Chapter 4: Routing Concepts

Objectives

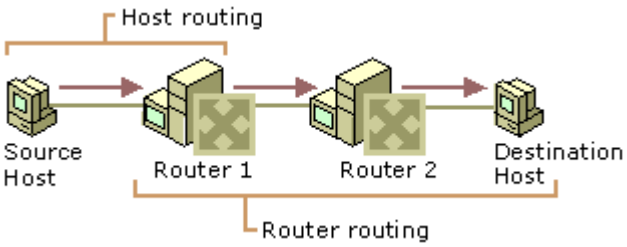
- What are the primary functions and features of a router?
- How do you connect devices for a small routed network?
- Basic configuration settings on a router to route between two directly connected networks?
- Connectivity verification between two networks that are directly connected to a router?
- How do routers encapsulate and de-encapsulate packets when switching packets between directly connected interfaces?
- How routers determine the best path?
- How routers build a routing table of directly connected networks?
- How do routers build a routing table using a static and dynamic routing protocol?

1. Routing Concepts

Routing is the process of transferring data across an internetwork from a source host to a destination host. Routing can be understood in terms of two processes: host routing and router routing.

Host routing occurs when the sending host forwards a packet. Based on the destination network address, the sending host must decide whether to forward the packet to the destination or to a router. In the Figure , the Source Host forwards the packet destined for the Destination Host to Router 1.

Router routing occurs when a router receives a packet that is to be forwarded. The packet is forwarded between routers (when the destination network is not directly attached to the router) or between a router and the destination host (when the destination network is directly attached). In the Figure, Router 1 forwards the packet to Router 2. Router 2 forwards the packet to the Destination Host.



- ✓ Networks allow people to communicate, collaborate, and interact in many ways. Networks are used to access web pages, talk using IP telephones, participate in video conferences, compete in interactive gaming, shop using the Internet, complete online coursework, and more.
- ✓ At the core of the network is the router. A router connects one network to another network. The router is responsible for the delivery of packets across different networks. The destination of the IP packet might be a web server in another country or an email server on the local-area network.
- ✓ The router uses its routing table to determine the best path to use to forward a packet. It is the responsibility of the routers to deliver those packets in a timely manner. The effectiveness of internetwork communications depends, to a large degree, on the ability of routers to forward packets in the most efficient way possible.
- ✓ When a host sends a packet to a device on a different IP network, the packet is forwarded to the default gateway because a host device cannot communicate directly with devices outside of the local network. The ***default gateway*** is the destination that routes traffic from the local network to devices on remote networks. It is often used to connect a local network to the Internet.
- ✓ This chapter will also answer the question, "What does a router do with a packet received from one network and destined for another network?" Details of the routing table will be examined, including connected, static, and dynamic routes.
- ✓ Because the router can route packets between networks, devices on different networks can communicate. This chapter will introduce the router, its role in the networks, its main hardware and software components, and the routing process.

2. Initial Configuration of a Router

A router is essentially a special-purpose computer with an internetwork operating system optimized for the purpose of routing and securing networks. This section will examine the functions of a router and how a router determines the best path. It will also review the command-line interface (CLI) commands required to configure the base settings of a router.

Characteristics of a Network

Networks have had a significant impact on our lives. They have changed the way we live, work, and play.

Networks allow us to communicate, collaborate, and interact in ways we never did before. We use the network in a variety of ways, including web applications, IP telephony, video conferencing, interactive gaming, electronic commerce, education, and more.

There are many terms, key structures, and performance-related characteristics that are referred to when discussing networks. These include:

- **Topology:** There are physical and logical topologies. The ***physical topology*** is the arrangement of the cables, network devices, and end systems. It describes how the network devices are actually interconnected with wires and cables. The ***logical topology*** is the path over which the data is transferred in a network. It describes how the network devices appear connected to network users.

- **Speed**: Speed is a measure of the data rate in bits per second (b/s) of a given link in the network.
- **Cost**: Cost indicates the general expense for purchasing of network components, and installation and maintenance of the network.
- **Security**: Security indicates how protected the network is, including the information that is transmitted over the network. The subject of security is important, and techniques and practices are constantly evolving. Consider security whenever actions are taken that affect the network.
- ***Availability***: Availability is a measure of the probability that the network is available for use when it is required.
- ***Scalability***: Scalability indicates how easily the network can accommodate more users and data transmission requirements. If a network design is optimized to only meet current requirements, it can be very difficult and expensive to meet new needs when the network grows.
- ***Reliability***: Reliability indicates the dependability of the components that make up the network, such as the routers, switches, PCs, and servers. Reliability is often measured as a probability of failure or as the mean time between failures (MTBF).

These characteristics and attributes provide a means to compare different networking solutions.

While the term “speed” is commonly used when referring to the network bandwidth, it is not technically accurate. The actual speed that the bits are transmitted does not vary over the same medium. The difference in bandwidth is due to the number of bits transmitted per second, not how fast they travel over wire or wireless medium.

Why Routing?

How does clicking a link in a web browser return the desired information in mere seconds? Although there are many devices and technologies collaboratively working together to enable this, the primary device is the router. Stated simply, a router connects one network to another network.

Communication between networks would not be possible without a router determining the best path to the destination and forwarding traffic to the next router along that path. The router is responsible for the routing of traffic between networks.

When a packet arrives on a router interface, the router uses its routing table to determine how to reach the destination network. The destination of the IP packet might be a web server in another country or an email

server on the local-area network. It is the responsibility of routers to deliver those packets efficiently. The effectiveness of internetwork communications depends, to a large degree, on the ability of routers to forward packets in the most efficient way possible.

Routers As Computers

Most network capable devices (i.e., computers, tablets, and smartphones) require the following components to operate:

- Central processing unit (CPU)
- Operating system (OS)
- Memory and storage (RAM, ROM, NVRAM, Flash, hard drive)

A router is essentially a specialized computer. It requires a CPU and memory to temporarily and permanently store data to execute operating system instructions, such as system initialization, routing functions, and switching functions.

Cisco devices use the Cisco Internetwork Operating System (IOS) as the system software.

Routers store data using:

- ***Random Access Memory (RAM)***: Provides temporary storage for various applications and processes, including the running IOS, the running configuration file, various tables (i.e., IP routing table, Ethernet ARP table), and buffers for packet processing. RAM is referred to as volatile because it loses its contents when power is turned off.
- ***Read-Only Memory (ROM)***: Provides permanent storage for bootup instructions, basic diagnostic software, and a limited IOS in case the router cannot load the full featured IOS. ROM is firmware and referred to as non-volatile because it does not lose its contents when power is turned off.
- ***Non-Volatile Random Access Memory (NVRAM)***: Provides permanent storage for the startup configuration file (startup-config). NVRAM is non-volatile and does not lose its contents when power is turned off.
- ***Flash***: Provides permanent storage for the IOS and other system-related files. The IOS is copied from flash into RAM during the bootup process. Flash is non-volatile and does not lose its contents when power is turned off. Contents of flash may be overwritten.

Table 4-1 provides a summary of the types of router memory, their volatility, and examples of what is stored in each.

Table 4-1 Router Memory

Memory	Volatile/Non-Volatile	Stores
RAM	Volatile	• Running IOS • Running configuration file • IP routing and ARP tables • Packet buffer
ROM	Non-volatile	• Bootup instructions • Basic diagnostic software • Limited IOS
NVRAM	Non-volatile	• Startup configuration file
Flash	Non-volatile	• IOS file • Other system files

Unlike a computer, a router does not have video adapters or sound card adapters. Instead, routers have specialized ports and network interface cards to interconnect devices to other networks. Figure 4-1 displays the back panel of a Cisco 1941 ISRG2 and identifies those special ports and interfaces.

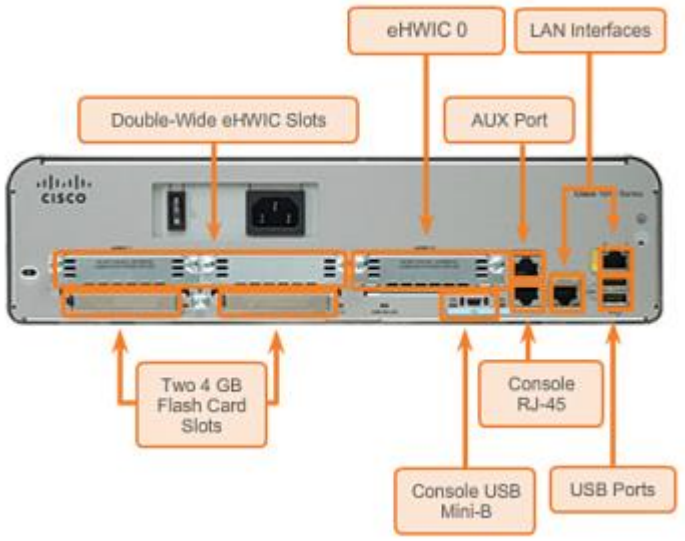


Figure 4-1 Back Panel of a 1941 ISRG2

Routers Interconnect Networks

Most users are unaware of the presence of numerous routers on their own network or on the Internet. Users expect to be able to access web pages, send emails, and download music, regardless of whether the server accessed is on their own network or on another network. Networking professionals know that it is the router that is responsible for forwarding packets from network to network, from the original source to the final destination.

A router connects multiple networks, which means routers support a variety of interface types unlike switches typically supports Ethernet interfaces. When a router receives an IP packet on one interface, it determines which interface to use to forward the packet to the destination. The interface that the router uses to forward the packet may be the final destination, or it may be a network connected to another router that is used to reach the destination network.

Each network that a router connects to typically requires a separate interface. These interfaces are used to connect a combination of both local-area networks (LANs) and wide-area networks (WANs). LANs are commonly Ethernet networks that contain devices, such as PCs, printers, and servers. WANs are used to connect networks over a large geographical area. For example, a WAN connection is commonly used to connect a LAN to the Internet service provider (ISP) network.

Notice that each site in Figure 4-2 requires the use of a router to interconnect to other sites. Even the Home Office requires a router. In this topology, the router located at the Home Office is a specialized device that performs multiple services for the home network.

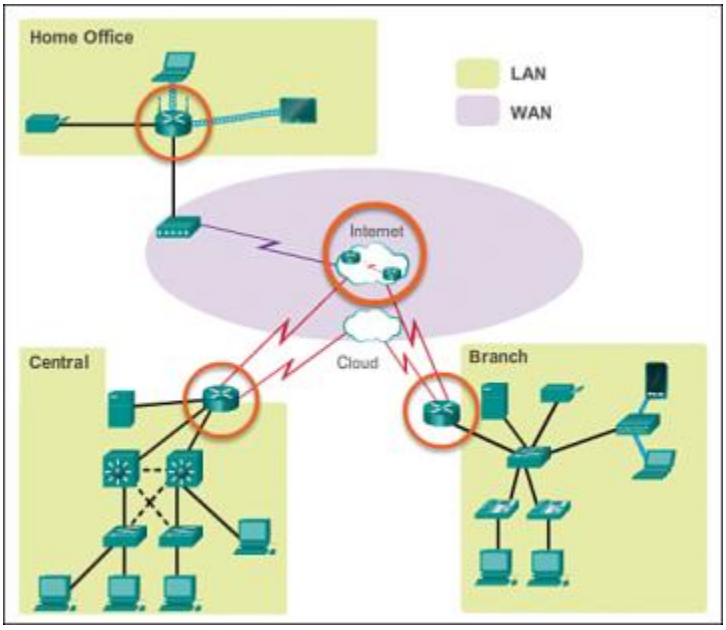


Figure 4-2 Sample Routed Topology

Routers Choose Best Paths

The primary functions of a router are to:

- Path selection

- Packet Switching

The router uses its routing table to determine the best path to use to forward a packet. When the router receives a packet, it examines the destination address of the packet and uses the routing table to search for the best path to that network. The routing table also includes the interface to be used to forward packets for each known network. When a match is found, the router encapsulates the packet into the data link frame of the outgoing or exit interface, and the packet is forwarded toward its destination.

It is possible for a router to receive a packet that is encapsulated in one type of data link frame, and to forward the packet out of an interface that uses a different type of data link frame. For example, a router may receive a packet on an Ethernet interface, but must forward the packet out of an interface configured with the Point-to-Point Protocol (PPP). The data link encapsulation depends on the type of interface on the router and the type of medium to which it connects. The different data link technologies that a router can connect to include Ethernet, PPP, Frame Relay, DSL, cable, and wireless (802.11, Bluetooth).

Routers use static routes and dynamic routing protocols to learn about remote networks and build their routing tables.

Packet Forwarding Mechanisms

Routers support three packet-forwarding mechanisms:

- ***Process switching***: An older packet-forwarding mechanism still available for Cisco routers. When a packet arrives on an interface, it is forwarded to the control plane, where the CPU matches the destination address with an entry in its routing table, and then determines the exit interface and forwards the packet. It is important to understand that the router does this for every packet, even if the destination is the same for a stream of packets. This process-switching mechanism is very slow and rarely implemented in modern networks. Figure 4-3 illustrates how packets are process-switched.

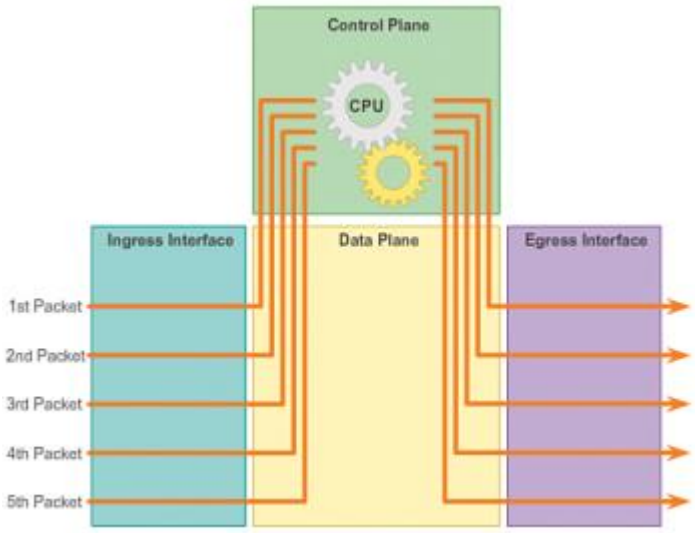


Figure 4-3 Process Switching

- ***Fast switching***: This is a common packet-forwarding mechanism which uses a fast-switching cache to store next-hop information. When a packet arrives on an interface, it is forwarded to the control plane, where the CPU searches for a match in the fast-switching cache. If it is not there, it is process-switched and forwarded to the exit interface. The flow information for the packet is also stored in the fast-switching cache. If another packet going to the same destination arrives on an interface, the next-hop information in the cache is re-used without CPU intervention. Figure 4-4 illustrates how packets are fast-switched.

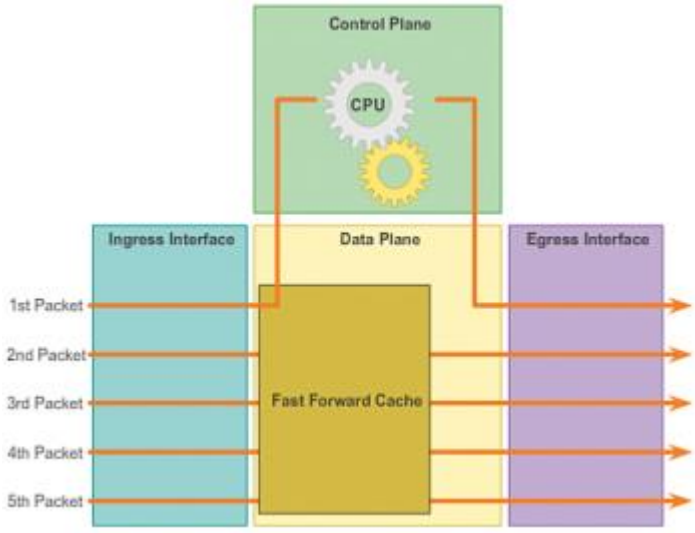


Figure 4-4 Fast Switching

- ***Cisco Express Forwarding (CEF)***: CEF is the most recent and preferred Cisco IOS packet-forwarding mechanism. Like fast switching, CEF builds a Forwarding Information Base (FIB) and an adjacency table. However, the table entries are not packet-triggered like fast switching but change-triggered such as when something changes in the network topology. Therefore, when a network has converged, the FIB and adjacency tables contain all the information a router would have to consider when forwarding a packet. The FIB contains pre-computed reverse lookups and next-hop information for routes, including the interface and Layer 2 information. Cisco Express Forwarding is the fastest forwarding mechanism and the preferred choice on Cisco routers. Figure 4-5 illustrates how packets are forwarded using CEF.

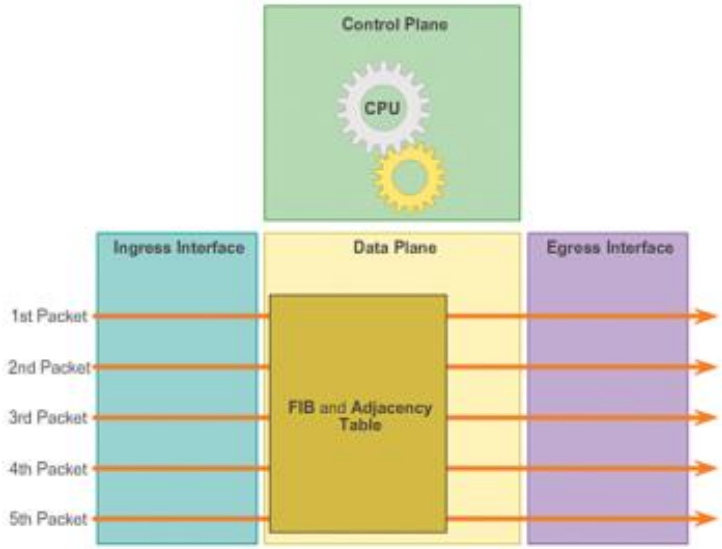


Figure 4-5 Cisco Express Forwarding

Figures 4-3 to 4-5 illustrate the differences between the three packet-forwarding mechanisms. Assume a traffic flow consisting of five packets all going to the same destination. As shown in Figure 4-3, with process switching, each packet must be processed by the CPU individually. Contrast this with fast switching, as shown in Figure 4-4. With fast switching, notice how only the first packet of a flow is process-switched and added to the fast-switching cache. The next four packets are quickly processed based on the information in the fast-switching cache. Finally, in Figure 4-5, CEF builds the FIB and adjacency tables, after the network has converged. All five packets are quickly processed in the data plane.

A common analogy used to describe the three packet-forwarding mechanisms is as follows:

- Process switching solves a problem by doing math long hand, even if it is the identical problem.
- Fast switching solves a problem by doing math long hand one time and remembering the answer for subsequent identical problems.
- CEF solves every possible problem ahead of time in a spreadsheet.

Connect to a Network

Network devices and end users typically connect to a network using a wired Ethernet or wireless connection. Refer to the sample reference topology in Figure 4-6. The LANs in the figure serve as an example of how users and network devices could connect to networks.

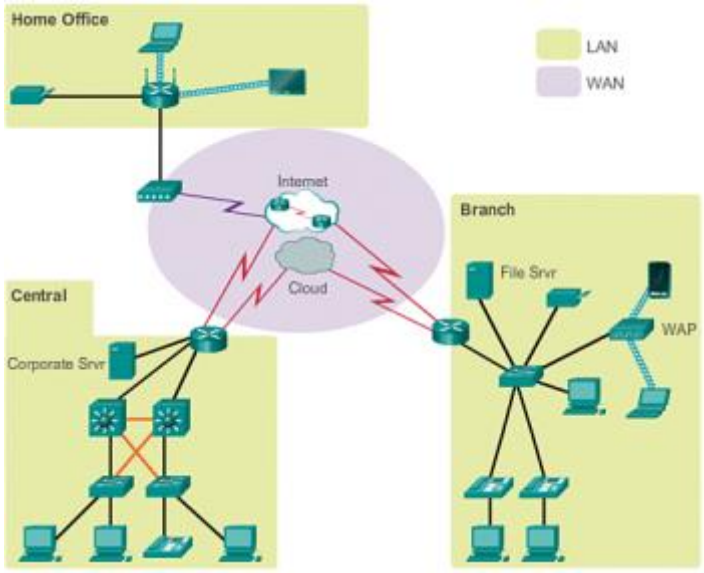


Figure 4-6 Sample LAN and WAN Connections

Home Office devices can connect as follows:

- Laptops and tablets connect wirelessly to a home router.
- A network printer connects using an Ethernet cable to the switch port on the home router.
- The home router connects to the service provider cable modem using an Ethernet cable.
- The cable modem connects to the Internet service provider (ISP) network.

The Branch site devices connect as follows:

- Corporate resources (i.e., file servers and printers) connect to Layer 2 switches using Ethernet cables.
- Desktop PCs and voice over IP (VoIP) phones connect to Layer 2 switches using Ethernet cables.
- Laptops and smartphones connect wirelessly to wireless access points (WAPs).
- The WAPs connect to switches using Ethernet cables.
- Layer 2 switches connect to an Ethernet interface on the edge router using Ethernet cables. An edge router is a device that sits at the edge or boundary of a network and routes between that network and another, such as between a LAN and a WAN.
- The edge router connects to a WAN service provider (SP).
- The edge router also connects to an ISP for backup purposes.

The Central site devices connect as follows:

- Desktop PCs and VoIP phones connect to Layer 2 switches using Ethernet cables.
- Layer 2 switches connect redundantly to multilayer Layer 3 switches using Ethernet fiber-optic cables (orange connections).
- Layer 3 multilayer switches connect to an Ethernet interface on the edge router using Ethernet cables.
- The corporate website server is connected using an Ethernet cable to the edge router interface.
- The edge router connects to a WAN SP.
- The edge router also connects to an ISP for backup purposes.

In the Branch and Central LANs, hosts are connected either directly or indirectly (via WAPs) to the network infrastructure using a Layer 2 switch.

Default Gateways

To enable network access, devices must be configured with IP address information to identify the appropriate:

- ***IP address***: Identifies a unique host on a local network
- ***Subnet mask***: Identifies with which network subnet the host can communicate
- **Default gateway**: Identifies the router to send a packet to when the destination is not on the same local network subnet

When a host sends a packet to a device that is on the same IP network, the packet is simply forwarded out of the host interface to the destination device.

When a host sends a packet to a device on a different IP network, then the packet is forwarded to the default gateway, because a host device cannot communicate directly with devices outside of the local network. The default gateway is the destination that routes traffic from the local network to devices on remote networks. It is often used to connect a local network to the Internet.

The default gateway is usually the address of the interface on the router connected to the local network. The router maintains routing table entries of all connected networks as well as entries of remote networks, and determines the best path to reach those destinations.

For example, if PC1 sends a packet to the Web Server located at 172.16.1.99, it would discover that the Web Server is not on the local network and it, therefore, must send the packet to the Media Access Control (MAC) address of its default gateway. The packet protocol data unit (PDU) in Figure 4-7 identifies the source and destination IP and MAC addresses.

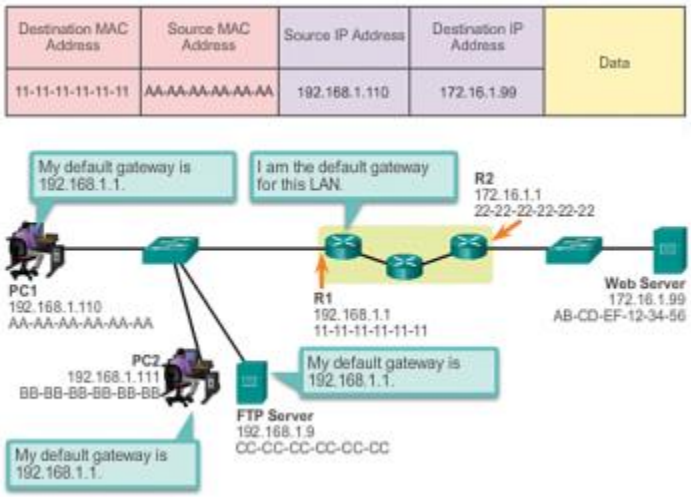


Figure 4-7 Getting the Pieces to the Correct Network

A router is also usually configured with its own default gateway. This is sometimes known as the Gateway of Last Resort.

Document Network Addressing

When designing a new network or mapping an existing network, document the network. At a minimum, the documentation should identify:

- Device names
- Interfaces used in the design
- IP addresses and subnet masks
- Default gateway addresses

This information is captured by creating two useful network documents:

- ***Topology diagram***: Provides a visual reference that indicates the physical connectivity and logical Layer 3 addressing. Often created using software, such as Microsoft Visio.
- ***Addressing table***: A table that captures device names, interfaces, IPv4 addresses, subnet masks, and default gateway addresses.

Figure 4-8 displays the sample topology diagram, while Table 1-2 provides a sample addressing table for the topology.

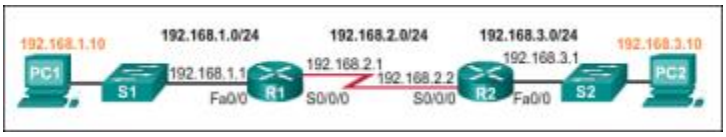


Figure 4-8 Documenting Network Addressing

Table 1-2 Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	Fa0/0	192.168.1.1	255.255.255.0	N/A
	S0/0/0	192.168.2.1	255.255.255.0	N/A
R2	Fa0/0	192.168.3.1	255.255.255.0	N/A
	S0/0/0	192.168.2.1	255.255.255.0	N/A
PC1	N/A	192.168.1.10	255.255.255.0	192.168.1.1
PC2	N/A	192.168.1.10	255.255.255.0	192.168.3.1

Enable IP on a Host

A host can be assigned its IP address information in one of two ways. A host can get a:

- ***Statically Assigned IP Address***: The host is manually assigned the correct IP address, subnet mask, and default gateway. The DNS server IP address can also be configured.
- ***Dynamically Assigned IP Address***: IP address information is provided by a server using the Dynamic Host Configuration Protocol (DHCP). The DHCP server provides a valid IP address, subnet mask, and default gateway for end devices. Other information may be provided by the server.

Figures 4-9 and 4-10 provide static and dynamic IPv4 address configuration examples.

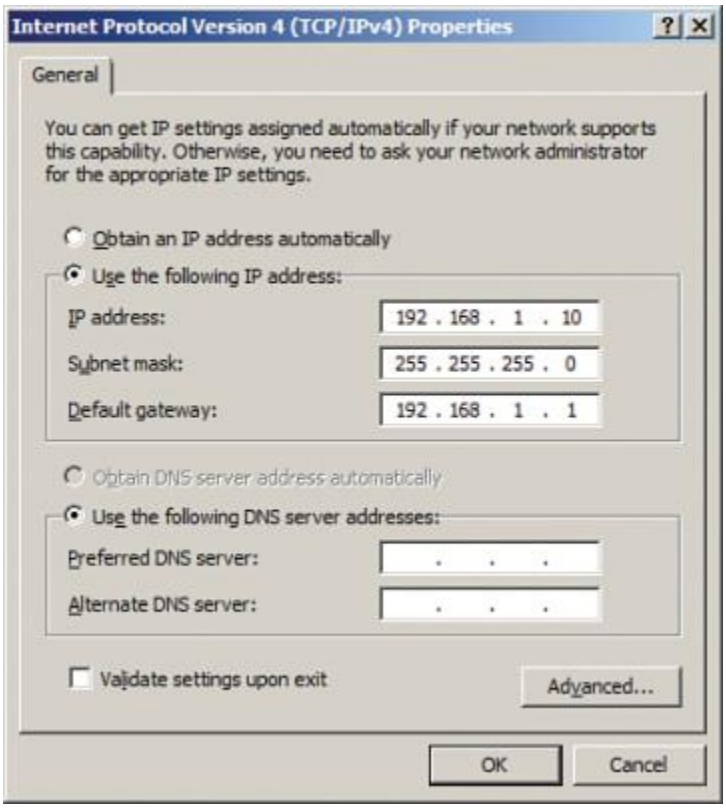


Figure 4-9 Statically Assigning an IP Address

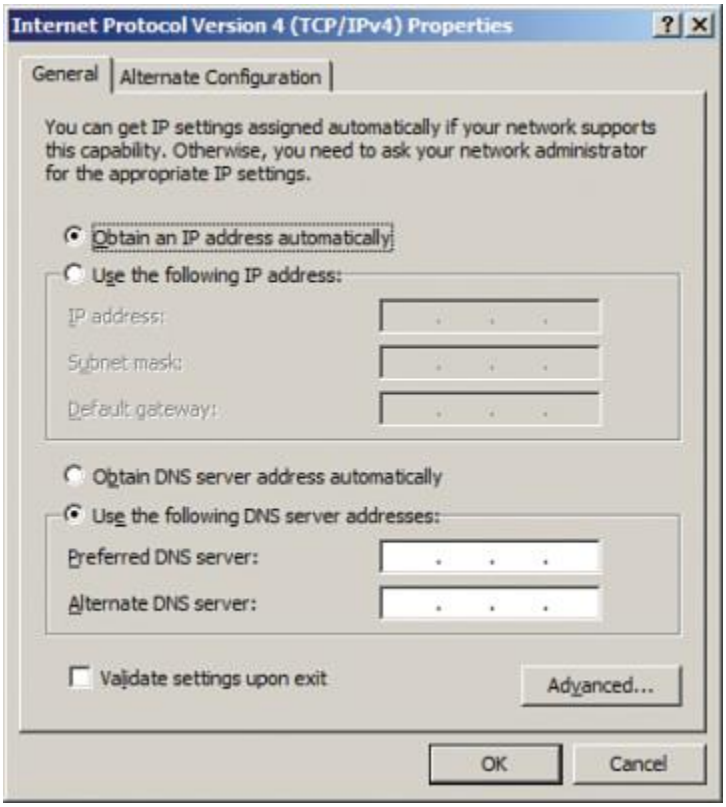


Figure 4-10 Dynamically Assigning an IP Address

Statically assigned addresses are commonly used to identify specific network resources, such as network servers and printers. They can also be used in smaller networks with few hosts. However, most host devices acquire their IPv4 address information by accessing a DHCP server. In large enterprises, dedicated DHCP servers providing services to many LANs are implemented. In a smaller branch or small office setting, DHCP services can be provided by a Cisco Catalyst switch or a Cisco ISR.

Device LEDs

Host computers connect to a wired network using a network interface and RJ-45 Ethernet cable. Most network interfaces have one or two LED link indicators next to the interface. Typically, a green LED means a good connection while a blinking green LED indicates network activity.

If the link light is not on, then there may be a problem with either the network cable or the network itself. The switch port where the connection terminates would also have an LED indicator lit. If one or both ends are not lit, try a different network cable.

The actual function of the LEDs varies between computer manufacturers.

Similarly, network infrastructure devices commonly use multiple LED indicators to provide a quick status view. For example, a Cisco Catalyst 2960 switch has several status LEDs to help monitor system activity and performance. These LEDs are generally lit green when the switch is functioning normally and lit amber when there is a malfunction.

Cisco ISRs use various LED indicators to provide status information. The LEDs on the router help the network administrator conduct some basic troubleshooting. Each device has a unique set of LEDs. Consult the device-specific documentation for an accurate description of the LEDs.

The LEDs of the Cisco 1941 router shown in Figure 4-11 are explained in Table 1-3.

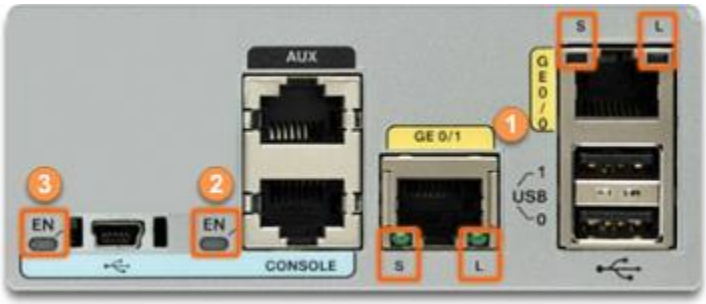


Figure 4-11 Cisco 1941 LEDs

Table 1-3 Description of the Cisco 1941 LEDs

# Port	LED	Color	Description
1 GE0/0 and GE0/1	S (Speed)	1 blink + pause	Port operating at 10 Mb/s
		2 blink + pause	Port operating at 100 Mb/s
		3 blink + pause	Port operating at 1000 Mb/s
	L (Link)	Green	Link is active
		Off	Link is inactive
2 Console	EN	Green	Port is active
		Off	Port is inactive
3 USB	EN	Green	Port is active

# Port	LED	Color	Description
		Off	Port is inactive

Console Access

In a production environment, infrastructure devices are commonly accessed remotely using Secure Shell (SSH) or HyperText Transfer Protocol Secure (HTTPS). Console access is really only required when initially configuring a device, or if remote access fails.

Console access requires:

- ***Console cable***: RJ-45-to-DB-9 console cable
- ***Terminal emulation software***: Tera Term, PuTTY, HyperTerminal

The cable is connected between the serial port of the host and the console port on the device. Most computers and notebooks no longer include built-in serial ports. If the host does not have a serial port, the USB port can be used to establish a console connection. A special USB-to-RS-232 compatible serial port adapter is required when using the USB port.

The Cisco ISR G2 supports a USB serial console connection. To establish connectivity, a USB Type-A to USB Type-B (mini-B USB) is required, as well as an operating system device driver. This device driver is available from http://www.cisco.com. Although these routers have two console ports, only one console port can be active at a time. When a cable is plugged into the USB console port, the RJ-45 port becomes inactive. When the USB cable is removed from the USB port, the RJ-45 port becomes active.

Table 1-4 summarizes the console connection requirements, while Figure 4-12 displays the various ports and cables required.

Table 1-4 Console Connection Requirements

Port on Computer	Cable Required	Port on ISR	Terminal Emulation
Serial Port	RJ-45 to DB-9 Console Cable USB to RS-232 compatible serial port adapter	RJ-45 Console Port	Tera Term PuTTY
USB Type-A Port	• Adapter may require a software driver RJ-45 to DB-9 Console Cable USB Type-A to USB Type-B (Mini-B USB)	USB Type-B(Mini-B USB)	

Port on Computer	Cable Required	Port on ISR	Terminal Emulation
	• A device driver is required and available from Cisco.com		

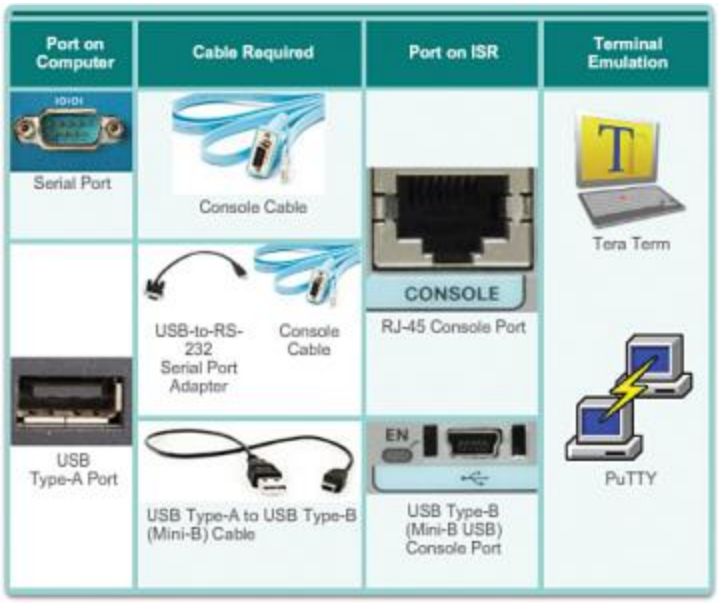


Figure 4-12 Ports and Cables

Enable IP on a Switch

Network infrastructure devices require IP addresses to enable remote management. Using the device IP address, the network administrator can remotely connect to the device using Telnet, SSH, HTTP, or HTTPS.

A switch does not have a dedicated interface to which an IP address can be assigned. Instead, the IP address information is configured on a virtual interface called a ***switched virtual interface (SVI).***

The steps to configure the basic settings on a switch are as follows:

Step 1. Name the device.

Step 2. Configure the SVI. This makes the switch accessible for network management.

Step 3. Enable the SVI.

Step 4. Configure the default gateway for the switch. Packets generated by the switch and destined for an address other than its management network segment will be forwarded to this address. This default gateway is used by the switch only for the packets it generates, not any hosts connected to the switch.

For example, the following commands would configure the management VLAN interface and default gateway of switch S1 shown in Figure 4-13.

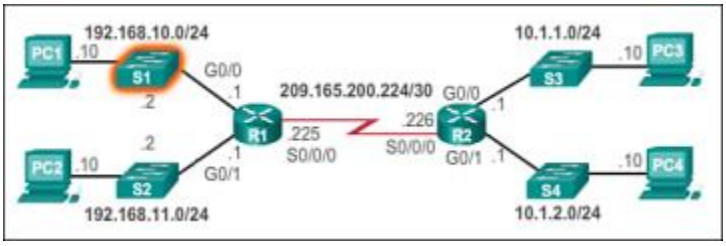


Figure 4-13 Configuring the SVI of S1

```
S1(config)# interface vlan 1
S1(config-if)# ip address 192.168.10.2 255.255.255.0
S1(config-if)# no shutdown
%LINK-5-CHANGED: Interface Vlan1, changed state to up
S1(config-if)# exit
S1(config)#
S1(config)# ip default-gateway 192.168.10.1
S1(config)#
```

In the example, the switch SVI is configured and enabled with the IP address 192.168.10.2/24 and a default gateway of the router located at 192.168.10.1. Packets generated by the switch and destined for an address outside of the 192.168.1.0/24 network segment will be forwarded to this address. In the example, the address is that of the G0/0 interface of R1.

3. Basic Settings on a Router

The basic addressing and configuration of Cisco devices was covered in either the Introduction to Networks or Network Basics course. However, we will spend some time reviewing these topics as well as preparing you for the hands-on lab experience in this course.

Configure Basic Router Settings

Cisco routers and Cisco switches have many similarities. They support a similar modal operating system, similar command structures, and many of the same commands. In addition, both devices have similar initial configuration steps.

When initially configuring a Cisco switch or router, the following steps should be executed:

- **Step 1.** Name the device. This changes the router prompt and helps distinguish the device from others.
- **Step 2.** Secure management access. Specifically, secure the privileged EXEC, user EXEC, and Telnet access, and encrypt passwords to their highest level.
- **Step 3.** Configure a banner. Although optional, this is a recommended step to provide legal notice to anyone attempting to access the device.
- **Step 4.** Save the configuration.

For example, the following commands would configure the basic settings for router R1 shown in Figure 4-14.

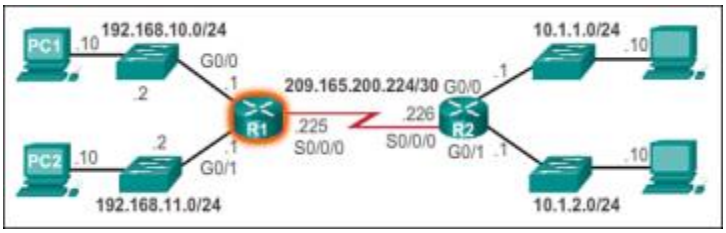


Figure 4-14 Configuring the Basic Settings of R1

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# hostname R1
R1(config)#
R1(config)# enable secret class
R1(config)#
R1(config)# line console 0
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)#
R1(config)# line vty 0 4
R1(config-line)# password cisco
R1(config-line)# login
R1(config-line)# exit
R1(config)#
R1(config)# service password-encryption
R1(config)#
R1(config)# banner motd $ Authorized Access Only! $
R1(config)# end
R1#
R1# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

The use of password protection to control or restrict access to the command line interface (CLI) of your router is one of the fundamental elements of an overall security plan.

Protecting the router from unauthorized remote access, typically Telnet, is the most common security that needs configuring, but protecting the router from unauthorized local access cannot be overlooked.

Note: Password protection is just one of the many steps you should use in an effective in-depth network security regimen. Firewalls, access-lists, and control of physical access to the equipment are other elements that must be considered when implementing your security plan.

Command line, or EXEC, access to a router can be made in a number of ways, but in all cases the inbound connection to the router is made on a TTY line. There are four main types of TTY lines, as seen in this sample **show line** output:

```
2509#show line
   Tty Typ     Tx/Rx    A Modem  Roty AccO AccI   Uses   Noise  Overruns   Int
*    0 CTY               -    -      -    -    -      0       0     0/0       -
     1 TTY   9600/9600   -    -      -    -    -      0       0     0/0       -
     2 TTY   9600/9600   -    -      -    -    -      0       0     0/0       -
     3 TTY   9600/9600   -    -      -    -    -      0       0     0/0       -
     4 TTY   9600/9600   -    -      -    -    -      0       0     0/0       -
     5 TTY   9600/9600   -    -      -    -    -      0       0     0/0       -
     6 TTY   9600/9600   -    -      -    -    -      0       0     0/0       -
     7 TTY   9600/9600   -    -      -    -    -      0       0     0/0       -
     8 TTY   9600/9600   -    -      -    -    -      0       0     0/0       -
     9 AUX   9600/9600   -    -      -    -    -      0       0     0/0       -
    10 VTY               -    -      -    -    -      0       0     0/0       -
    11 VTY               -    -      -    -    -      0       0     0/0       -
    12 VTY               -    -      -    -    -      0       0     0/0       -
    13 VTY               -    -      -    -    -      0       0     0/0       -
    14 VTY               -    -      -    -    -      0       0     0/0       -

2509#
```

The **CTY** line-type is the Console Port. On any router, it appears in the router configuration as **line con 0** and in the output of the **show line** command as **cty**. The console port is mainly used for local system access using a console terminal.

The **TTY** lines are asynchronous lines used for inbound or outbound modem and terminal connections and can be seen in a router or access server configuration as **line x**. The specific line numbers are a function of the hardware built into or installed on the router or access server.

The **AUX** line is the Auxiliary port, seen in the configuration as **line aux 0**.

The **VTY** lines are the Virtual Terminal lines of the router, used solely to control inbound Telnet connections. They are virtual, in the sense that they are a function of software - there is no hardware associated with them. They appear in the configuration as **line vty 0 4**.

Each of these types of lines can be configured with password protection. Lines can be configured to use one password for all users, or for user-specific passwords. User-specific passwords can be configured locally on the router, or you can use an authentication server to provide authentication.

There is no prohibition against configuring different lines with different types of password protection. It is, in fact, common to see routers with a single password for the console and user-specific passwords for other inbound connections.

Below is an example of router output from the **show running-config** command:

```
2509#show running-config
Building configuration...

Current configuration : 655 bytes
!
version 12.2
.
.
.

!--- Configuration edited for brevity


line con 0
line 1 8
line aux 0
line vty 0 4
!
end
```

Configure Passwords on the Line

To specify a password on a line, use the **password** command in line configuration mode. To enable password checking at login, use the **login** command in line configuration mode.

Note: To find additional information on the commands used in this document, use the Command Lookup Tool (registered customers only) .

Configuration Procedure

In this example, a password is configured for all users attempting to use the console.

1. From the privileged EXEC (or "enable") prompt, enter configuration mode and then switch to line configuration mode using the following commands. Notice that the prompt changes to reflect the current mode.

```
2. router#configure terminal
3. Enter configuration commands, one per line.  End with CNTL/Z.
4. router(config)#line con 0
   router(config-line)#
```

5. Configure the password, and enable password checking at login.

```
6. router(config-line)#password letmein
7. router(config-line)#login
```

8. Exit configuration mode.

```
9. router(config-line)#end
10.    router#
   %SYS-5-CONFIG_I: Configured from console by console
```

Note: Do not save configuration changes to **line con 0** until your ability to log in has been verified.

Note: Under the line console configuration, **login** is a required configuration command to enable password checking at login. Console authentication requires both the **password** and the **login** commands to work.

Verify the Configuration

Examine the configuration of the router to verify that the commands have been properly entered:

Certain **show** commands are supported by the Output Interpreter Tool (registered customers only) , which allows you to view an analysis of **show** command output.

- **show running-config** - displays the current configuration of the router.

```
router#show running-config
Building configuration...
...

!--- Lines omitted for brevity


!
line con 0
password letmein
login
line 1 8
line aux 0
line vty 0 4
!
end
```

To test the configuration, log off the console and log in again, using the configured password to access the router:

```
router#exit
```

```
router con0 is now available

Press RETURN to get started.

User Access Verification
Password:

!--- Password entered here is not displayed by the router


router>
```

Note: Before performing this test, ensure that you have an alternate connection into the router, such as Telnet or dial-in, in case there is a problem logging back into the router.

Troubleshoot Login Failure

If you cannot log back into the router and you have not saved the configuration, reloading the router will eliminate any configuration changes you have made.

If the configuration changes were saved and you cannot login to the router, you will have to perform a password recovery. See Password Recovery Procedures to find instructions for your particular platform.

Configure Local User-Specific Passwords

To establish a username-based authentication system, use the **username** command in global configuration mode. To enable password checking at login, use the **login local** command in line configuration mode.

Configuration Procedure

In this example, passwords are configured for users attempting to connect to the router on the VTY lines using Telnet.

1. From the privileged EXEC (or "enable") prompt, enter configuration mode and enter username/password combinations, one for each user for whom you want to allow access to the router:

```
router#configure terminal
    Enter configuration commands, one per line.  End with CNTL/Z.
    router(config)#username russ password montecito
    router(config)#username cindy password belgium
    router(config)#username mike password rottweiler
```

7. Switch to line configuration mode, using the following commands. Notice that the prompt changes to reflect the current mode.

```
router(config)#line vty 0 4
```

```
router(config-line)#
```

9. Configure password checking at login.

```
10.    router(config-line)#login local
```

11. Exit configuration mode.

```
12.    router(config-line)#end
13.    router#
   %SYS-5-CONFIG_I: Configured from console by console
```

Note: In order to disable auto Telnet when you type a name on the CLI, configure **no logging preferred** on the line that is used. While **transport preferred none** provides the same output, it also disables auto Telnet for the defined host that are configured with the **ip host** command. This is unlike the **no logging preferred** command, which stops it for undefined hosts and lets it work for the defined ones.

Verify the Configuration

Examine the configuration of the router to verify that the commands have been properly entered:

- **show running-config** - displays the current configuration of the router.

```
router#show running-config
Building configuration...
!

!--- Lines omitted for brevity



!
username russ password 0 montecito
username cindy password 0 belgium
username mike password 0 rottweiler
!

!--- Lines omitted for brevity



!
line con 0
line 1 8
line aux 0
line vty 0 4
 login local
!
end
```

To test this configuration, a Telnet connection must be made to the router. This can be done by connecting from a different host on the network, but you can also test from the router itself by telnetting to the IP address of any interface on the router that is in an up/up state as seen in the output of the **show interfaces** command.

Here is a sample output if the address of **interface ethernet 0** were 10.1.1.1:

```
router#telnet 10.1.1.1
Trying 10.1.1.1 ... Open


User Access Verification


Username: mike
Password:

!--- Password entered here is not displayed by the router



router
```

Troubleshoot User-specific Password Failure

Usernames and passwords are case-sensitive. Users attempting to log in with an incorrectly cased username or password will be rejected.

If users are unable to log into the router with their specific passwords, reconfigure the username and password on the router.

Configure AUX Line Password

In order to specify a password on the AUX line, issue the **password** command in line configuration mode. In order to enable password checking at login, issue the **login** command in line configuration mode.

Configuration Procedure

In this example, a password is configured for all users attempting to use the AUX port.

1. Issue the **show line** command in order to verify the line used by the AUX port.

```
2. R1#show line
3.
4.    Tty Typ     Tx/Rx    A Modem Roty AccO AccI  Uses  Noise  Overruns  Int
5. *    0 CTY                   -    -     -    -     -     0      0     0/0
   -
6.     65 AUX  9600/9600  -    -     -    -    -    0     1     0/0
   -
```

```
7.      66 VTY                    -    -      -    -    -     0       0      0/0
   -
        67 VTY                    -    -      -    -    -     0       0      0/0
   -
```

8. In this example, the AUX port is on line 65. Issue these commands in order to configure the router AUX line:

```
9. R1# conf t
10.    R1(config)# line 65
11.    R1(config-line)#modem inout
12.    R1(config-line)#speed 115200
13.    R1(config-line)#transport input all
14.    R1(config-line)#flowcontrol hardware
15.    R1(config-line)#login
16.    R1(config-line)#password cisco
17.    R1(config-line)#end
   R1#
```

Verify Configuration

Examine the configuration of the router in order to verify that the commands have been properly entered:

- The **show running-config** command displays the current configuration of the router:

```
R1#show running-config
Building configuration...
!

!--- Lines omitted for brevity.

line aux 0
 password cisco
 login
 modem InOut
 transport input all
 speed 115200
 flowcontrol hardware


!--- Lines omitted for brevity.

!
end
```

Configure AAA Authentication for Login

To enable authentication, authorization, and accounting (AAA) authentication for logins, use the **login authentication** command in line configuration mode. AAA services must also be configured.

Configuration Procedure

In this example, the router is configured to retrieve users' passwords from a TACACS+ server when users attempt to connect to the router.

Note: Configuring the router to use other types of AAA servers (RADIUS, for example) is similar. See Configuring Authentication for additional information.

Note: This document does not address configuration of the AAA server itself. Refer to Security Server Protocols for information on configuring the AAA server.

1. From the privileged EXEC (or "enable") prompt, enter configuration mode and enter the commands to configure the router to use AAA services for authentication:

```
router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
router(config)#aaa new-model
router(config)#aaa authentication login my-auth-list tacacs+
router(config)#tacacs-server host 192.168.1.101
router(config)#tacacs-server key letmein
```

8. Switch to line configuration mode using the following commands. Notice that the prompt changes to reflect the current mode.

```
router(config)#line 1 8
router(config-line)#
```

10. Configure password checking at login.

```
router(config-line)#login authentication my-auth-list
```

12. Exit configuration mode.

```
router(config-line)#end
router#
%SYS-5-CONFIG_I: Configured from console by console
```

Verify the Configuration

Examine the configuration of the router to verify that the commands have been properly entered:

- **show running-config** - displays the current configuration of the router.

```
router#write terminal
Building configuration...

Current configuration:
!
```

```
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname router
!
aaa new-model
aaa authentication login my-auth-list tacacs+
!

!--- Lines omitted for brevity



...
!
tacacs-server host 192.168.1.101
tacacs-server key letmein
!
line con 0
line 1 8
 login authentication my-auth-list
line aux 0
line vty 0 4
!
end
```

banner motd

- To define and enable a message-of-the-day (MOTD) banner, use the **banner motd** global configuration command. To delete the MOTD banner, use the **no** form of this command.
- **banner motd** *d message d*
- no **banner motd**

- **Syntax Description**

d	Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the banner message.
message	Message text. You can include tokens in the form $(*token*) in the message text. Tokens will be replaced with the corresponding configuration variable.

- **Defaults**
- Disabled (no MOTD banner is displayed).

- **Command Modes**
- Global configuration

- **Command History**

Release	Modification
10.0	This command was introduced.
11.3(7.5) AA	Token functionality was introduced.
12.0(3) T	Token functionality was integrated in the 12.0 T release train.

- **Usage Guidelines**
- Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.
- This MOTD banner is displayed to all terminals connected and is useful for sending messages that affect all users (such as impending system shutdowns). Use the **no exec-banner** or **no motd-banner** command to disable the MOTD banner on a line. The **no exec-banner** command also disables the EXEC banner on the line.
- When a user connects to the router, the MOTD banner appears before the login prompt. After the user logs in to the router, the EXEC banner or incoming banner will be displayed, depending on the type of connection. For a reverse Telnet login, the incoming banner will be displayed. For all other connections, the router will display the EXEC banner.
- To customize the banner, use tokens in the form $(token) in the message text. Tokens will display current Cisco IOS configuration variables, such as the router's host name and IP address. The tokens are described in Table 12.

Table 12 banner motd Tokens

Token	Information Displayed in the Banner
$(hostname)	Displays the host name for the router.
$(domain)	Displays the domain name for the router.
$(line)	Displays the vty or tty (asynchronous) line number.
$(line-desc)	Displays the description attached to the line.

- **Examples**
- The following example configures an MOTD banner. The pound sign (#) is used as a delimiting character.
- `Router# banner motd # Building power will be off from 7:00 AM until 9:00 AM this coming`
- `Tuesday. #`
-
- The following example configures an MOTD banner with a token. The percent sign (%) is used as a delimiting character.
- `darkstar(config)# banner motd %`
- `Enter TEXT message.  End with the character '%'.`
- `Notice: all routers in $(domain) will be upgraded beginning April 20`
- `%`
-
- When the MOTD banner is executed, the user will see the following. Notice that the $(*token*) syntax is replaced by the corresponding configuration variable.

You can configure three main types of banners on your Cisco switch, as shown here:

- **Message of the Day (MOTD):** This type of logon message has been around for a long time on Unix and mainframe systems. The idea of the message is to display a temporary notice to users, such as issues with system availability.

  However, because the message displays when a user connects to the device prior to login, most network administrators are now using it to display legal notices regarding access to the switch, such as *unauthorized access to this device is prohibited and violators will be prosecuted to the full extent of the law* and other such cheery endearments.

- **Login:** This banner is displayed before login to the system, but after the MOTD banner is displayed. Typically, this banner is used to display a permanent message to the users.
- **Exec:** This banner displays after the login is complete when the connecting user enters User EXEC mode. Whereas all users who attempt to connect to the switch see the other banners, only users who successfully log on to the switch see this banner, which can be used to post reminders to your network administrators.

Configure an IPv4 Router Interface

One distinguishing feature between switches and routers is the type of interfaces supported by each. For example, Layer 2 switches support LANs and, therefore, have multiple FastEthernet or Gigabit Ethernet ports.

Routers support LANs and WANs and can interconnect different types of networks; therefore, they support many types of interfaces. For example, G2 ISRs have one or two integrated Gigabit Ethernet interfaces and

High-Speed WAN Interface Card (HWIC) slots to accommodate other types of network interfaces, including serial, DSL, and cable interfaces.

To be available, an interface must be:

- **If using IPv4, configured with an address and a subnet mask**: Use the **ip address** *ip-address subnet-mask* interface configuration command.
- **Activated**: By default, LAN and WAN interfaces are not activated (**shutdown**). To enable an interface, it must be activated using the **no shutdown** command. (This is similar to powering on the interface.) The interface must also be connected to another device (a hub, a switch, or another router) for the physical layer to be active.

Optionally, the interface could also be configured with a short description. It is good practice to configure a description on each interface. The description text is limited to 240 characters. On production networks, a description can be helpful in troubleshooting by providing information about the type of network to which the interface is connected. If the interface connects to an ISP or service carrier, it is helpful to enter the third-party connection and contact information.

Depending on the type of interface, additional parameters may be required. For example, in the lab environment, the serial interface connecting to the serial cable end labeled DCE must be configured with the **clock rate** command.

Accidentally using the **clock rate** command on a DTE interface generates a "%Error: This command applies only to DCE interface" message.

The steps to configure an IPv4 interface on a router are:

- **Step 1.** Add a description. Although optional, it is a necessary component for documenting a network.
- **Step 2.** Configure the IPv4 address.

- **Step 3.** Configure a clock rate on Serial interfaces. This is only necessary on the DCE device in our lab environment and does not apply to Ethernet interfaces.
- **Step 4.** Enable the interface.

For example, the following commands would configure the three directly connected interfaces of router R1 shown in Figure 4-14 (in the previous section):

```
R1(config)# interface gigabitethernet 0/0
R1(config-if)# description Link to LAN 1
R1(config-if)# ip address 192.168.10.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)#
R1(config)# interface gigabitethernet 0/1
R1(config-if)# description Link to LAN 2
R1(config-if)# ip address 192.168.11.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)#
R1(config)# interface serial 0/0/0
R1(config-if)# description Link to R2
R1(config-if)# ip address 209.165.200.225 255.255.255.252
R1(config-if)# clock rate 128000
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)#
```

Configure an IPv6 Router Interface

Configuring an IPv6 interface is similar to configuring an interface for IPv4. Most IPv6 configuration and verification commands in the Cisco IOS are very similar to their IPv4 counterparts. In many cases, the only difference uses **ipv6** in place of **ip** in commands.

An IPv6 interface must be:

- **Configured with IPv6 address and subnet mask**: Use the **ipv6 address** *ipv6-address*/*prefix-length* [**link-local** | **eui-64**] interface configuration command.
- **Activated**: The interface must be activated using the **no shutdown** command.

An interface can generate its own IPv6 link-local address without having a global unicast address by using the **ipv6 enable** interface configuration command.

Unlike IPv4, IPv6 interfaces will typically have more than one IPv6 address. At a minimum, an IPv6 device must have an IPv6 link-local address but will most likely also have an IPv6 global unicast address. IPv6 also supports the ability for an interface to have multiple IPv6 global unicast addresses from the same subnet. The following commands can be used to statically create a global unicast or link-local IPv6 address:

- **ipv6 address** *ipv6-address/prefix-length*: Creates a global unicast IPv6 address as specified.
- **ipv6 address** *ipv6-address/prefix-length* **eui-64**: Configures a global unicast IPv6 address with an interface identifier (ID) in the low-order 64 bits of the IPv6 address using the EUI-64 process.
- **ipv6 address** *ipv6-address/prefix-length* **link-local**: Configures a static link-local address on the interface that is used instead of the link-local address that is automatically configured when the global unicast IPv6 address is assigned to the interface or enabled using the **ipv6 enable** interface command. Recall, the **ipv6 enable** interface command is used to automatically create an IPv6 link-local address whether or not an IPv6 global unicast address has been assigned.

The steps to configure an IPv6 interface on a router are:

- **Step 1.** Add a description. Although optional, it is a necessary component for documenting a network.
- **Step 2.** Configure the IPv6 global unicast address. Configuring a global unicast address automatically creates a link-local IPv6 address.
- **Step 3.** Configure a link-local unicast address which automatically assigns a link-local IPv6 address and overrides any previously assigned address.
- **Step 4.** Configure a clock rate on Serial interfaces. This is only necessary on the DCE device in our lab environment and does not apply to Ethernet interfaces.
- **Step 5.** Enable the interface.

In the example topology shown in Figure 4-15, R1 must be configured to support the following IPv6 global network addresses:

- 2001:0DB8:ACAD:0001:/64 (2001:DB8:ACAD:1::/64)
- 2001:0DB8:ACAD:0002:/64 (2001:DB8:ACAD:2::/64)
- 2001:0DB8:ACAD:0003:/64 (2001:DB8:ACAD:3::/64)

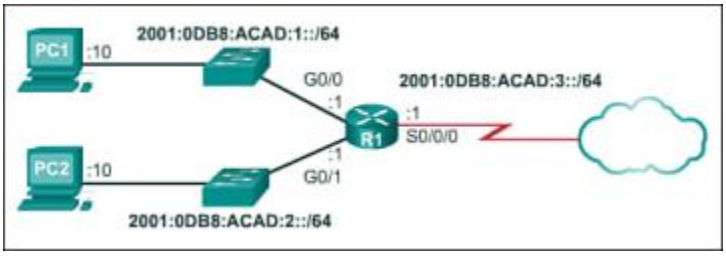


Figure 4-15 IPv6 Topology

When the router is configured using the **ipv6 unicast-routing** global configuration command, the router begins sending ICMPv6 Router Advertisement messages out the interface. This enables a PC connected to the interface to automatically configure an IPv6 address and to set a default gateway without needing the services of a DHCPv6 server. Alternatively, a PC connected to the IPv6 network can get its IPv6 address statically assigned, as shown in Figure 4-16. Notice that the default gateway address configured for PC1 is the IPv6 global unicast address of the R1 Gigabit Ethernet 0/0 interface.

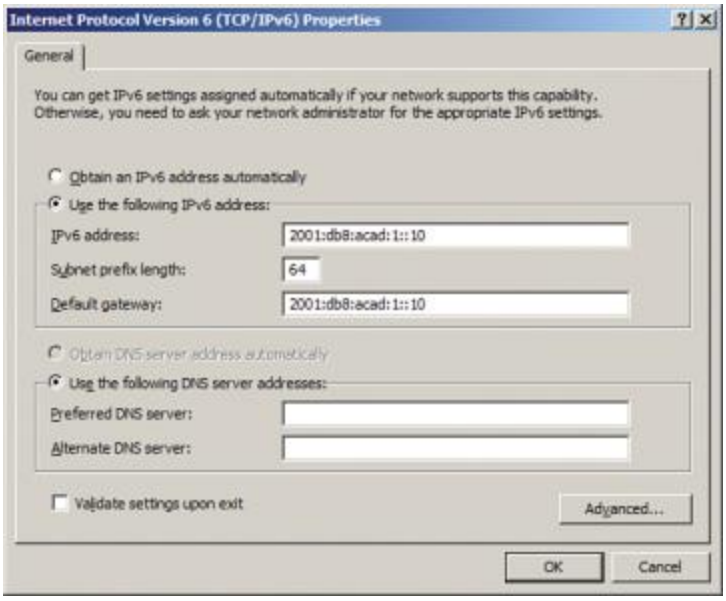


Figure 4-16 Statically Assign an IPv6 Address to PC1

For example, the following commands would configure the IPv6 global unicast addresses of the three directly connected interfaces of the R1 router shown in Figure 4-15:

```
R1# configure terminal
R1(config)# interface gigabitethernet 0/0
R1(config-if)# description Link to LAN 1
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)#
```

```
R1(config)# interface gigabitethernet 0/1
R1(config-if)# description Link to LAN 2
R1(config-if)# ipv6 address 2001:db8:acad:2::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)#
R1(config)# interface serial 0/0/0
R1(config-if)# description Link to R2
R1(config-if)# ipv6 address 2001:db8:acad:3::1/64
R1(config-if)# clock rate 128000
R1(config-if)# no shutdown
R1(config-if)#
```

Configure an IPv4 Loopback Interface

Another common configuration of Cisco IOS routers is enabling a loopback interface.

The ***loopback interface*** is a logical interface internal to the router. It is not assigned to a physical port and can therefore never be connected to any other device. It is considered a software interface that is automatically placed in an "up/up" state, as long as the router is functioning.

The loopback interface is useful in testing and managing a Cisco IOS device because it ensures that at least one interface will always be available. For example, it can be used for testing purposes, such as testing internal routing processes, by emulating networks behind the router.

Additionally, the IPv4 address assigned to the loopback interface can be significant to processes on the router that use an interface IPv4 address for identification purposes, such as the Open Shortest Path First (OSPF) routing process. By enabling a loopback interface, the router will use the always available loopback interface address for identification, rather than an IP address assigned to a physical port that may go down.

The steps to configure a loopback interface on a router are:

- **Step 1.** Create the loopback interface using the **interface loopback** *number* global configuration command.
- **Step 2.** Add a description. Although optional, it is a necessary component for documenting a network.
- **Step 3.** Configure the IP address.

For example, the following commands configure a loopback interface of the R1 router shown in Figure 4-14 (shown earlier in the chapter):

```
R1# configure terminal
R1(config)# interface loopback 0
R1(config-if)# ip address 10.0.0.1 255.255.255.0
R1(config-if)# exit
R1(config)#
```

A loopback interface is always enabled and therefore does not require a **no shutdown** command. Multiple loopback interfaces can be enabled on a router. The IPv4 address for each loopback interface must be unique and unused by any other interface.

How to Configure DNS

- Mapping Hostnames to IP Addresses
- Customizing DNS
- Configuring DNS Spoofing
- Configuring the Device as a DNS Server
- Disabling DNS Queries for ISO CLNS Addresses
- Verifying DNS

Mapping Hostnames to IP Addresses

Perform this task to map hostnames to IP addresses.

A name server is used to keep track of information associated with domain names. A name server can maintain a database of hostname-to-address mappings. Each name can map to one or more IP addresses. In order to use this service to map domain names to IP addresses, you must specify a name server.

The name lookup system can be statically configured using the commands described in this task. Some other functions in Cisco IOS software, such as DHCP, can dynamically modify the state of the name lookup system. Use the show hosts command to display the cached hostnames and the DNS configuration.

SUMMARY STEPS

1. enable
2. configure terminal
3. ip host name [tcp-port-number] address1 [address2 ... address8]
4. Do one of the following:
    - ip domain name name

- ip domain list name

5. ip name-server server-address1 [server-address2 ... server-address6]
6. ip domain lookup [source-interface interface-type interface-number]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	ip host name [tcp-port-number] address1 [address2 ... address8] **Example:** `Device(config)# ip host cisco-rtp 192.168.0.148`	Defines a static hostname-to-address mapping in the hostname cache. • The host IP address can be an IPv4 or IPv6 address. • Typically, it is easier to refer to network devices by symbolic names rather than numerical addresses (services such as Telnet can use hostnames or addresses). Hostnames and IP addresses can be associated with one another through static or dynamic means. • Manually assigning hostnames to addresses is useful when dynamic mapping is not available.
Step 4	Do one of the following: • ip domain name name • ip domain list name	(Optional) Defines a default domain name that the Cisco IOS software will use to complete unqualified hostnames. or

	Example:
	`Device(config)# ip domain name cisco.com`
	
	Example:
	
	Example:
	`Device(config)# ip domain list cisco1.com`
		(Optional) Defines a list of default domain names to complete unqualified hostnames.
		
		• You can specify a default domain name that the Cisco IOS software will use to complete domain name requests. You can specify either a single domain name or a list of domain names. Any hostname that does not contain a complete domain name will have the default domain name you specify appended to it before the name is looked up.
		
		Note If there is no domain list, the domain name that you specified with the ip domain name global configuration command is used. If there is a domain list, the default domain name is not used. The ip domain list command is similar to the ip domain name command, except that with the ip domain list command you can define a list of domains, each to be tried in turn until the system finds a match.

Step 5	ip name-server server-address1 [server-address2 ... server-address6] **Example:** `Device(config)# ip name-server 172.16.1.111 172.16.1.2`	Specifies one or more hosts (up to six) that can function as a name server to supply name information for DNS.
Step 6	ip domain lookup [source-interface interface-type interface-number] **Example:** `Device(config)# ip domain lookup`	(Optional) Enables DNS-based address translation. • DNS is enabled by default. Use this command if DNS has been disabled.

Customizing DNS

Perform this task to customize your DNS configuration.

In a multiple server configuration without the DNS round-robin functionality, many programs will use the first host server/IP address for the whole time to live (TTL) of the cache and use the second and third host servers/IP addresses only in the event of host failure. This behavior presents a problem when a high volume of users all arrive at the first host during the TTL time. For example, the network access server (NAS) sends out a DNS query. The DNS servers reply with a list of the configured IP addresses to the NAS. The NAS then caches these IP addresses for a given time (for example, five minutes). All users that dial in during the five minute TTL time will land on one host, the first IP address in the list.

In a multiple server configuration with the DNS round-robin functionality, the DNS server returns the IP address of all hosts to rotate between the cache of hostnames. During the TTL of the cache, users are distributed among the hosts. This functionality distributes calls across the configured hosts and reduces the number of DNS queries.

In a scheduling algorithm, processes are activated in a fixed cyclic order. Processes that are waiting for other events, like termination of a child process or an input or output operation, cannot proceed and hence they return control to the scheduler. If the TTL of the process times out just before the event (for which it was waiting) occurs, then the event will not be handled until all the other processes are activated.

The DNS round-robin functionality is applicable only for the DNS lookups on a device and is not applicable to another client pointing to the device.

SUMMARY STEPS

1. enable
2. configure terminal
3. ip domain timeout seconds
4. ip domain retry number
5. ip domain round-robin

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	ip domain timeout seconds **Example:** `Device(config)# ip domain timeout 17`	(Optional) Specifies the amount of time to wait for a response to a DNS query. • If the ip domain timeout command is not configured, the Cisco IOS software will wait 3 seconds for a response to a DNS query.
Step 4	ip domain retry number **Example:** `Device(config)# ip domain retry 10`	(Optional) Specifies the number of times to retry sending DNS queries. • If the ip domain retry command is not configured, the Cisco IOS software will retry DNS queries twice.
Step 5	ip domain round-robin **Example:** `Device(config)# ip domain round-robin`	(Optional) Enables round-robin functionality on DNS servers.

Configuring DNS Spoofing

Perform this task to configure DNS spoofing.

DNS spoofing is designed to allow a device to act as a proxy DNS server and "spoof" replies to any DNS queries using either the configured IP address in the ip dns spoofing ip-address command or the IP address of the incoming interface for the query. This feature is useful for devices where the interface toward the Internet service provider (ISP) is not up. Once the interface to the ISP is up, the device forwards DNS queries to the real DNS servers.

This feature turns on DNS spoofing and is functional if any of the following conditions are true:

- The **no ip domain lookup** command is configured.
- IP name server addresses are not configured.
- There are no valid interfaces or routes for sending to the configured name server addresses.

If these conditions are removed, DNS spoofing will not occur.

SUMMARY STEPS

1. enable
2. configure terminal
3. ip dns server
4. ip dns spoofing [ip-address]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	ip dns server **Example:** `Device(config)# ip dns server`	Activates the DNS server on the device.
Step 4	ip dns spoofing [ip-address] **Example:** `Device(config)# ip dns spoofing 192.168.15.1`	Configures DNS spoofing. • The IP address used for DNS spoofing can be an IPv4 or IPv6 address. • The device will respond to the DNS query with the configured ip-address when queried for any hostname other than its own. • The device will respond to the DNS query with the IP address of the incoming interface when queried for its own hostname.

Configuring the Device as a DNS Server

Perform this task to configure the device as a DNS server.

A Cisco IOS device can provide service to DNS clients, acting as both a caching name server and as an authoritative name server for its own local host table.

When configured as a caching name server, the device relays DNS requests to other name servers that resolve network names into network addresses. The caching name server caches information learned from other name servers so that it can answer requests quickly, without having to query other servers for each transaction.

When configured as an authoritative name server for its own local host table, the device listens on port 53 for DNS queries and then answers DNS queries using the permanent and cached entries in its own host table.

An authoritative name server usually issues zone transfers or responds to zone transfer requests from other authoritative name servers for the same zone. However, the Cisco IOS DNS server does not perform zone transfers.

When it receives a DNS query, an authoritative name server handles the query as follows:

- If the query is for a domain name that is not under its zone of authority, the authoritative name server determines whether to forward the query to specific back-end name servers based on whether IP DNS-based hostname-to-address translation has been enabled via the ip domain lookup command.
- If the query is for a domain name that is under its zone of authority and for which it has configuration information, the authoritative name server answers the query using the permanent and cached entries in its own host table.
- If the query is for a domain name that is under its zone of authority but for which it does not have any configuration information, the authoritative name server does not forward the query elsewhere for a response; instead the authoritative name server simply replies that no such information exists.

Unless Distributed Director is enabled, the TTL on locally defined resource records will always be ten seconds, regardless of any authority record parameters that may have been specified for the DNS name server by the use of the ip dns primary command.

SUMMARY STEPS

1. enable

2. configure terminal

3. ip dns server

4. ip name-server server-address1 [server-address2... server-address6]

5. ip dns server queue limit {forwarder queue-size-limit | director queue-size-limit}

6. ip host [vrf vrf-name] [view view-name] hostname {address1 [address2 ... address8] | additional address9 [address10 ... addressn]}

7. ip dns primary domain-name soa primary-server-name mailbox-name [refresh-interval [retry-interval [expire-ttl [minimum-ttl]]]]

8. ip host domain-name ns server-name

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	ip dns server **Example:** `Device(config)# ip dns server`	Enables the DNS server.
Step 4	ip name-server server-address1 [server-address2... server-address6] **Example:** `Device(config)# ip name-server 192.168.2.120 192.168.2.121`	(Optional) Configures other DNS servers: • Cisco IOS resolver name servers • DNS server forwarders **Note** If the Cisco IOS name server is being configured to respond only to domain names for which it is authoritative, there is no need to configure other DNS servers.
Step 5	ip dns server queue limit {forwarder queue-size-limit | director queue-size-limit} **Example:** `Device(config)# ip dns server queue limit forwarder 10`	(Optional) Configures a limit to the size of the queues used by the DNS server processes. • The director keyword was removed in Cisco IOS Release 12.4(24)T.

Step 6	ip host [vrf vrf-name] [view view-name] hostname {address1 [address2 ... address8] | additional address9 [address10 ... addressn]} **Example:** `Device(config)# ip host user1.example.com 192.168.201.5 192.168.201.6`	(Optional) Configures local hosts.
Step 7	ip dns primary domain-name soa primary-server-name mailbox-name [refresh-interval [retry-interval [expire-ttl [minimum-ttl]]]] **Example:** `Device(config)# ip dns primary example.com soa ns1.example.com mb1.example.com`	Configures the device as the primary DNS name server for a domain (zone) and as the start of authority (SOA) record source (which designates the start of a zone). **Note** Unless Distributed Director is enabled, the TTL on locally defined resource records will always be ten seconds.
Step 8	ip host domain-name ns server-name **Example:** `Device(config)# ip host example.com ns ns1.example.com`	(Optional) Configures the device to create an name server (NS) resource record to be returned when the DNS server is queried for the associated domain. • This configuration is needed only if the zone for which the system is authoritative will also be served by other name servers.

Examples

This section provides examples of debugging output that is logged when a device is configured as an authoritative name server for its own local host table and the debug domain command is in effect:

For DNS-based X.25 routing, the debug x25 events command supports functionality to describe the events that occur while the X.25 address is being resolved to an IP address using a DNS server. The

debug domain command can be used along with debug x25 events to observe the whole DNS-based X.25 routing data flow.

Debugging Output for Relaying a DNS Query to Another Name Server Example

The following is sample output from the debug domain command that corresponds to relaying a DNS query to another name server when the device is configured as an authoritative name server for its own local host table:

```
Apr  4 22:18:32.183: DNS: Incoming UDP query (id#18713)
Apr  4 22:18:32.183: DNS: Type 1 DNS query (id#18713) for host 'ns1.example.com' from
192.0.2.120(1283)
Apr  4 22:18:32.183: DNS: Re-sending DNS query (type 1, id#18713) to 192.0.2.121
Apr  4 22:18:32.211: DNS: Incoming UDP query (id#18713)
Apr  4 22:18:32.211: DNS: Type 1 response (id#18713) for host <ns1.example.com> from
192.0.2.121(53)
Apr  4 22:18:32.215: DOM: dom2cache: hostname is ns1.example.com, RR type=1, class=1,
ttl=86400, n=4
Apr  4 22:18:32.215: DNS: Forwarding back A response - no director required
Apr  4 22:18:32.215: DNS: Finished processing query (id#18713) in 0.032 secs
Apr  4 22:18:32.215: DNS: Forwarding back reply to 192.0.2.120/1283
```

Debugging Output for Servicing a DNS Query from the Local Host Table Example

The following is sample output from the debug domain command that corresponds to servicing a DNS query from the local host table when the device is configured as an authoritative name server for its own local host table:

```
Apr  4 22:16:35.279: DNS: Incoming UDP query (id#8409)
Apr  4 22:16:35.279: DNS: Type 1 DNS query (id#8409) for host 'ns1.example.com' from
192.0.2.120(1279)
Apr  4 22:16:35.279: DNS: Finished processing query (id#8409) in 0.000 secs
```

Disabling DNS Queries for ISO CLNS Addresses

Perform this task to disable DNS queries for International Organization for Standardization (ISO) Connectionless Network Service (CLNS) addresses.

If your device has both IP and ISO CLNS enabled and you want to use ISO CLNS network service access point (NSAP) addresses, you can use the DNS to query these addresses, as documented in RFC 1348. This feature is enabled by default.

SUMMARY STEPS

1. enable
2. configure terminal
3. no ip domain lookup nsap

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	no ip domain lookup nsap **Example:** `Device(config)# no ip domain lookup nsap`	Disables DNS queries for ISO CLNS addresses.

Verifying DNS

Perform this task to verify your DNS configuration.

1. enable
2. ping hosts
3. show hosts

SUMMARY STEPS

1. enable
2. ping hosts
3. show hosts

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	ping hosts **Example:** `Device# ping cisco-rtp`	Diagnoses basic network connectivity. • After the DNS configuration is set, you can verify the DNS server by using a hostname to ping or telnet to a device.
Step 3	show hosts **Example:** `Device# show hosts`	Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses. • After a name is resolved using DNS, use the show hosts command to view the cached hostnames and the DNS configuration.

3. Verify Connectivity of Directly Connected Networks

The first task to undertake once the basic settings and interfaces are configured is to verify and validate the configured settings. This is an important step and should be done before any other configurations are added to the router.

Verify Interface Settings

There are several **show** commands that can be used to verify the operation and configuration of an interface. The following three commands are especially useful to quickly identify an interface status:

- **show ip interface brief**: Displays a summary for all interfaces, including the IPv4 address of the interface and current operational status.
- **show ip route**: Displays the contents of the IPv4 routing table stored in RAM. In Cisco IOS 15, active interfaces should appear in the routing table with two related entries identified by the code `'C'` (Connected) or `'L'` (Local). In previous IOS versions, only a single entry with the code `'C'` will appear.
- **show running-config interface** *interface-id*: Displays the commands configured on the specified interface.

Figure 4-17 displays the output of the **show ip interface brief** command.

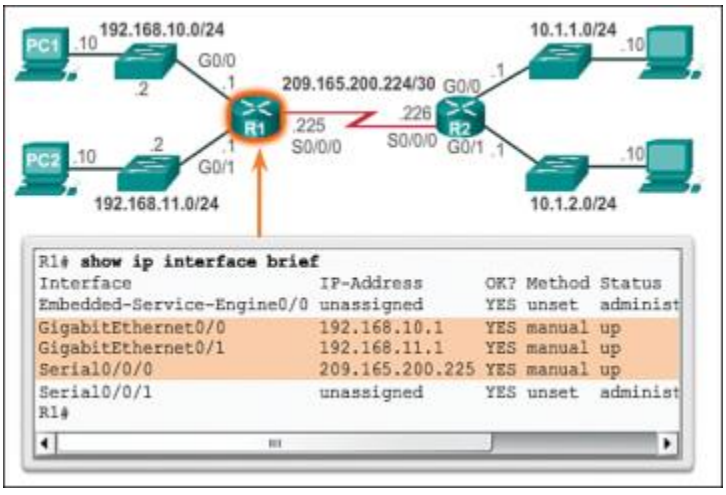


Figure 4-17 Display Interface Summaries

The output reveals that the LAN interfaces and the WAN link are all activated and operational as indicated by the Status of “up” and Protocol of “up.” A different output would indicate a problem with either the configuration or the cabling.

In Figure 4-17, the Embedded-Service-Engine0/0 interface is displayed because Cisco ISRs G2 have dual-core CPUs on the motherboard. The Embedded-Service-Engine0/0 interface is outside the scope of this course.

Figure 4-18 displays the output of the **show ip route** command.

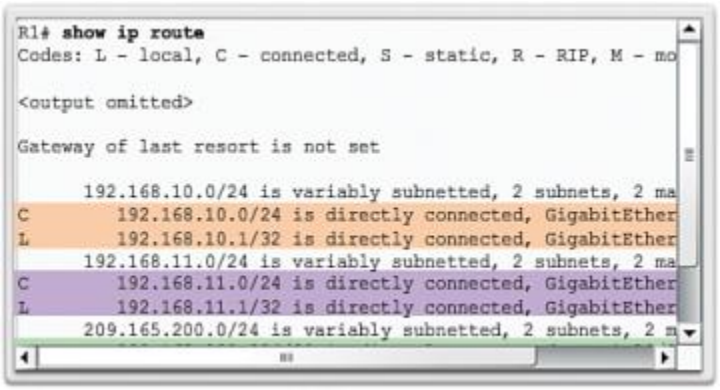


Figure 4-18 Verify the IPv4 Routing Table

The entire output of the **show ip route** command in Figure 4-18 can be viewed in the online course on page 1.1.4.1 graphic number 2.

Notice the three directly connected network entries and the three local host route interface entries. A local host route has an administrative distance of 0. It also has a /32 mask for IPv4, and a /128 mask for IPv6. The local host route is for routes on the router owning the IP address. It is used to allow the router to process packets destined to that IP.

Figure 4-19 displays the output of the **show running-config interface** command. The output displays the current commands configured on the specified interface.

```
R1# show running-config interface gigabitEthernet 0/0
Building configuration...

Current configuration : 128 bytes
!
interface GigabitEthernet0/0
 description Link to LAN 1
 ip address 192.168.10.1 255.255.255.0
 duplex auto
 speed auto
end

R1#
```

Figure 4-19 Verify an Interface Configuration

The following two commands are used to gather more detailed interface information:

- **show interfaces**: Displays interface information and packet flow count for all interfaces on the device
- **show ip interface**: Displays the IPv4-related information for all interfaces on a router

Verify IPv6 Interface Settings

The commands to verify the IPv6 interface configuration are similar to the commands used for IPv4.

The **show ipv6 interface brief** command in Figure 4-20 displays a summary for each of the interfaces.

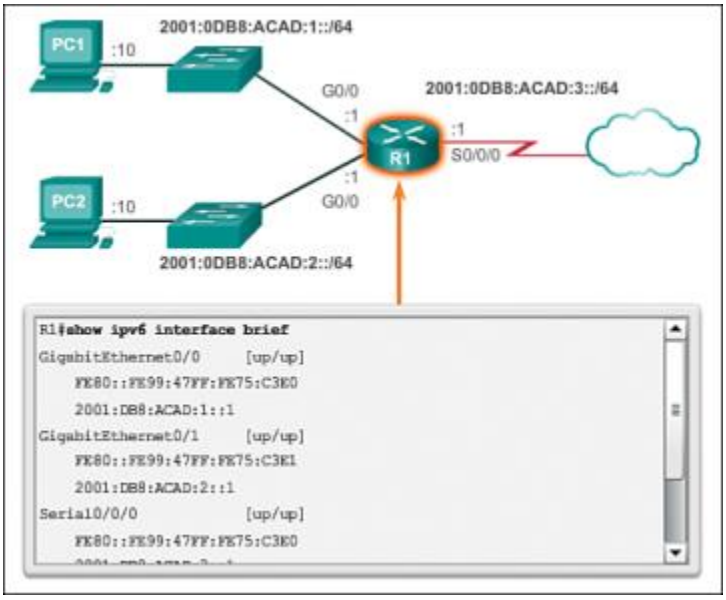


Figure 4-20 Verify the R1 IPv6 Interface Status

The entire output of the **show ipv6 interface brief** command in Figure 4-20 can be viewed in the online course on page 1.1.4.2 graphic number 1.

The "up/up" output on the same line as the interface name indicates the Layer 1/Layer 2 interface state. This is the same as the Status and Protocol columns in the equivalent IPv4 command.

The output displays two configured IPv6 addresses per interface. One address is the IPv6 global unicast address that was manually entered. The other address, which begins with FE80, is the link-local unicast address for the interface. A link-local address is automatically added to an interface whenever a global unicast address is assigned. An IPv6 network interface is required to have a link-local address, but not necessarily a global unicast address.

The **show ipv6 interface gigabitethernet 0/0** command output shown in Figure 4-21 displays the interface status and all of the IPv6 addresses belonging to the interface. Along with the link-local address and global unicast address, the output includes the multicast addresses assigned to the interface, beginning with prefix FF02.

```
R1#show ipv6 interface gigabitEthernet 0/0
GigabitEthernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::32F7:DFF:FEA3:DA0
  No Virtual link-local address(es):
  Global unicast address(es):
    2001:DB8:ACAD:1::1, subnet is 2001:DB8:ACAD:1::/64
  Joined group address(es):
    FF02::1
    FF02::1:FF00:1
```

Figure 4-21 Verify the IPv6 Configuration on R1 G0/0

The **show ipv6 route** command shown in Figure 4-22 can be used to verify that IPv6 networks and specific IPv6 interface addresses have been installed in the IPv6 routing table. The **show ipv6 route** command will only display IPv6 networks, not IPv4 networks.

```
<output omitted>

C   2001:DB8:ACAD:1::/64 [0/0]
     via GigabitEthernet0/0, directly connected
L   2001:DB8:ACAD:1::1/128 [0/0]
     via GigabitEthernet0/0, receive
C   2001:DB8:ACAD:2::/64 [0/0]
     via GigabitEthernet0/1, directly connected
L   2001:DB8:ACAD:2::1/128 [0/0]
```

Figure 4-22 Verify the R1 IPv6 Routing Table

The entire output of the **show ipv6 route** command in Figure 4-22 Within the routing table, a '`C`' next to a route indicates that this is a directly connected network. When the router interface is configured with a global unicast address and is in the "up/up" state, the IPv6 prefix and prefix length is added to the IPv6 routing table as a connected route.

The IPv6 global unicast address configured on the interface is also installed in the routing table as a local route, as indicated with an '`L`' next to the route entry. The local route has a /128 prefix. Local routes are used by the routing table to efficiently process packets with the interface address of the router as the destination.

The **ping** command for IPv6 is identical to the command used with IPv4 except that an IPv6 address is used. As shown in Figure 4-23, the **ping** command is used to verify Layer 3 connectivity between R1 and PC1.

```
R1#ping 2001:db8:acad:1::10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:ACAD:1::10, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5)
R1#
```

Figure 4-23 Verify Connectivity on R1

Other useful IPv6 verification commands include:

- **show interface**
- **show ipv6 routers**

Filter Show Command Output

Commands that generate multiple screens of output are, by default, paused after 24 lines. At the end of the paused output, the --More-- text displays. Pressing Enter displays the next line and pressing the spacebar displays the next set of lines. Use the **terminal length** *number* command to specify the number of lines to be displayed. A value of 0 (zero) prevents the router from pausing between screens of output.

Another very useful feature that improves the user experience in the command-line interface (CLI) is the filtering of **show** output. Filtering commands can be used to display specific sections of output. To enable the filtering command, enter a pipe (|) character after the **show** command and then enter a filtering parameter and a filtering expression.

The filtering parameters that can be configured after the pipe include:

- **section**: Shows entire section that starts with the filtering expression
- **include**: Includes all output lines that match the filtering expression
- **exclude**: Excludes all output lines that match the filtering expression
- **begin**: Shows all the output lines from a certain point, starting with the line that matches the filtering expression

Output filters can be used in combination with any **show** command.

Figures 4-24 through 4-27 provide examples of the various output filters. The example in Figure 4-24 uses the pipe character and the **section** keyword.

```
R1# show running-config | section line vty
line vty 0 4
 password 7 030752180500
 login
 transport input all
R1#
```

Figure 4-24 Filter **show** Commands by Section

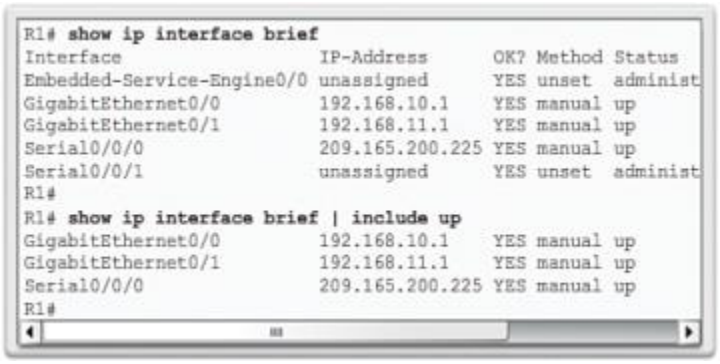


Figure 4-25 Filter **show** Commands by Common Keyword

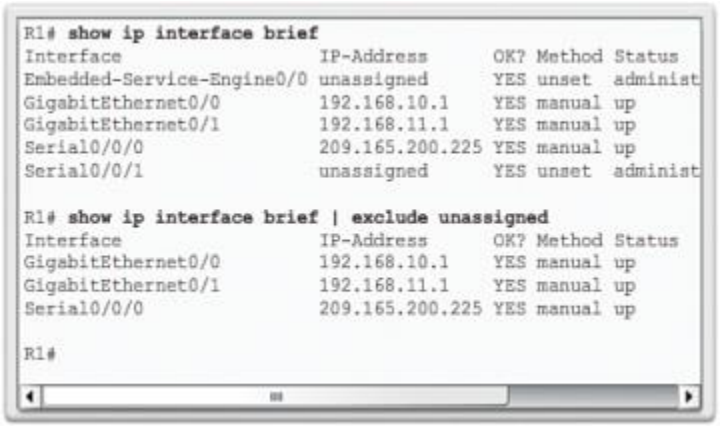


Figure 4-26 Filter **show** Commands to Exclude Rows of Output

Figure 4-27 Filter **show** Commands Beginning from a Keyword

The example in Figure 4-25 uses the pipe character and the **include** keyword.

Command History Feature

The command history feature is useful, because it temporarily stores the list of executed commands to be recalled.

To recall commands in the history buffer, press **Ctrl**+**P** or the **Up Arrow** key. The command output begins with the most recent command. Repeat the key sequence to recall successively older commands. To return to more recent commands in the history buffer, press **Ctrl**+**N** or the **Down Arrow** key. Repeat the key sequence to recall successively more recent commands.

By default, command history is enabled and the system captures the last 10 command lines in its history buffer. Use the **show history** privileged EXEC command to display the contents of the buffer.

It is also practical to increase the number of command lines that the history buffer records during the current terminal session only. Use the **terminal history size** user EXEC command to increase or decrease the size of the buffer.

For example, the following displays a sample of the **terminal history size** and **show history** commands:

```
R1# terminal history size 200
R1#
R1# show history
  show ip interface brief
  show interface g0/0
  show ip interface g0/1
  show ip route
  show ip route 209.165.200.224
  show running-config interface s0/0/0
  terminal history size 200
  show history
R1#
```

4. Routing Decisions

The key to understanding the role of a router in the network is to understand that a router is a Layer 3 device responsible for forwarding packets. However, a router also operates at Layers 1 and 2.

Router Switching Function

A primary function of a router is to forward packets toward their destination. This is accomplished by using a switching function, which is the process used by a router to accept a packet on one interface and forward it out of another interface. A key responsibility of the switching function is to encapsulate packets in the appropriate data link frame type for the outgoing data link.

In this context, the term "switching" literally means moving packets from source to destination and should not be confused with the function of a Layer 2 switch.

After the router has determined the exit interface using the path determination function, the router must encapsulate the packet into the data link frame of the outgoing interface.

What does a router do with a packet received from one network and destined for another network? The router performs the following three major steps:

- **Step 1.** De-encapsulates the Layer 3 packet by removing the Layer 2 frame header and trailer.
- **Step 2.** Examines the destination IP address of the IP packet to find the best path in the routing table.
- **Step 3.** If the router finds a path to the destination, it encapsulates the Layer 3 packet into a new Layer 2 frame and forwards the frame out the exit interface.

As shown in Figure 4-28, devices have Layer 3 IPv4 addresses and Ethernet interfaces have Layer 2 data link addresses. For example, PC1 is configured with IPv4 address 192.168.1.10 and an example MAC address of 0A-10. As a packet travels from the source device to the final destination device, the Layer 3 IP addresses do not change. However, the Layer 2 data link addresses change at every hop as the packet is de-encapsulated and re-encapsulated in a new frame by each router. It is very likely that the packet is encapsulated in a different type of Layer 2 frame than the one in which it was received. For example, an Ethernet encapsulated frame might be received by the router on a FastEthernet interface, and then processed to be forwarded out of a serial interface as a Point-to-Point Protocol (PPP) encapsulated frame.

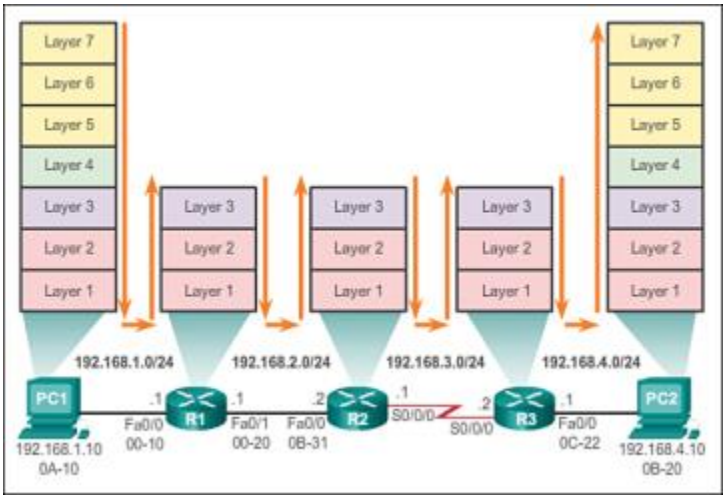


Figure 4-28 Encapsulating and De-Encapsulating Packets

Send a Packet

In the animation in the online course, PC1 is sending a packet to PC2.

PC1 must determine if the destination IPv4 address is on the same network. PC1 determines its own subnet by doing an **AND** operation on its own IPv4 address and subnet mask. This produces the network address that PC1 belongs to. Next, PC1 does this same **AND** operation using the packet destination IPv4 address and the PC1 subnet mask.

If the destination network address is the same network as PC1, then PC1 does not use the default gateway. Instead, PC1 refers to its ARP cache for the MAC address of the device with that destination IPv4 address. If the MAC address is not in the cache, then PC1 generates an ARP request to acquire the address to complete the packet and send it to the destination. If the destination network address is on a different network, then PC1 forwards the packet to its default gateway.

To determine the MAC address of the default gateway, PC1 checks its ARP table for the IPv4 address of the default gateway and its associated MAC address.

If an ARP entry does not exist in the ARP table for the default gateway, PC1 sends an ARP request. Router R1 sends back an ARP reply. PC1 can then forward the packet to the MAC address of the default gateway, the Fa0/0 interface of router R1.

A similar process is used for IPv6 packets. Instead of the ARP process, IPv6 address resolution uses ICMPv6 Neighbor Solicitation and Neighbor Advertisement messages. IPv6-to-MAC address mappings are kept in a table similar to the ARP cache, called the neighbor cache.

Forward to the Next Hop

The following processes take place when R1 receives the Ethernet frame from PC1:

1. R1 examines the destination MAC address, which matches the MAC address of the receiving interface, FastEthernet 0/0. R1, therefore, copies the frame into its buffer.
2. R1 identifies the Ethernet Type field as 0x800, which means that the Ethernet frame contains an IPv4 packet in the data portion of the frame.
3. R1 de-encapsulates the Ethernet frame.
4. Because the destination IPv4 address of the packet does not match any of the directly connected networks of R1, R1 consults its routing table to route this packet. R1 searches the routing table for a network address that would include the destination IPv4 address of the packet as a host address within that network. In this example, the routing table has a route for the 192.168.4.0/24 network. The destination IPv4 address of the packet is 192.168.4.10, which is a host IPv4 address on that network.

The route that R1 finds to the 192.168.4.0/24 network has a next-hop IPv4 address of 192.168.2.2 and an exit interface of FastEthernet 0/1. This means that the IPv4 packet is encapsulated in a new Ethernet frame with the destination MAC address of the IPv4 address of the next-hop router.

Because the exit interface is on an Ethernet network, R1 must resolve the next-hop IPv4 address with a destination MAC address using ARP:

1. R1 looks up the next-hop IPv4 address of 192.168.2.2 in its ARP cache. If the entry is not in the ARP cache, R1 would send an ARP request out of its FastEthernet 0/1 interface and R2 would send back an ARP reply. R1 would then update its ARP cache with an entry for 192.168.2.2 and the associated MAC address.
2. The IPv4 packet is now encapsulated into a new Ethernet frame and forwarded out the FastEthernet 0/1 interface of R1.

Packet Routing

The following processes take place when R2 receives the frame on its Fa0/0 interface:

1. R2 examines the destination MAC address, which matches the MAC address of the receiving interface, FastEthernet 0/0. R2, therefore, copies the frame into its buffer.
2. R2 identifies the Ethernet Type field as 0x800, which means that the Ethernet frame contains an IPv4 packet in the data portion of the frame.
3. R2 de-encapsulates the Ethernet frame.
4. Because the destination IPv4 address of the packet does not match any of the interface addresses of R2, R2 consults its routing table to route this packet. R2 searches the routing table for the destination IPv4 address of the packet using the same process R1 used.
5. The routing table of R2 has a route to the 192.168.4.0/24 network, with a next-hop IPv4 address of 192.168.3.2 and an exit interface of Serial 0/0/0. Because the exit interface is not an Ethernet network, R2 does not have to resolve the next-hop IPv4 address with a destination MAC address.
6. The IPv4 packet is now encapsulated into a new data link frame and sent out the Serial 0/0/0 exit interface.

When the interface is a point-to-point (P2P) serial connection, the router encapsulates the IPv4 packet into the proper data link frame format used by the exit interface (HDLC, PPP, etc.). Because there are no MAC addresses on serial interfaces, R2 sets the data link destination address to an equivalent of a broadcast (MAC address: FF:FF:FF:FF:FF:FF).

Reach the Destination

The following processes take place when the frame arrives at R3:

1. R3 copies the data link PPP frame into its buffer.
2. R3 de-encapsulates the data link PPP frame.
3. R3 searches the routing table for the destination IPv4 address of the packet. The routing table has a route to a directly connected network on R3. This means that the packet can be sent directly to the destination device and does not need to be sent to another router.

Because the exit interface is a directly connected Ethernet network, R3 must resolve the destination IPv4 address of the packet with a destination MAC address:

1. R3 searches for the destination IPv4 address of the packet in its Address Resolution Protocol (ARP) cache. If the entry is not in the ARP cache, R3 sends an ARP request out of its FastEthernet 0/0 interface. PC2 sends back an ARP reply with its MAC address. R3 then updates its ARP cache with an entry for 192.168.4.10 and the MAC address that is returned in the ARP reply.
2. The IPv4 packet is encapsulated into a new Ethernet data link frame and sent out the FastEthernet 0/0 interface of R3.
3. When PC2 receives the frame, it examines the destination MAC address, which matches the MAC address of the receiving interface, its Ethernet network interface card (NIC). PC2, therefore, copies the rest of the frame into its buffer.
4. PC2 identifies the Ethernet Type field as 0x800, which means that the Ethernet frame contains an IPv4 packet in the data portion of the frame.
5. PC2 de-encapsulates the Ethernet frame and passes the IPv4 packet to the IPv4 process of its operating system.

5.Path Determination

This section discusses the best path to send packets, load balancing, and the concept of administrative distance.

Routing Decisions

A primary function of a router is to determine the best path to use to send packets. To determine the best path, the router searches its routing table for a network address that matches the destination IP address of the packet.

The routing table search results in one of three path determinations:

- ***Directly connected network***: If the destination IP address of the packet belongs to a device on a network that is directly connected to one of the interfaces of the router, that packet is forwarded directly to the destination device. This means that the destination IP address of the packet is a host address on the same network as the interface of the router.

- ***Remote network***: If the destination IP address of the packet belongs to a remote network, then the packet is forwarded to another router. Remote networks can only be reached by forwarding packets to another router.
- **No route determined**: If the destination IP address of the packet does not belong to either a connected or remote network, the router determines if there is a Gateway of Last Resort available. A ***Gateway of Last Resort*** is set when a default route is configured on a router. If there is a default route, the packet is forwarded to the Gateway of Last Resort. If the router does not have a default route, then the packet is discarded. If the packet is discarded, the router sends an ICMP Unreachable message to the source IP address of the packet.

The logic flowchart in Figure 4-29 illustrates the router packet-forwarding decision process.

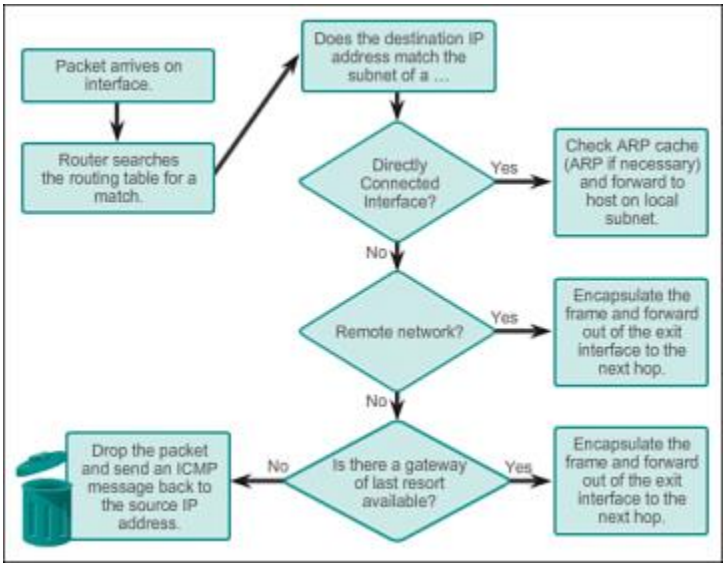


Figure 4-29 Packet Forwarding Decision Process

Best Path

Determining the best path involves the evaluation of multiple paths to the same destination network and selecting the optimum or shortest path to reach that network. Whenever multiple paths to the same network exist, each path uses a different exit interface on the router to reach that network.

The best route is selected by a routing protocol based on the value or metric it uses to determine the distance to reach a network. A ***metric*** is the quantitative value used to measure the distance to a given network. The ***best path*** to a network is the path with the lowest metric.

Dynamic routing protocols typically use their own rules and metrics to build and update routing tables. The routing algorithm generates a value, or a metric, for each path through the network. Metrics can be based on either a single characteristic or several characteristics of a path. Some routing protocols can base route selection on multiple metrics, combining them into a single metric.

The following lists some dynamic protocols and the metrics they use:

- **Routing Information Protocol (RIP)**: Hop count
- **Open Shortest Path First (OSPF)**: Cisco routers use a cost based on cumulative bandwidth from source to destination
- **Enhanced Interior Gateway Routing Protocol (EIGRP)**: Bandwidth, delay, load, reliability

Load Balancing

What happens if a routing table has two or more paths with identical metrics to the same destination network?

When a router has two or more paths to a destination with equal cost metrics, then the router forwards the packets using both paths equally. This is called ***equal cost load balancing.*** The routing table contains the single destination network, but has multiple exit interfaces, one for each equal cost path. The router forwards packets using the multiple exit interfaces listed in the routing table.

If configured correctly, load balancing can increase the effectiveness and performance of the network. Equal cost load balancing can be configured to use both dynamic routing protocols and static routes.

By default, Cisco routers can load balance up to four equal cost paths. The maximum number of equal cost paths depends on the routing protocol and IOS version.

EIGRP supports equal cost load balancing and is also the only routing protocol to support ***unequal cost load balancing***. Unequal cost load balancing is when a router distributes traffic over network interfaces, even those that are different distances from the destination address.

EIGRP supports unequal cost load balancing by using the `variance` command.

Administrative Distance

It is possible for a router to be configured with multiple routing protocols and static routes. If this occurs, the routing table may have more than one route source for the same destination network. For example, if both RIP and EIGRP are configured on a router, both routing protocols may learn of the same destination network. However, each routing protocol may decide on a different path to reach the destination based on that routing

protocol's metrics. RIP chooses a path based on hop count, whereas EIGRP chooses a path based on its composite metric. How does the router know which route to use?

Cisco IOS uses what is known as the ***administrative distance*** (AD) to determine the route to install into the IP routing table. The AD represents the "trustworthiness" of the route; the lower the AD, the more trustworthy the route source. For example, a static route has an AD of 1, whereas an EIGRP-discovered route has an AD of 90. Given two separate routes to the same destination, the router chooses the route with the lowest AD. When a router has the choice of a static route and an EIGRP route, the static route takes precedence. Similarly, a directly connected route with an AD of 0 takes precedence over a static route with an AD of 1.

Table 4-5 lists various routing protocols and their associated ADs.

Table 4-5 Default Administrative Distances

Route Source	Administrative Distance
Connected	0
Static	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
External EIGRP	170
Internal BGP	200
Unknown	255

Metric

If two routing updates for same network have same AD value then metric will use to choose the best path. Metric is a measurement to calculate best path. Route with the lowest metric will be chosen. Different routing protocols use different metrics. It may use single metric or multiple metrics. For example EIGRP uses bandwidth, delay, load, MTU and reliability while RIP only uses hop count as metric.

Routing Protocol	Metric	Description
EIGRP	Bandwidth	Capacity of link in Kbps
EIGRP	Delay	Time to reach in destination
EIGRP	Load	Path that is least utilize
EIGRP	MTU	Path that support largest frame size
EIGRP	Reliability	Path that have least down time

OSPF	Cost	Inverse of bandwidth links
RIP	Hop count	Hops (Routers) in the way of destination

6. Routing Protocols

There are three types of routing protocols:

1. Distance Vector
2. Link State
3. Hybrid

Distance Vector

Distance vector routing protocol uses distance (metric value) and direction (vector) to find the best path to destination network. Router receives routing update from neighboring router and these neighboring routers receive updates from their neighboring routers until the destination network. Every router in the way of destination network called hop. Each time a packet goes through a router, it add one in hop count value. Route with the least hop count value will be chosen as best path and will be placed in routing table. RIP is the example of distance vector routing protocol. These protocol shares entire routing table to the directly connected neighbors.

Link State

Link state routing protocols use more composite metric to locate the best path for destination network. It maintains three separate tables. First table keeps track of directly connected neighbors. Second table determines the entire network topology. Third is the routing table that keeps actual path. OSPF is the example of link state protocol. Link state protocols share their own links to all other routers in network.

Hybrid

Hybrid routing protocols are the mix of distance vector and link state protocol. To locate more accurate path, it uses aspect from both distance vector and link state. EIGRP is the example of hybrid routing protocols.

All three types of protocol have their own advantage and disadvantage. They take different approach in sharing routing updates and in choosing the best path. In next articles of this section we will explain these protocols in details with examples.

Summary

This chapter introduced the router. The main purpose of a router is to connect multiple networks and forward packets from one network to the next. This means that a router typically has multiple interfaces. Each interface is a member or host on a different IP network.

Cisco IOS uses what is known as the administrative distance (AD) to determine the route to install into the IP routing table. The routing table is a list of networks known by the router. The routing table includes network addresses for its own interfaces, which are the directly connected networks, as well as network addresses for remote networks. A remote network is a network that can only be reached by forwarding the packet to another router.

Remote networks are added to the routing table in one of two ways: either by the network administrator manually configuring static routes or by implementing a dynamic routing protocol. Static routes do not have as much overhead as dynamic routing protocols; however, static routes can require more maintenance if the topology is constantly changing or is unstable.

Dynamic routing protocols automatically adjust to changes without any intervention from the network administrator. Dynamic routing protocols require more CPU processing and also use a certain amount of link capacity for routing updates and messages. In many cases, a routing table will contain both static and dynamic routes.

Routers make their primary forwarding decision at Layer 3, the network layer. However, router interfaces participate in Layers 1, 2, and 3. Layer 3 IP packets are encapsulated into a Layer 2 data link frame and encoded into bits at Layer 1. Router interfaces participate in Layer 2 processes associated with their encapsulation. For example, an Ethernet interface on a router participates in the ARP process like other hosts on that LAN.

The Cisco IP routing table is not a flat database. The routing table is actually a hierarchical structure that is used to speed up the lookup process when locating routes and forwarding packets.

Components of the IPv6 routing table are very similar to the IPv4 routing table. For instance, it is populated using directly connected interfaces, static routes, and dynamically learned routes.

Reference:

- http://www.ciscopress.com/articles/article.asp?p=2180208
- http://www.cisco.com/c/en/us/td/docs/ios/12_2/configfun/command/reference/ffun_r/frf004.html