (built in the early days of the Internet) is an example of a Class A network.
- *Class B* is used for networks that still need a lot of node addresses, such as a large company or institution. There are 16,384 Class B network addresses, with each Class B supplying over 65,000 host addresses.

The reason MAC addresses aren't used for routing

The alternative to using a hierarchical addressing system like IP is to use flat addressing (there are MAC addresses on all the computers with network cards). However, with flat addressing systems like MAC addresses, routers would have to remember all the unique MAC addresses in the world (a technical impossibility). It would be like using social security numbers to deliver mail. A letter to you would be addressed with your social security number, meaning the routing of it to your home would require a Herculean effort by the postal system.

Getting your own IP numbers

There are a couple of ways that your company can be issued a range of IP numbers. You can get your IP addresses from your Internet Service Provider (which supplies you with a portion of the address pool that they have purchased). Or you can be issued your IP numbers directly by the American Registry for Internet Numbers. IP ranges aren't cheap, and you can check out the details at <http://www.arin.net/>. You will also need a domain name if you are going to have a company presence on the Web; go to www.internic.net for more information on establishing a domain name (such as Microsoft.com or Habraken.net).

Network 127 is reserved for loopback

In Table 10.1, you will notice that Class A networks end with a first octet decimal value of 126 and Class B networks begin with a first octet value of 128. So what happened to network 127? It is reserved for loopback testing that allows computers to send a packet to themselves without tying up network bandwidth.

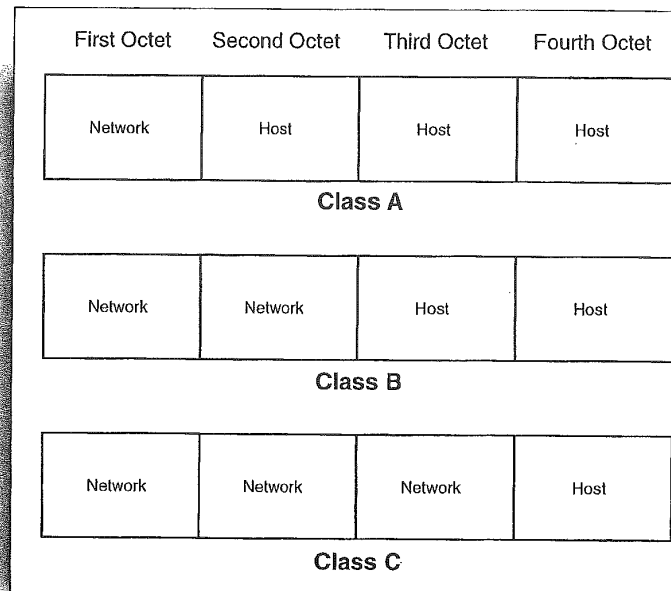
- *Class C* is used for small networks and there are over 2 million Class C network addresses available. Class C networks only provide 254 node addresses, however.

Each of these classes used a certain number of *octets* in an IP address to denote the network portion of the address and the node portion of the address. For example, a Class A IP address such as 10.5.25.8 denotes the IP network using the first octet. This means that the network number is 10. The rest of the address, 5.25.8, denotes the host address. So, if only the first octet is used for network addresses, there can only be a limited number of network addresses in a Class A (because you limit the possibilities to one octet), whereas you are using three octets to specify the host address, which gives you a lot of different possible combinations. This is why there are only a limited number of Class A networks available, but each Class A network supplies a huge number of host addresses (over 16 million).

Contrasting this Class A address with a Class C address, 200.44.26.3, will help emphasize the point. The first three octets of a Class C address denote the IP network number (200.44.26). Only the last octet is available to assign host numbers. So, you can see that having three octets available for network numbers gives you a huge number of possibilities; whereas using only one octet as a source of host addresses really limits the possibilities.

Figure 10.3 shows each of the IP Network classes and the octets that they use for network addresses and for host addresses. The greater the number of octets used for host addresses, the more possible hosts. The greater the number of octets used for network addresses, the greater the number of possible networks; it's that straightforward.

Table 10.1 summarizes the decimal range for the first octet of each of the IP network classes and the number of networks and nodes that are available with each class. A sample IP address is also provided for each of the different classes.

**FIGURE 10.3**

Each of the IP classes uses a certain number of octets for network addressing and a certain number of octets for node addressing.

Table 10.1 IP Network Classes

Class	First Octet Range	Number of Networks	Number of Hosts	Sample Address
A	1-126	127	16,777,214	10.15.121.5
B	128-191	16,384	65,534	130.13.44.52
C	192-223	2,097,152	254	200.15.23.8

Binary Equivalents and First Octets

Remember that when you see an IP address such as 200.1.25.7 (and the sample addresses shown in Table 10.1), you are actually looking at a convenient dotted decimal representation of a series of 32 bits that are divided into four 8-bit octets. Each octet consists of 8 bits, which is one byte. So in actuality the IP address 200.1.25.7 is really a series of 32 1s and 0s—11001000 00000001 00011001 00000111.

Class D and Class E addresses

Two additional classes of IP network addresses exist: *Class D* and *Class E*. Class D network addresses are used by multicast groups receiving data on an inter-network from a particular application or server service. An example of a multicast use of Class D addresses is Microsoft NetShow, which can broadcast the same content to a group of users at one time. Class E addresses belong to an experimental class, which isn't available for use by folks like you and me.

Converting decimal to binary or vice versa

You can quickly convert decimal to binary or binary to decimal using the Windows calculator. Start the Calculator (from the **Start** menu; choose **Programs**). Click the **View** menu and select **Scientific**. The default numbering system is decimal type in the decimal number from any octet in an IP address such as 126. Then click the **Bin** (binary) radio button on the format bar of the calculator. The number is converted to binary—1111110. Note that the calculator doesn't place the lead zeros in the binary numbers so to show all 8-bit places for an IP address octet, you would have to add the lead 0 to this number—01111110. To convert binary to decimal, click **Bin** and then type in the 8-bit binary number (1s and 0s) for the octet. Then click **Dec** to convert to decimal.

How the decimal number 200 is converted to the binary number 11001000 or vice versa will be discussed in "Subnetting IP Addresses," later in this chapter. So, for now, you only need to understand that IP addresses are written in the dotted decimal format really as a convenience and they actually exist as a series of 1s and 0s.

Rules have been established for the *leading bits* in the first octet of each of the classes you've discussed (A, B, and C). This enables a router to look at the first octet of an IP address and immediately know which class of IP address it is looking at (it's also a convenient way for you to quickly tell a Class A address from a Class B or C address).

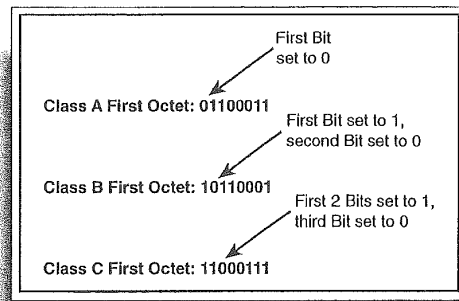
- In Class A addresses the first bit of the first octet is set to 0.
- In Class B addresses the first bit of the first octet is set to 1, and the second bit is set to 0.
- In Class C addresses the first two bits of the first octet are set to 1 and the third bit is set to 0.

Figure 10.4 shows the first octet of a Class A, B, and C IP address in binary format respectively. Converting binary to decimal is a subject that you shall cover in a moment. However, you should take a look at IP subnet masks before you get into the math.

Basic Subnet Masks

Another aspect of IP addressing that is extremely important to how IP addressing works is the use of *subnet masks*. An IP address without the appropriate subnet mask is like Laurel without Hardy (or I guess now that's Beavis without Butthead). The subnet mask for a particular IP address is actually used by the router to resolve which part of the IP address is providing the network address and which part of the address is providing host address.

The basic subnet masks for each class are provided in Table 10.2. Subnet masks also consist of four octets of information. A router matches up the information in the subnet mask with the actual IP address and determines the network address and the node address.

**FIGURE 10.4**

The First Octet rule helps define the different classes of IP addresses.

Table 10.2 Basic Subnet Masks

Class	Subnet Mask
A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

In the basic subnet masks (where no subnetting has been done) the octet either has all the bits turned on (represented by 1s) or all the bits turned off (represented by 0s). When all the bits are turned on (all 8 bits are represented by 1s) the decimal equivalent is 255. When all the bits are set to the binary 0, the decimal equivalent is 0. Figure 10.5 shows the binary equivalent of the Class B basic subnet mask.

The big question is how does a router use the subnet mask to determine which part of an IP address refers to the network address. It actually uses a process called *anding* where it “ands” the bits in the subnet mask with the bits in the IP address to determine the network address.

Here’s how anding works: the IP address and the subnet mask are both viewed by the router in binary format (which you will learn to do in the next section of this chapter). The bits in the subnet mask are then “anded” with the corresponding bits in the IP address. Table 10.3 shows the results of anding binary bits (1s and 0s).

An early bird brain teaser

Even though you haven’t talked about converting binary numbers to decimal yet, take my word for it that the first bit in an octet when represented by the binary 1 has the decimal value of 128. Because Class A addresses always have the first bit set to 0, it isn’t worth anything (meaning it’s 0 in decimal too). So, the first octet value for Class A networks is always less than 128 (take a look at the first octet decimal range for Class A addresses in Table 10.1).

FIGURE 10.5

Subnet masks can also be represented in decimal or binary.

Class B Subnet Mask
 Decimal: 255.255.0.0
 Binary: 11111111 11111111 00000000 00000000

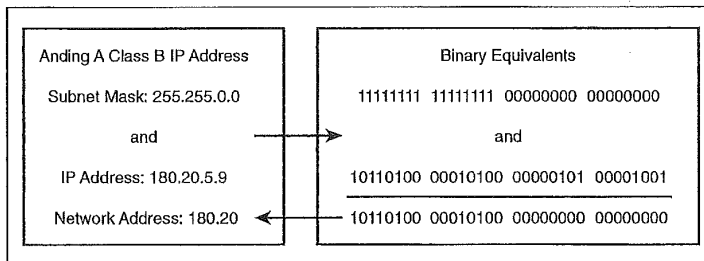
Table 10.3 Anding

Bit Combination	Result
1 and 1	1
1 and 0	0
0 and 0	0

Let's take a look at some actual anding. In Figure 10.6 a Class B IP address and its basic subnet mask are converted to binary. The binary equivalents of the IP address and the subnet mask are anded. The result is the IP address for the network (in this case 180.20.0.0.00).

FIGURE 10.6

The network number is resolved by anding the IP address and the Subnet mask.



Subnetting IP Addresses

Now that you've been introduced to the format for IP addresses and their subnet masks, you can tackle subnetting. Basically, subnetting enables you to take a number of LANs and connect them together into one internetwork. It also provides you with the capability to break a large network into subnets that are connected with routers. Segmenting a large network using routers allows you to maximize

the bandwidth of the network because the routers keep the traffic on each subnet local; the data isn't broadcast to the entire network.

Each of the classes that you discussed in the previous section (Class A, B, and C) can be subnetted. Before you get into the actual math involved in determining subnets and the new network subnet mask, let's look at how dotted decimal IP addresses are converted to decimal and vice versa.

SEE ALSO

➤ To review how routers work, see page 78.

Binary and Decimal Conversions

Each octet in the IP address (although represented as a decimal number) consists of 8 bits. Each bit position has a decimal equivalent. That decimal equivalent isn't realized, however, unless the bit is represented as a 1 (0 bits have no decimal value). Figure 10.7 shows the decimal value of each bit position in the octet and the total value for the octet when certain bits have the binary value of 1.

The first bit in any octet of an IP address will have a decimal value of 128 followed by a bit that has a decimal value of 64. The bits on the far left of the octet are referred to as the *high-order bits*. If you move down to the right end of the octet, where the last bit's decimal value is 1 (followed by a bit on the left that has a decimal value of 2), you are working with the *lower-order bits*.

Decimal Value of Bit Positions								Decimal Total
128	64	32	16	8	4	2	1	
1	0	0	0	0	0	0	0	128
1	1	0	0	0	0	0	0	192
1	1	1	0	0	0	0	0	224
1	1	1	1	0	0	0	0	240
1	1	1	1	1	0	0	0	248
1	1	1	1	1	1	0	0	252
1	1	1	1	1	1	1	0	254
1	1	1	1	1	1	1	1	255

Does the router have common sense?

You're probably looking at Figure 10.7 and thinking, "Why doesn't the router just look for 255 in the subnet mask and then use the numbers in the IP address that appear in the same octet as the Network address. Well, in effect it does, although it must crosscheck the numbers in binary because it receives the data in a bit stream (a stream of 1s and 0s). Also, when you throw in subnets and have subnet masks that include subnetting bits, it isn't quite as obvious which part of the address is network information and which part of the address is providing subnet or node information.

FIGURE 10.7
Decimal equivalent for octet bit combinations.

Figure 10.7 gives you the total decimal value for an octet when you turn on bits (bit values of 1) working from the high-order bits to the low-order bits. Note that when all the bit values are set to 1, the total decimal value is 255.

Obviously, you will run across IP address octets where only lower order bits have decimal values. For example, if the first low-order bit and the second low-order bit are both set to the binary 1, you have an octet decimal value of 3 ($1+2$).

When you do IP subnetting, you work with both the high-order and low-order bits. And although the math involved in the subnetting process has you hopping from one end of the octet to the other (using the low-order bits for some calculations and the high-order bits for others), the process is pretty straightforward.

Just to make a point, let's convert the octet 01110001 to decimal using the information in Figure 10.8. The answer would be $64+32+16+1=113$. All you have to do is add the decimal equivalents of the bits in the octet that are set to 1.

Creating Subnets on a Class A Network

The easiest way to learn subnetting is to actually do it. So let's take a look at a Class A network and walk through the steps of subnetting it (remember that they are only very few Class A networks, but the subnetting math is actually easiest when working with Class A and Class B networks).

The first octet of a Class A network can be in the decimal range of 1-126. So let's say that you've been assigned the network address 10.0.0.0.

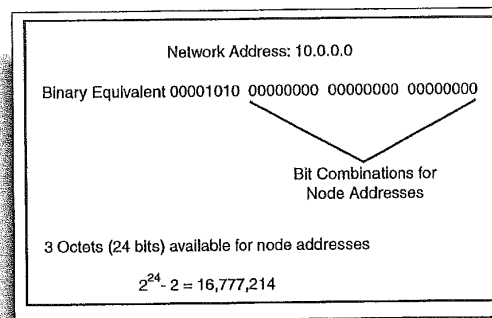
In Class A networks, the first octet defines the network address. The remaining three octets provide the node address information because you have all the possible bit combinations available in 3 octets. That's 24 bit positions, so the number of node addresses available would be $2^{24}-2$ or 16,777,214 (you take 2 to the power of the number of bits that are available to create node addresses—in this case 3 octets or 24 bits).

When to subnet?

Subnetting IP networks is required when you attach remote sites together using routers. Another reason to subnet is when you have a network with a huge number of nodes that is really chewing up your bandwidth. So, while you may never work with a Class A IP network, a network providing that many possible hosts would certainly be subnetted.

The reason that you must subtract 2 from the possible node addresses (2^{24}) is that you lose two possibilities because the bits in the node octets cannot be set to all 1s or 0s. When the node octets are all set to 1, that address is used to broadcast messages to all the nodes on the network—it means all nodes—and so can't be used for an actual node address. When the node octets are all set to 0, that address signifies the network wire address. In our case, if all the node octets are set to 0, you get the address 10.0.0.0, which remember is our network address, which becomes very important when you configure IP networks on a router.

Figure 10.8 summarizes what you've talked about so far. Now you also know that you take the number of bits available for node addresses and can quickly figure out how many possible node addresses that gives you by raising 2 to the power of the node bit total and then subtracting 2.

**FIGURE 10.8**

The number of node addresses available can be determined using the number of bits available for node addressing.

Now, if you're with me so far, you probably feel pretty good, but you've only scratched the surface of this whole subnetting issue. The next step is to determine how many subnets you will need for your network. If you have a Class A network, your operation will probably be spread across a wide geographical area and use both LAN and WAN technology. Let's keep this simple, however. Say you want to divide your large network into 30 subnets (you will also have to have a separate router interface to service each subnet so, even for 30 subnets you are talking about several routers that each have a number of interfaces (such as Ethernet interfaces) to connect to the different subnets).

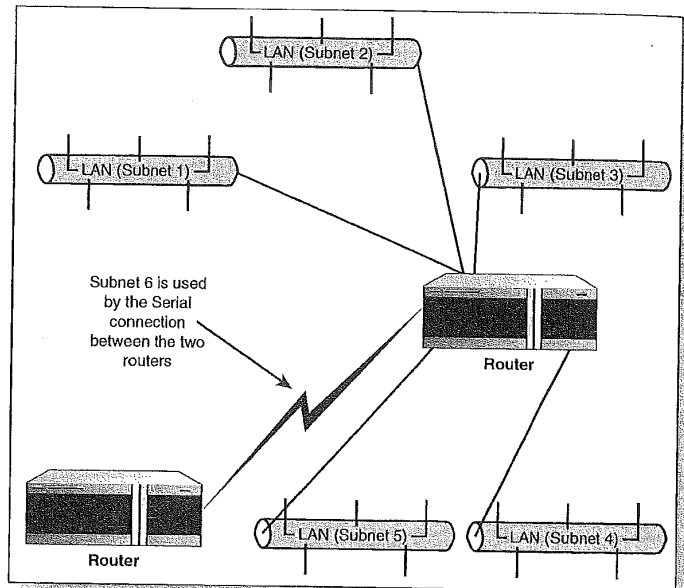
Figure 10.9 shows a portion of the network that you are dividing into 30 subnets. You are looking at just one location (one building) in the larger network, where six subnets out of the total 30 will be used. Each LAN will be its own subnet (meaning the LAN interface on the router connected to the LAN will be part of the same subnet). The serial connection between the two routers also requires that it be a separate subnet, so one of the subnets you create will be used for that set of interfaces (on both the routers).

FIGURE 10.9

A subnetted network will consist of separate LAN subnets and WAN subnets that connect routers.

How many subnets should I create?

When you divide your network into logical subnets, you want to create enough subnets to take care of the current locations on your networks (using both LAN and WAN connections). You also want to keep growth in mind as well. You really only want to do this subnetting math once because IP addresses will be assigned to the various routers and computers on the network. Having to do this all over because you only created six subnets and now need 14 is probably not going to set well with your boss. Oh, well, you can always work in sales.



Now that you know how many subnets you need, you can begin to work through the process of stealing bits to create our subnets. The first thing that you will do is create the new subnet mask that will be used for the entire network.

Creating the Network Subnet Mask

You want 30 subnets. Right now our network address 10.0.0.0 only supplies bits for the network address (the first octet) and bits for node addresses (the other three octets). So, how do you create subnets? You steal some bits from the node octets and use them to create our subnets (you can't steal bits from the network octet because this is provided to you by the people who assign IP networks—it is basically cast in stone).

So, you will steal bits from the first node octet to create our subnets (the second octet in the 10.0.0.0 address—from left to right). This means that the possible number of node addresses is going to be decreased because you are going to take some of the bits to create subnets (with bits removed for subnets, you get less node addresses).

Stealing the bits will not only let us compute ranges of IP addresses for each subnet (each of the 30 subnets will have a different range of IP addresses), but it also lets us create a new subnet mask for the entire network. This new subnet mask will let routers and other devices on the network know that you have divided our network into subnets and it will also tell them how many logical subnets have been created.

But first things first, you must figure out how many bits you need to steal to come up with 30 subnets. Remember that each bit in an octet has a decimal value. For example, the first low-order bit on the far right of the octet has a decimal value of 1, the bit to its left has a value of 2 and so on. So, to create 30 subnets you add the lower order bits' decimal values until you come up with a value of 31. Why 31 and not 30? You cannot use subnet 0, which is what you derive when you steal only the first lower-order bit. So the formula is actually: total decimal value of stolen lower-order bits minus 1. Figure 10.10 shows you how five lower order bits were used to come up with 30 subnets.

When you know how many bits it takes to create 30 subnets—5 bits—you can create the new subnet mask for the entire Class A network. Forget for the moment that you used lower order bits (adding from right to left) to come up with the 30 subnets.

Take the first five high-order bits (128, 64, 32, 16, and 8) working from left to right. Add them together: $128+64+32+16+8=248$. The 248 is very important. Normally, a Class A subnet mask is 255.0.0.0. But this Class A network has been subnetted (using bits in the second octet). So the new subnet mask is 255.248.0.0.

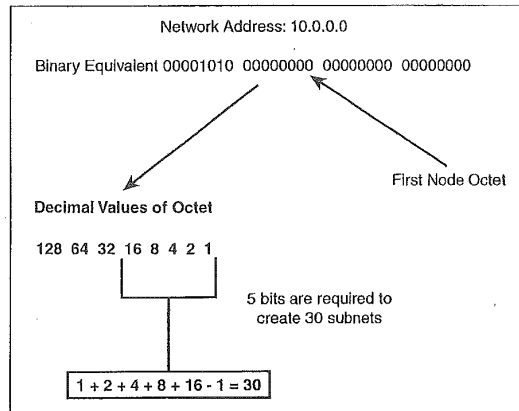
This new subnet mask tells routers and other devices that this Class A network contains 30 subnets. Now that you have the subnet mask for the entire network (this subnet mask would be used as the subnet mask for router interfaces and computers on the network no matter which of the 30 subnets that node is on) you can figure out the range of IP addresses that would be available in each of the 30 subnets.

You may have to create more subnets than you need

When you figure out the number of subnets that you need, you may find that when you start converting lower order bits to decimal and start adding them, you end up with more subnets than you actually want. For example, if you want 26 subnets, you will have to create 30 subnets because the decimal equivalents of the bits themselves. This doesn't mean you have to use them all, you can still set up 26 subnets on your network; it's just that you can't create that exact number.

FIGURE 10.10

Lower order bits are added and then 1 is subtracted to come up with the number of available subnets.



Calculating IP Subnet Ranges

Calculating the subnet ranges is pretty straightforward. You used five high-order bits to determine the binary number used in the second octet of our new subnet mask for the network. These high-order bits also provide the secret for determining the IP address ranges for each subnet. The high-order decimal values that you used for the subnet mask were: 128, 64, 32, 16, and 8.

Take the lowest of the high-order bits that you used to calculate the new subnet mask, in this case 8. This number becomes the increment used to create the IP address ranges for the 30 subnets.

For example, the first subnet (of our 30) will begin with the IP address 10.8.0.1. The 8 is used as the starting increment for the second octet in the IP address. Remember, it was the second octet that you stole the bits from to create our subnets. So, all IP addresses that have a second octet decimal value of less than 8 are invalid values. To calculate the beginning number of our next subnet add 8 to the second octet, you get 16. So, the starting address for the second subnet will be 10.16.0.1. Continue to add 8 to the second octet to determine the start address for all 30 of the subnets.

Now, you probably wonder where I came up with the 0 in the third octet and the 1 in the fourth octet. The possible decimal values of any octet range from 0 (where all bits are set to 0) to 255 (where all bits are set to 1). So the first IP address in the subnet can have all 0s in the third octet. So, why does the fourth position start with 1? Remember, I said earlier that the node address could not be represented by octets containing all 0s or all 1s. If the fourth octet was 0, both the node octets (the third and the fourth) would be all 0s, which is used to denote the subnetwork address, and so it isn't a legal address for a node.

To determine the range of addresses for a particular subnet, you take that subnet's starting address and use all the addresses that are between it and the starting address of the next subnet. For example, the first subnet will contain all the addresses between 10.8.0.1 and 10.16.0.1 (but not including 10.16.0.1).

Table 10.4 gives the start and end address for the first 10 of the 30 subnets that you created. To figure out the other 20 ranges, simply add the increment (8) to the second octet (the subnet octet).

Table 10.4 IP Address Ranges for Subnets (First 10 of 30)

Subnet #	Start Address	End Address
1	10.8.0.1	10.15.255.254
2	10.16.0.1	10.23.255.254
3	10.24.0.1	10.31.255.254
4	10.32.0.1	10.39.255.254
5	10.40.0.1	10.47.255.254
6	10.48.0.1	10.55.255.254
7	10.56.0.1	10.63.255.254
8	10.64.0.1	10.71.255.254
9	10.72.0.1	10.79.255.254
10	10.80.0.1	10.87.255.254

Why does the end address for each subnet stop at 254?

Remember that the node portion of the IP address (in this case the third and fourth octet) cannot be all 1s (or 255 in decimal format). So, you can have all 1s in the third octet (255), but can only go to 254 in the fourth octet.

How many IP addresses do you lose when subnetting?

Be advised that subnetting (stealing bits for subnets) reduces the number of IP addresses available for your network nodes. For example, a Class A network that isn't subnetted provides 16,777,214 node addresses. Now, you computed that if you create 30 subnets on a class A network you get 524,286 IP addresses per subnet. Multiply 524,286 by 30. You get 15,728,580. So, 16,777,214 minus 15,728,580 is 1,048,634. You lose a lot of potential node addresses by subnetting.

Calculating Available Node Addresses

I've already stressed the importance of creating the appropriate number of IP subnets for your network (with growth figured in). But you also need to make sure that the number of node addresses available for each subnet will accommodate the number of computers and other devices that you plan to deploy on the subnets. Each subnet is a mini-network unto itself and you can't steal IP addresses from one of the other subnets, if you find that you don't have enough addresses for all your devices.

Calculating the number of node addresses available in each subnet is very straightforward. In our Class A network, you originally had 24 bits dedicated to node addressing. To create the 30 subnets, you had to steal 5 bits from the second octet. This means that now only 19 bits (24-5) are available to create node IP addresses. To calculate the nodes addresses per subnet, take 2 and raise it to the 19th power and then subtract 2 ($2^{19} - 2$). This results in 524,286 IP addresses per subnet. Obviously, Class A networks provide a huge number of addresses and coming up short is pretty improbable. But when you work with the subnetting of Class B and Class C addresses, you need to make special note of how many addresses you have available in each subnet.

Creating Class B and Class C Subnets

The process of creating Class B and Class C subnets is very similar to creating Class A subnets. The math is all the same, however, you are working with a smaller pool of potential node addresses when you subnet. Let's look at each of these classes briefly.

Class B Subnetting

Class B networks that aren't subnetted provide 2 octets (16 bits) for node addressing. This provides 65,534 node addresses. The basic subnet mask for a Class B network is 255.255.0.0.