Chapter 5

Networking Devices

THE FOLLOWING COMPTIA NETWORK+ EXAM OBJECTIVES ARE COVERED IN THIS CHAPTER:

✓ **3.1 Install, configure and differentiate between common network devices**

- Hub
- Repeater
- Modem
- NIC
- Media converters
- Basic switch
- Bridge
- Wireless access point
- Basic router
- Basic firewall
- Basic DHCP server

✓ **3.2 Identify the functions of specialized network devices**

- Multilayer switch
- Content switch
- IDS/IPS
- Load balancer
- Multifunction network devices
- DNS server
- Bandwidth shaper
- Proxy server
- CSU/DSU

In this chapter, I'll tell you all about the networking devices I've introduced so far. I'll go into much greater detail about each device, and yes—I'm going to present even more of them to you! Because all the components that you'll learn about shortly are typically found in today's networks and internetworks, it's very important that you be familiar with them.

We'll start by covering the more common network devices that you would be most likely to come across, and then move on to discuss some of the more specialized devices that you may or may not always find running in a network.

I'll finish the chapter by using examples to discuss how routers, hubs, and switches work within internetworks today.

NOTE For up-to-the-minute updates for this chapter, please see `www.lammle.com` or `www.sybex.com/go/comptianetwork+studyguide`.

Common Network Connectivity Devices

Okay—by now, you should be fairly savvy regarding the various types of network media and connections, so it's time to learn about some of the devices they hook up to that are commonly found on today's networks.

First, I'll define the basic terms; then, later in this chapter, I'll show you how these devices actually work within a network. At that time, I'll give you more detailed descriptions of these devices and terms used along with them.

Because these devices connect network entities, they're known as *connectivity devices*. Here's a list of the devices I'll be covering in this chapter:

- Hub
- Repeater
- Modem
- Network Interface Card (NIC)
- Transceiver (media converter)
- Bridge
- Basic switch
- Wireless access point (AP)

- Basic router
- Basic firewall
- Basic Dynamic Host Configuration Protocol (DHCP) server
- Other specialized devices

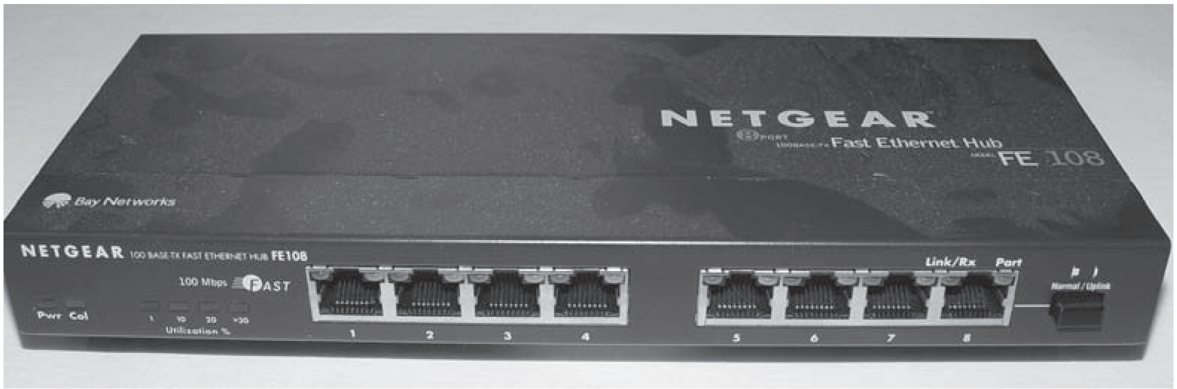
Hub

As you learned earlier, a *hub* is the device that connects all the segments of the network together in a star topology Ethernet network. Every device in the network connects directly to the hub through a single cable and is used to connect multiple devices without segmenting a network. Any transmission received on one port will be sent out all the other ports in the hub, including the receiving pair for the transmitting device, so that Carrier Sense Multiple Access with Collision Detection (CSMA/CD) on the transmitter can monitor for collisions.

So, basically, this means that if one station sends a broadcast, all the others will receive it; yet based on the addressing found in the frame, only the intended recipient will actually listen to it. This arrangement simulates the physical bus that the CSMA/CD standard was based on, and it's why we call the use of a hub in an Ethernet environment a physical star/logical bus topology.

Figure 5.1 depicts a typical hub as you might find it employed within a home network. Most of the time, hubs really aren't recommended for corporate networks because of their limitations.

FIGURE 5.1 A typical hub



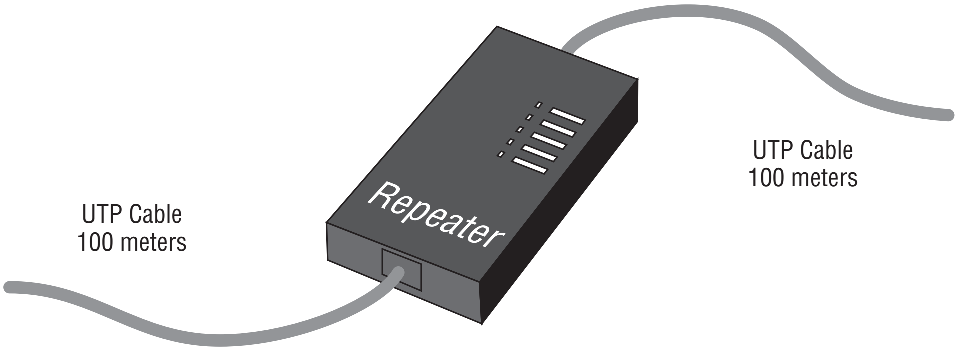
It's important to note that hubs are nothing more than glorified repeaters that are incapable of recognizing frames and data structures—the reason why they act with such a lack of intelligence. A broadcast sent out by any device on the hub will be propagated to all devices connected to it. And just as in a physical bus topology configuration, any two or more of those connected devices have the potential of causing a collision with each other, which means that this hardware device will create a LAN with the most network traffic collisions. Hubs are not suggested in today's corporate network for this reason.

Repeater

Most of the time, repeaters were used in the old Thinnet networks of yesteryear. Today, they're just employed as the multi-port repeaters that we call hubs.

But there's another way we currently use them—Figure 5.2 shows a repeater being used to connect two unshielded twisted-pair (UTP) connectors. This configuration will provide an extension to your Ethernet segment and give you a gain of another 100 meters (328 feet).

FIGURE 5.2 Ethernet repeater



I really don't recommend using a repeater in networks because of the latency involved, but it can be a good thing if you employ one in a very limited role. Even so, I'd personally go with using a wireless network for a solution in a long-distance connection instead of a repeater because it will provide you with good distance without losing bandwidth or adding latency. In other words, just say no—repeaters and hubs shouldn't be used in today's networks because there are better solutions available!

NOTE It is important to remember that both hubs and repeaters are layer-1 devices and do not segment a network in any way.

Modem

You're probably (ummm—I hope) familiar with modems, but you may not be aware of their formal description. A *modem* is a device that modulates digital data onto an analog carrier for transmission over an analog medium and then demodulates from the analog carrier to a digital signal again at the receiving end. A mouthful, yes, but the term *modem* is actually an acronym that stands for MOdulator/DEModulator.

When you hear the term *modem*, three different types should come to mind:

- Traditional (plain old telephone service [POTS])
- DSL
- Cable

Traditional (POTS)

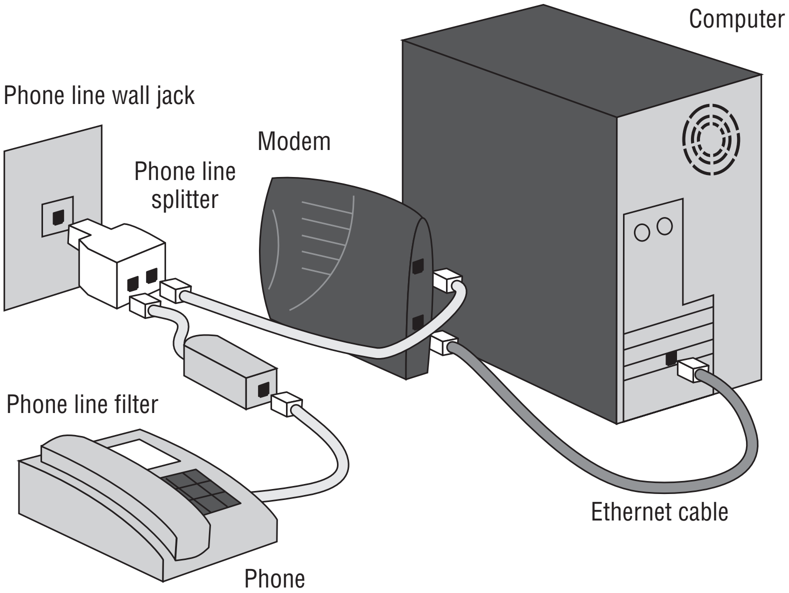
Most modems you find in computers today fall into the category of traditional modems. These modems convert the signals from your computer into those that travel over plain old telephone service (POTS) lines. The majority of modems that exist today are POTS modems, mainly because PC manufacturers include one with the computer, built right into the motherboard.

DSL

Digital subscriber line (DSL) has replaced traditional modem access because it offers higher data throughput rates for a reasonable cost. Plus, you get to make regular, land-line phone calls while online. DSL uses higher frequencies (above 3200Hz) than regular voice phone calls use, which provides greater bandwidth than regular POTS modems—up to several megabits per second. And it does so while still allowing standard voice data to travel in its normal frequency range and remain compatible with traditional POTS phones and devices.

DSL “modems” are the devices that allow the network signals to pass over phone lines on these higher frequencies. Check out Figure 5.3.

FIGURE 5.3 A typical DSL modem setup



Usually, when you sign up for DSL service, the company you sign up with will send you a DSL modem for free or pretty close to it. This modem is generally an external one (although internal DSL modems are available), and it usually has both a phone line and an Ethernet connection. You have to connect the phone line to a wall jack and the Ethernet connection to your computer (shown in Figure 5.3), so it follows that you need to have an Ethernet NIC in your computer to connect to the DSL modem. Sometimes a router, hub, or switch is connected to the Ethernet port of the DSL modem, increasing the options for your Ethernet network.

If you have DSL service on the same phone line you use to make voice calls, you must install DSL filters on all the phone jacks where you have a phone (again, shown in Figure 5.3). Or, DSL filters may be installed after the DSL modem for all the phones in a building. Unless, of course, you can put up with an overwhelmingly annoying hissing noise (the DSL signals) on your voice calls!

Cable

Another popular high-speed Internet-access technology is cable-modem access. Cable modems connect an individual PC or network to the Internet using your television cable. The cable TV companies use their existing cable infrastructure to deliver data services on unused frequency bands.

The cable modem itself is a fairly simple device. It has a standard coax connector on the back as well as an Ethernet port. You can connect one PC to a cable modem—again, this requires that your computer have an Ethernet NIC installed—or you can connect the modem to multiple PCs on a network using a hub or switch. And you can always use a router to enhance your Ethernet network's capabilities.

Network Interface Card (NIC)

Those of you who aren't familiar with NICs probably want to be, at this point, so here goes: a *Network Interface Card* (NIC) is installed in your computer to connect, or interface, your computer to the network. It provides the physical, electrical, and electronic connections to the network media.

A NIC either is an expansion card or is built right into the computer's motherboard. The NIC usually connects to the computer through *expansion slots* located on the motherboard that allow peripherals to be plugged in directly. In some notebook computers, NIC adapters can be connected to the printer port or through a PC card slot.

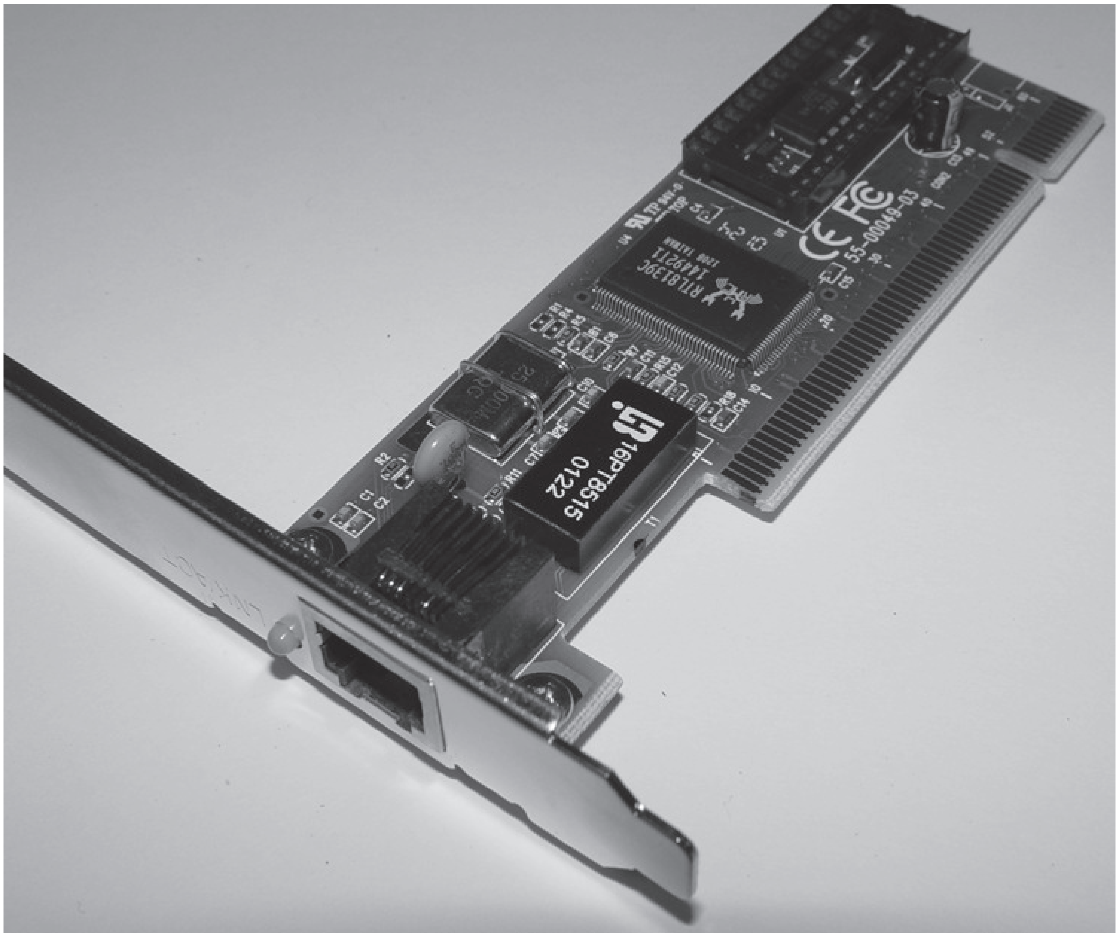
Figure 5.4 shows a typical 100Mbps Ethernet NIC.

Nowadays, most PC's and laptops of all types come with an Ethernet connector built into the motherboard, so you usually don't need a separate card. It's also rare to find a laptop today without a built-in wireless network card, but you can buy external wireless cards for desktops and laptops if you've got legacy equipment that needs them.

NIC cards generally have one or two light-emitting diodes (LEDs) that help in diagnosing any functional problems. If there are two separate LEDs on the NIC, one of them is most likely the Link LED, which illuminates when proper connectivity to an active network has been detected. But it's not always that cut and dried—that blinking LED can mean the NIC is receiving a proper signal from the hub or switch, but it can also indicate connectivity to and detection of a carrier on a segment. Another possibility is that it's found connectivity with a router or other end device using a crossover cable.

The other LED is the aptly named Activity LED, which tends to flicker constantly. That activity indicates the intermittent transmission and reception of frames arriving at the network or leaving it.

FIGURE 5.4 Network Interface Card



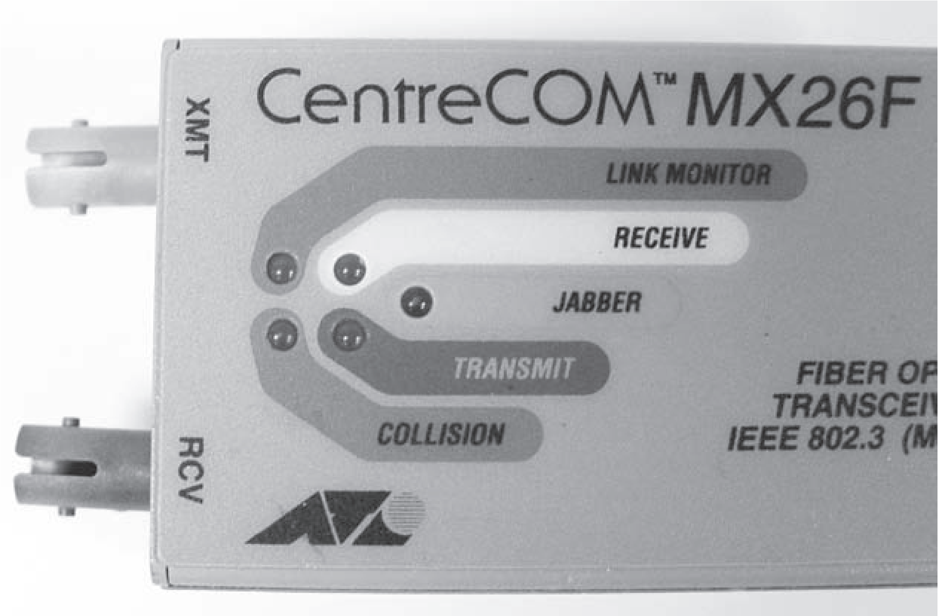
The first LED you should verify is the Link LED because if it's not illuminated, the Activity LED simply cannot illuminate.

Transceiver (Media Converter)

Another small device that you might come across on a network is an external transceiver, otherwise known as a media converter. These simple devices allow a NIC or other networking device to connect to a different type of media than it was designed to connect to. Many hubs, switches, and NICs have special connectors that allow for this.

For instance, let's say you've got a 100Base-TX switch, and you want to connect it to another switch using fiber-optic cabling. To make this happen, you need to connect a fiber transceiver to each switch's transceiver port and then connect the two transceivers together with the appropriate fiber-optic cabling. Figure 5.5 shows an Ethernet-to-fiber transceiver.

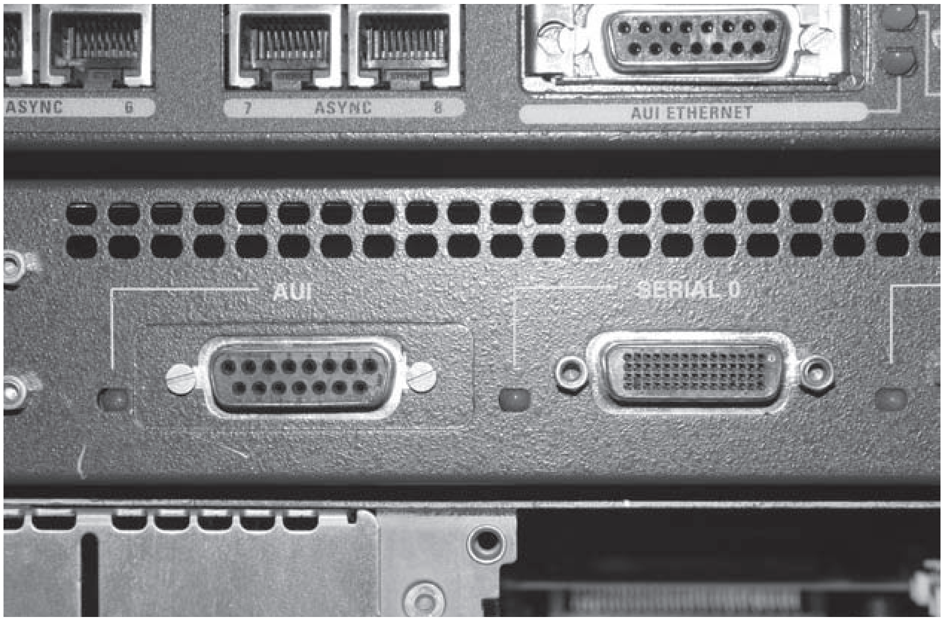
FIGURE 5.5 A 100Base-TX to 100Base-FX transceiver



With early Ethernet-style DB-15 female Digital, Intel, and Xerox (DIX) connectors—often referred to as Attachment Unit Interface (AUI) connectors—NIC interfaces are still available as medium-independent connectors on more advanced NICs and other networking devices. But you'll need an external transceiver to convert the electrical signal from the device to one that's compatible with the cabling medium.

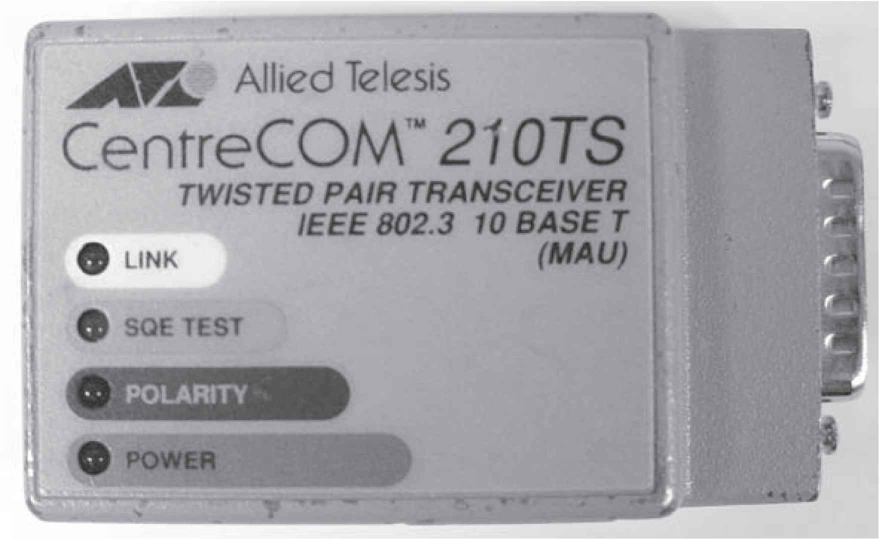
Figure 5.6 shows a router with a DIX (AUI) connector.

FIGURE 5.6 Router with a DIX (AUI) connector



All *x*Base-T standards, and all other popular types of Ethernet technology, have a built-in transceiver (which transceives digital data) on the NIC card or device interface. With these technologies, an external transceiver is only required to act as a media converter, as shown in Figure 5.7.

FIGURE 5.7 DIX to RJ-45 transceiver

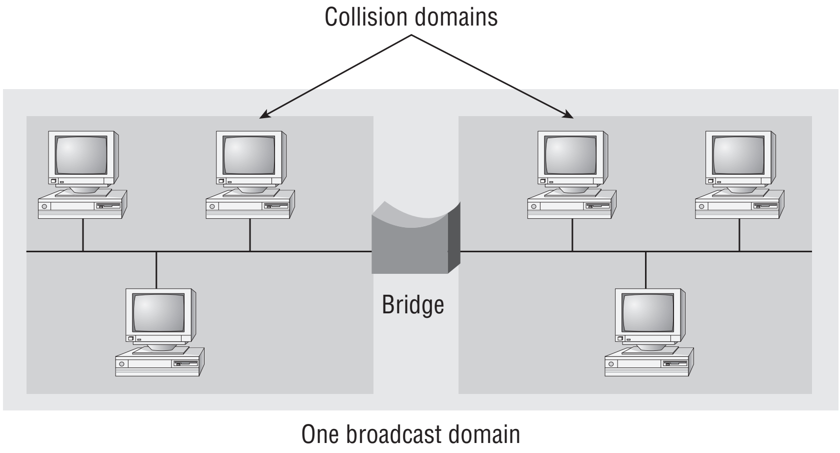


I'm surprised when I find these in today's networks; but believe it or not, I use them myself at times with some routers. This is the reason that you still need to know about them.

Bridge

A *bridge*—specifically, a transparent bridge—is a network device that connects two similar network segments together. Its primary function is to keep traffic separated on either side of the bridge, breaking up collision domains, as pictured in Figure 5.8.

FIGURE 5.8 Bridges break up collision domains.



What we can see here is that traffic is allowed to pass through the bridge only if the transmission is intended for a station on the opposite side. The main reasons you would place a bridge in your network would be to connect two segments together or to divide a busy network into two segments.

Bridges are software based; so, interestingly, you can think of a switch as a hardware-based, multiport bridge. In fact, the terms, *bridge* and *switch* are often used interchangeably because the two devices used basically the same bridging technologies. The past tense is there for a reason—you'd be hard-pressed to buy a bridge today.

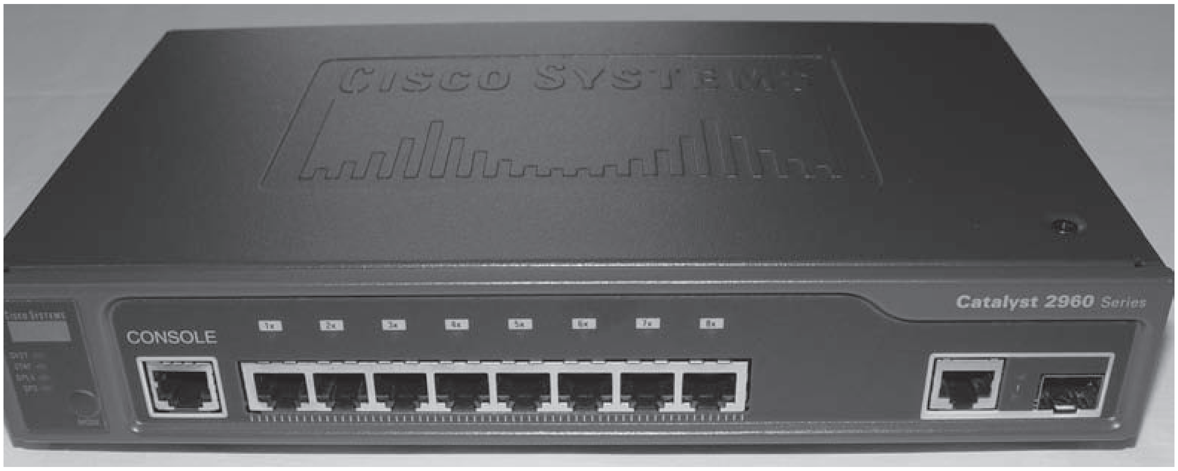
Switch

Switches connect multiple segments of a network together much like hubs do, but with three significant differences—a switch recognizes frames and pays attention to the source and destination MAC address of the incoming frame as well as the port on which it was received. Hubs don't do those things. They simply send out anything they receive on one port out to all the others.

So, if a switch determines that a frame's final destination happens to be on a segment that's connected via a different port than the one on which the frame was received, the switch will only forward the frame out from the specific port on which its destination is located. If the switch can't figure out the location of the frame's destination, it will flood the frame out every port except the one on which the frame port was received.

Figure 5.9 shows a typical low-cost Ethernet switch. It looks a lot like a hub. However, switches can come in very large, expensive sizes.

FIGURE 5.9 Typical Ethernet switch



That's as far as we're going with switches right now. I'll bring them up later on in this chapter and cover them in much greater detail in Chapter 11, "Switching and Virtual LANs (VLANs)." For now, you can think of a switch as a faster, smarter bridge that has more ports.

NOTE

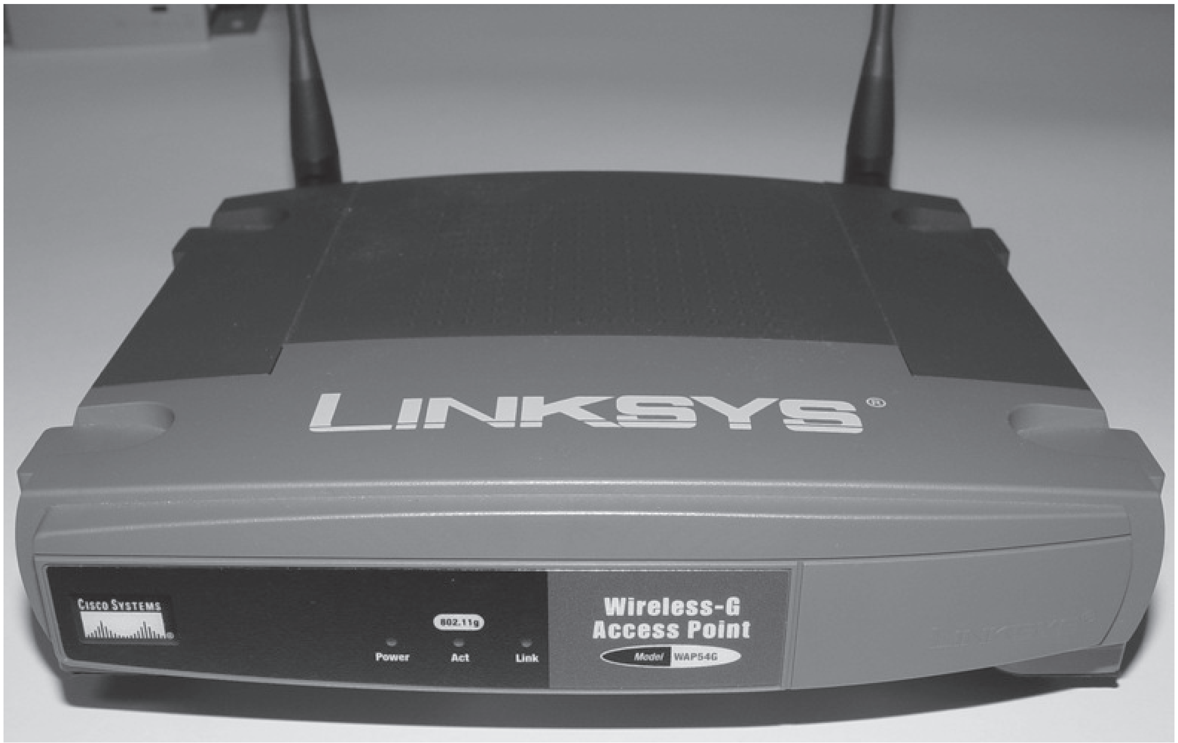
Switches are a Layer 2 device, which means they segment the network with MAC addresses. If you see the term “Layer 3 switch”, that means you are talking about a router, not a Layer 2 switch. The terms router and Layer 3 switch are interchangeable.

Wireless Access Point (AP)

A *wireless access point (AP)* allows mobile users to connect to a wired network wirelessly via radio frequency technologies. Using wireless technologies, APs also allow wired networks to connect to each other and are basically the wireless equivalent of hubs or switches because they can connect multiple wireless (and often wired) devices together to form a network.

Figure 5.10 shows a typical low-cost access point

FIGURE 5.10 A typical low-cost access point



One of the most popular uses for APs today is to provide Internet access in public areas like libraries, coffee shops, hotels, and airports. You may think WAPs are hard to set up, but they’re not—basically, you just need to plug them in to a wired network, power them up, and—voila! Another big plus is that without the clutter and added expense of cables, WAPs make ideal foundations for small business networks.

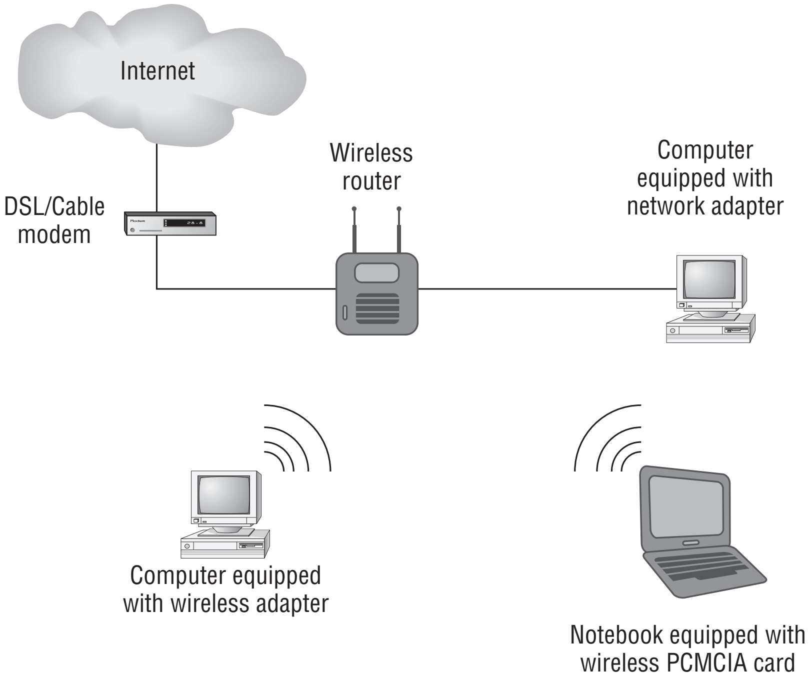
NOTE You'll learn all the critical details a Network+ technician needs to know about APs later, in Chapter 12, "Wireless Technologies."

Router

A *router* is a network device used to connect many, sometimes disparate, network segments together, combining them into what we call an *internetwork*. A well-configured router can make intelligent decisions about the best way to get network data to its destination. It gathers the information it needs to make these decisions based on a network's particular performance data.

Figure 5.11 shows a Small Office, Home Office (SOHO) router that provides wired and wireless access for hosts and connects them to the Internet without any necessary configuration. But know that I certainly don't recommend leaving a router with the default configuration! No worries, though—I'll go over the configuration process with you in Chapter 12.

FIGURE 5.11 Router connected to the Internet, providing access for hosts


Internet
Wireless router
Computer equipped with network adapter
DSL/Cable modem
Computer equipped with wireless adapter
Notebook equipped with wireless PCMCIA card

Routers can be multifaceted devices that behave like computers unto themselves with their own complex operating systems—for example, Cisco's IOS. You can even think of them as CPUs that are totally dedicated to the process of routing packets. And due to their complexity and flexibility, you can configure them to actually perform the functions of other types of network devices (like firewalls, for example) by simply implementing a specific feature within the router's software.

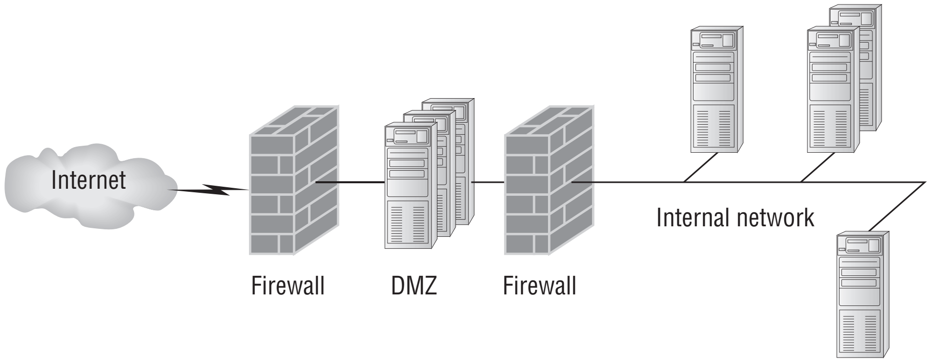
NOTE Routers can have many different names: layer-3 switch and multilayer switch are the most common, besides the name router, of course. Remember, if you just hear the word "switch", that means a layer-2 device. Routers, layer-3 switches, and multilayer switches are all layer-3 devices.

Firewall

So what, exactly, is a *firewall*? Basically, firewalls are your network's security guards; and to be real, they're probably the most important thing to implement on your network. That's because today's networks are almost always connected to the Internet—a situation that makes security crucial! A firewall protects your LAN resources from invaders that prowl the Internet for unprotected networks, while simultaneously preventing all or some of your LAN's computers from accessing certain services on the Internet. You can employ them to filter packets based on rules that you or the network administrator create and configure to strictly delimit the type of information allowed to flow in and out of the network's Internet connection.

A firewall can be either a stand-alone "black box" or a software implementation placed on a server or router. Either way, the firewall will have at least two network connections: one to the Internet (known as the *public* side) and one to the network (known as the *private* side). Sometimes, there is a second firewall, as shown in Figure 5.12. This firewall is used to connect servers and equipment that can be considered both public and private (like web and email servers). This intermediary network is known as a *demilitarized zone (DMZ)*.

FIGURE 5.12 Example of firewalls with a DMZ



Firewalls are the first line of defense for an Internet-connected network. Without them in place, any network that's connected to the Internet is essentially wide open to anyone with a little technical savvy who seeks to exploit LAN resources and/or access your network's sensitive information.

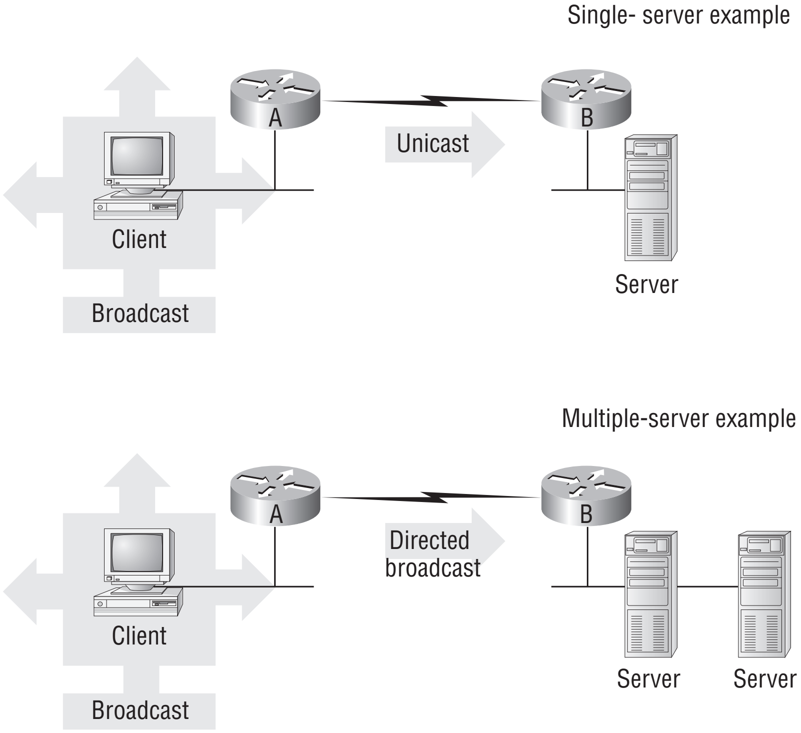
Dynamic Host Configuration Protocol (DHCP) Server

Even though I'm going to get into the finer points of DHCP soon, in Chapter 6, "Introduction to Internet Protocol (IP)," I want to give you some basic insight into this server service here.

In essence, DHCP servers assign IP addresses to hosts. This protocol gives us a much easier way to administrate—by automatically providing IP information—than the alternative and tedious method known as static IP addressing, where we have to address each host manually. It works well in any network environment, from tiny to huge, and allows all types of hardware to be employed as a DHCP server, including routers.

It works like this: A DHCP server receives request for IP information from a DHCP client using a broadcast (as Chapter 6 will show you in detail). The only hitch is that if the DHCP server isn't on the same segment as the DHCP client, the broadcast won't be received by the server because by default, routers won't forward broadcasts, as shown in Figure 5.13.

FIGURE 5.13 DHCP client sends broadcasts looking for a DHCP server



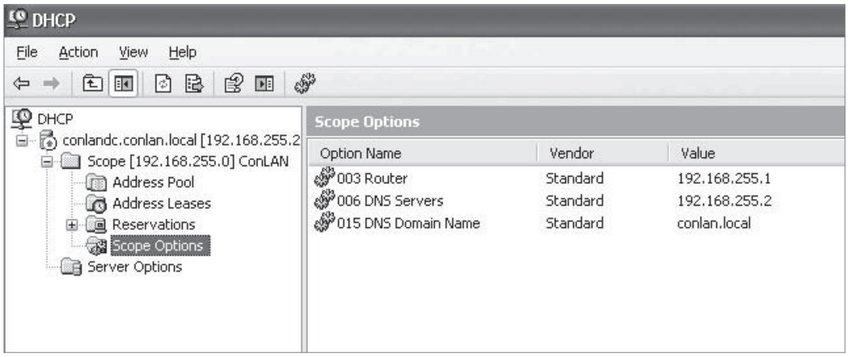
In Figure 5.13, Router A is configured with the IP helper address command on interface E0 of the router. Whenever interface E0 receives a broadcast request, Router A will forward those requests as a unicast (meaning instead of a broadcast, the packet now has the destination IP address of the DHCP server).

So, as shown in the figure, you can configure Router A to forward these requests and even use multiple DHCP servers for redundancy, if needed. This works because the router has been configured to forward the request to a single server using a unicast or by sending the request to multiple servers via a directed broadcast.

Personally, most of the time I use a Windows server to act as the DHCP server for my entire internetwork and have my routers forward client requests. It is possible to have a DHCP server on every network segment, but that is not necessary because of the routers' forwarding ability.

Figure 5.14 shows a picture of a Windows server with something called Scope Options.

FIGURE 5.14 A Windows DHCP server's Scope Options

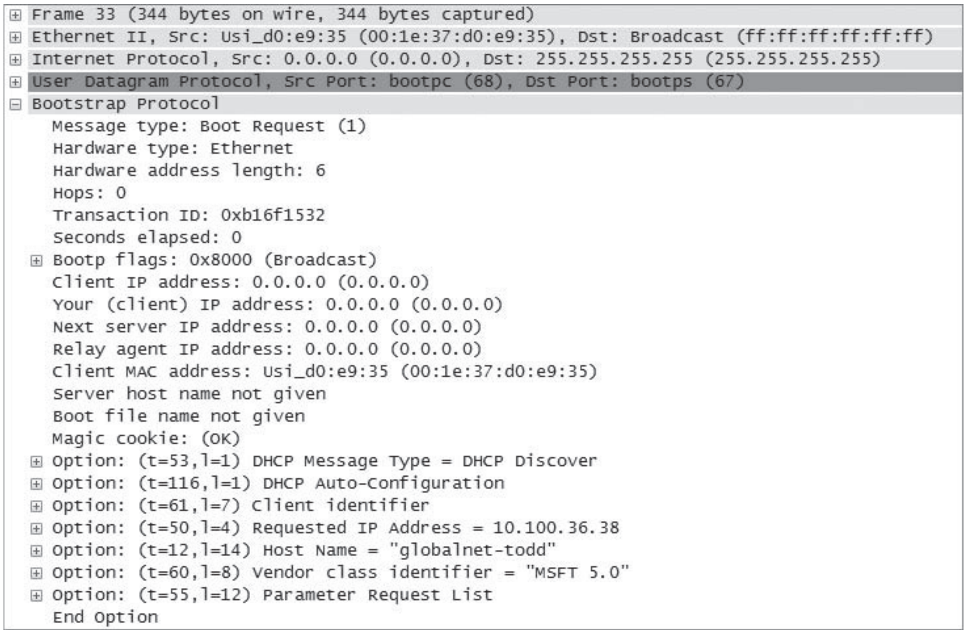


Option Name	Vendor	Value
003 Router	Standard	192.168.255.1
006 DNS Servers	Standard	192.168.255.2
015 DNS Domain Name	Standard	conlan.local

Scope Options provide IP configuration for hosts on a specific subnet. Below the Scope Options, you'll find Server Options, which provide IP information for all scopes configured on the server. If I had just one Domain Name Service (DNS) server for the entire network, I'd configure the Server Options with my DNS server information; that DNS server information would then show up automatically in all scopes configured on my server.

So, what exactly does a DHCP client ask for, and what does a DHCP server provide? Is it just an IP address, a mask, and a default gateway? No, it is much more than that. Let's take a look at a DHCP client request on an analyzer. Figure 5.15 shows the options that the client is requesting from the DHCP server.

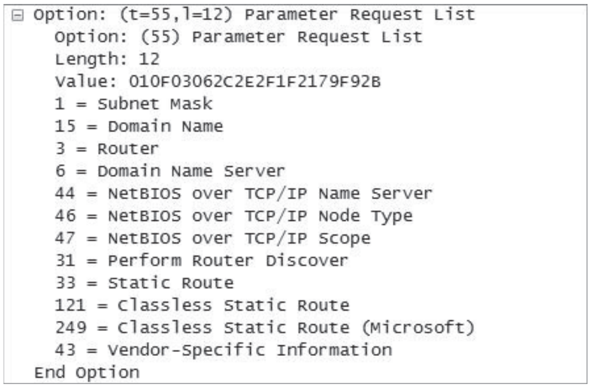
FIGURE 5.15 DHCP client request to a DHCP server


```
⊞ Frame 33 (344 bytes on wire, 344 bytes captured)
⊞ Ethernet II, Src: Usi_d0:e9:35 (00:1e:37:d0:e9:35), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊞ Internet Protocol, Src: 0.0.0.0 (0.0.0.0), Dst: 255.255.255.255 (255.255.255.255)
⊞ User Datagram Protocol, Src Port: bootpc (68), Dst Port: bootps (67)
⊟ Bootstrap Protocol
    Message type: Boot Request (1)
    Hardware type: Ethernet
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0xb16f1532
    Seconds elapsed: 0
  ⊞ Bootp flags: 0x8000 (Broadcast)
    Client IP address: 0.0.0.0 (0.0.0.0)
    Your (client) IP address: 0.0.0.0 (0.0.0.0)
    Next server IP address: 0.0.0.0 (0.0.0.0)
    Relay agent IP address: 0.0.0.0 (0.0.0.0)
    Client MAC address: Usi_d0:e9:35 (00:1e:37:d0:e9:35)
    Server host name not given
    Boot file name not given
    Magic cookie: (OK)
  ⊞ Option: (t=53,l=1) DHCP Message Type = DHCP Discover
  ⊞ Option: (t=116,l=1) DHCP Auto-Configuration
  ⊞ Option: (t=61,l=7) Client identifier
  ⊞ Option: (t=50,l=4) Requested IP Address = 10.100.36.38
  ⊞ Option: (t=12,l=14) Host Name = "globalnet-todd"
  ⊞ Option: (t=60,l=8) Vendor class identifier = "MSFT 5.0"
  ⊞ Option: (t=55,l=12) Parameter Request List
    End Option
```

First, you can see that the DHCP service runs on top of the bootP protocol (port 68) and that the DHCP client is looking for a bootp server (port 67). The client IP address is 0.0.0.0, and the client doesn't know the DHCP server address either because this is a broadcast to 255.255.255.255 (the Data Link layer broadcast shows ff:ff:ff:ff:ff:ff). Basically, all the DHCP client knows for sure is its own MAC address.

The DHCP client Parameter Request List option shown at the end of Figure 5.15 has been expanded and is shown in Figure 5.16. The client is "requesting" a certain IP address because this is the IP address it received from the server the last time it requested an IP address.

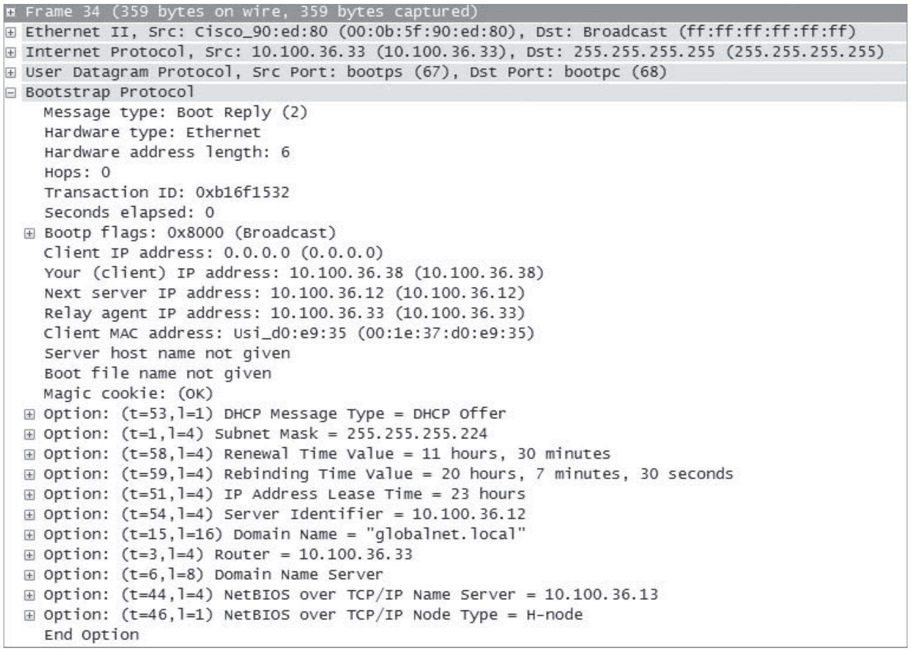
FIGURE 5.16 DHCP client parameter request list



```
⊟ Option: (t=55,l=12) Parameter Request List
    Option: (55) Parameter Request List
    Length: 12
    Value: 010F03062C2E2F1F2179F92B
    1 = Subnet Mask
    15 = Domain Name
    3 = Router
    6 = Domain Name Server
    44 = NetBIOS over TCP/IP Name Server
    46 = NetBIOS over TCP/IP Node Type
    47 = NetBIOS over TCP/IP Scope
    31 = Perform Router Discover
    33 = Static Route
    121 = Classless Static Route
    249 = Classless Static Route (Microsoft)
    43 = Vendor-Specific Information
  End Option
```

That is quite a request list! The DHCP server will respond with the options that it has configured and available to provide to a DHCP client. Let's take a look and see what the server responds with. Figure 5.17 shows the DHCP server response.

FIGURE 5.17 DHCP server response



```
⊞ Frame 34 (359 bytes on wire, 359 bytes captured)
⊞ Ethernet II, Src: Cisco_90:ed:80 (00:0b:5f:90:ed:80), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊞ Internet Protocol, Src: 10.100.36.33 (10.100.36.33), Dst: 255.255.255.255 (255.255.255.255)
⊞ User Datagram Protocol, Src Port: bootps (67), Dst Port: bootpc (68)
⊟ Bootstrap Protocol
    Message type: Boot Reply (2)
    Hardware type: Ethernet
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0xb16f1532
    Seconds elapsed: 0
  ⊞ Bootp flags: 0x8000 (Broadcast)
    Client IP address: 0.0.0.0 (0.0.0.0)
    Your (client) IP address: 10.100.36.38 (10.100.36.38)
    Next server IP address: 10.100.36.12 (10.100.36.12)
    Relay agent IP address: 10.100.36.33 (10.100.36.33)
    Client MAC address: Usi_d0:e9:35 (00:1e:37:d0:e9:35)
    Server host name not given
    Boot file name not given
    Magic cookie: (OK)
  ⊞ Option: (t=53,l=1) DHCP Message Type = DHCP Offer
  ⊞ Option: (t=1,l=4) Subnet Mask = 255.255.255.224
  ⊞ Option: (t=58,l=4) Renewal Time Value = 11 hours, 30 minutes
  ⊞ Option: (t=59,l=4) Rebinding Time Value = 20 hours, 7 minutes, 30 seconds
  ⊞ Option: (t=51,l=4) IP Address Lease Time = 23 hours
  ⊞ Option: (t=54,l=4) Server Identifier = 10.100.36.12
  ⊞ Option: (t=15,l=16) Domain Name = "globalnet.local"
  ⊞ Option: (t=3,l=4) Router = 10.100.36.33
  ⊞ Option: (t=6,l=8) Domain Name Server
  ⊞ Option: (t=44,l=4) NetBIOS over TCP/IP Name Server = 10.100.36.13
  ⊞ Option: (t=46,l=1) NetBIOS over TCP/IP Node Type = H-node
    End Option
```

The client is going to get the IP address that it asked for (10.100.36.38), a subnet mask of 255.255.255.224, a lease time of 23 hours (the amount of time before the IP address and other DHCP information expires on the client), the IP address of the DHCP server, the default gateway (router), the DNS server IP address (it gets two), the domain name used by DNS, and some NetBIOS information (used by Windows for name resolution).

The lease time is important and can even be used to tell you if you have a DHCP problem, or more specifically, that the DHCP server is no longer handing out IP addresses to hosts. If hosts start failing to get onto the network one at a time as they try to get a new IP address as their lease time expires, you need to check your server settings.

Here is another example of a possible DHCP problem: you arrive at work after a weekend where some hosts were left on and some were shut down. The hosts that were left running and not shut down are still working, but the hosts that were shut down and were restarted on Monday morning do not get a new IP address. This is a good indication that you need to head over to your DHCP server and take a look at what is going on.

A DHCP server can also be configured with a reservation list so that a host always receives the same IP address. You would use this reservation list for routers or servers if they were not statically assigned. However, you can use reservation lists for any host on your network as well.

Other Specialized Devices

In addition to the network connectivity devices I've discussed with you, there are several devices that, while maybe not directly connected to a network, do actively participate in moving network data. Here's a list of them:

- Multilayer switch
- Content switch
- Intrusion Detection or Prevention System (IDS/IPS)
- Load balancer
- Multifunction network devices
- DNS server
- Bandwidth shaper
- Proxy server
- Channel Service Unit/Data Service Unit (CSU/DSU)

Multilayer Switch

A *multilayer switch* (MLS) is a computer networking device that switches on Open Systems Interconnection (OSI) Layer 2 like an ordinary network switch but provides extra functions on higher OSI layers, like Layer 3, for routing.

The major difference between the packet-switching operation of a router and that of a Layer 3 or multilayer switch lies in the physical implementation. In routers, packet switching takes place using a microprocessor, whereas a Layer 3 switch handles this by using application-specific integrated circuit (ASIC) hardware. I'd show you a picture of a Layer 3 switch, but they look just like regular Layer 2 switches and you already know what those look like. The differences are the hardware inside and the operating system.

Content Switch

Believe it or not, we have switches around today that are capable of utilizing up to OSI Layer 7 information. Here's a list of these cool devices:

- Layer 4–7 switches
- Content switches
- Content services switches
- Web switches
- Application switches

We use the power given us by content switches for something known as *load balancing* within a whole group of servers. It really comes in handy dealing with things like application Transmission Control Protocol/Internet Protocol (TCP/IP) data, HTTP, HTTPS, and/or a virtual private network (VPN) concerning a particular port.

A good point to remember is that load balancing frequently requires Network Address Translation (NAT), which I'll talk about in Chapter 8, "IP Subnetting, Troubleshooting IP, and Introduction to NAT." Client machines connected to load-balanced services are totally in the dark about the specific server that's responding to the client. What's more, some Layer 4–7 switches are so fast they actually execute NAT at wire-speed rates that make it look as though the switch isn't even on the network because the latency, or response time, is so low, it's basically absent.

You can also use content switches for important tasks like encryption and decryption, as well as for streamlining the administration of digital certificates (these are all important terms you'll learn about in Chapters 13, "Authentication and Access Control"; in 14, "Network Threats and Mitigation"; and in 15, "Physical and Hardware Security"). The capabilities of these switches greatly minimize the load on any servers receiving network traffic and really enhance network performance.

Intrusion Detection or Prevention System (IDS/IPS)

Intrusion Detection System (IDS) is exactly what it sounds like—a powerful security tool that detects a plethora of nasty tactics that bad guys use to exploit systems, including unauthorized logins and privilege increases that can give them access to your sensitive data and files. Attacks on network resources, services, and applications—even the vile practice of placing viruses, worms, and trojans—are also detected by IDS. However, IDS only identifies, detects, and reports attempts of unauthorized access to the network as well as any suspicious activity, and is the best software type for identifying an attack. However, if you want to stop the attack in its track you need to add an IPS device.

An *Intrusion Prevention System (IPS)* provides computers with security by vigilantly watching for any suspicious and potentially malicious tactics. It works in real time and, as its name suggests, prevents these evil activities. For instance, network-based IPS monitors the network's traffic, looking for malicious code and other attacks and simply drops any offensive packets while permitting all proper network traffic to flow unimpeded. So, unlike IDS, which can identify an attack and report it, IPS can stop the attack in its tracks by shutting down a port or dropping certain types of packets.

Load Balancer

Your average router just sends incoming packets to their specified, correlative IP address on the network; but a *load balancer* can actually send incoming packets to multiple machines hidden behind one IP address—cool, right?

Today's load-balancing routers follow various rules to determine specifically how they will route network traffic. Depending on your needs, you can set rules based on the least load, fault tolerance, the fastest response times, or just dividing up (balancing) outbound requests for smooth network operations.

In fact, the fault tolerance, or redundancy, as well as the scalability so vital to large networking environments and e-commerce include some of the great benefits we gain using load balancers.

Think about this scenario: Say you have a web site where people are placing orders for the stuff you've got for sale. Obviously, the orders placed vary in size and the rate at which that they come in; and you definitely wouldn't want your servers becoming so overloaded that they hose up and crash your site, causing you to lose lots of money, now would you? That's where balancing the load of traffic between a group of servers comes to the rescue, because even if one of them freezes, your customers will still be able to access your site and place orders.

Multifunction Network Devices

This term applies to any multifunction device that's connected to the network, which provides any combination of printing, copying, faxing, and scanning. Figure 5.18 shows a multifunction network device.

These devices are all-in-one solutions, and you will find them in pretty much every network these days. Why buy a printer, a copier, and a separate fax machine when one machine can do it all for basically the same price?

Domain Name Service (DNS) Server

A *Domain Name Service (DNS) server* is one of the most important servers in your network and on the Internet as well. Why? Because without a DNS server, you would have to type **http://206.123.114.186** instead of simply entering **www.lammle.com**. So it follows that you can pretty much think of a DNS server as the phone book of the Internet.

FIGURE 5.18 A multifunction network device

A host name is typically the name of a device that has a specific IP address; on the Internet, it is part of what is known as a fully qualified domain name (FQDN). An FQDN consists of a host name and a domain name.

The process of finding the IP address for any given host name is known as *name resolution*, and it can be performed in several ways: a HOSTS file (meaning you statically type in all names and IP addresses on each and every host), a request broadcast on the local network (Microsoft's favorite—why ask a server when you can just broadcast, right?), DNS, and Microsoft's Windows Internet Naming Service (WINS). DNS is the most popular today and is the resolution method you really need to know.

On the Internet, domains are arranged in a hierarchical tree structure. The following list includes some of the top-level domains currently in use:

`.com` A commercial organization. Most companies end up as part of this domain.

`.edu` An educational establishment, such as a university.

`.gov` A branch of the U.S. government.

`.int` An international organization, such as NATO or the United Nations.

`.mil` A branch of the U.S. military.

`.net` A network organization.

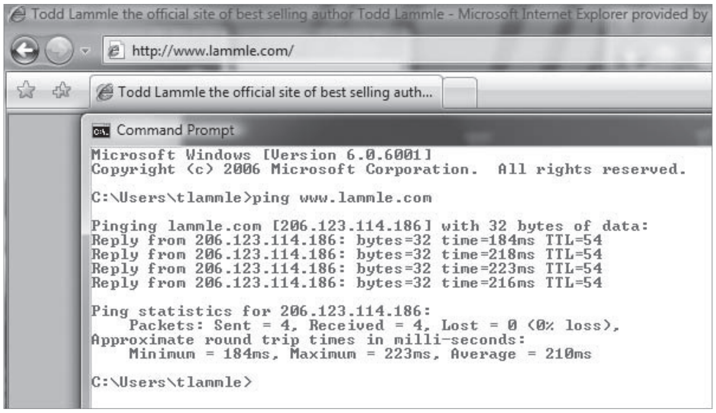
`.org` A nonprofit organization.

Your local ISP is probably a member of the `.net` domain, and your company is probably part of the `.com` domain. The `.gov` and `.mil` domains are reserved strictly for use by the government and the military within the United States. In other parts of the world, the final part of a domain name represents the country in which the server is located (`.ca` for Canada, `.jp` for Japan, `.uk` for Great Britain, and `.ru` for Russia, for example). Well over 130 countries are represented on the Internet.

The `.com` domain is by far the largest, followed by the `.edu` domain. Some new domain names are becoming popular, however, because of the increasing number of domain-name requests. These include `.firm` for businesses and companies, `.store` for businesses selling goods rather than services, `.arts` for cultural and entertainment organizations, and `.info` for informational services. The domains `.cc`, `.biz`, `.travel`, and `.post` are also in use on the Internet.

Let's see how a basic DNS server works in your network. Figure 5.19 shows how, when you type in a human name, the DNS server resolves it, allowing the host to send the HTTP packets to the server.

FIGURE 5.19 DNS resolution example



Todd Lammle the official site of best selling author Todd Lammle - Microsoft Internet Explorer provided by
http://www.lammle.com/
Todd Lammle the official site of best selling auth...
Command Prompt

```
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation.  All rights reserved.

C:\Users\tlammle>ping www.lammle.com

Pinging lammle.com [206.123.114.186] with 32 bytes of data:
Reply from 206.123.114.186: bytes=32 time=184ms TTL=54
Reply from 206.123.114.186: bytes=32 time=218ms TTL=54
Reply from 206.123.114.186: bytes=32 time=223ms TTL=54
Reply from 206.123.114.186: bytes=32 time=216ms TTL=54

Ping statistics for 206.123.114.186:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 184ms, Maximum = 223ms, Average = 210ms

C:\Users\tlammle>
```

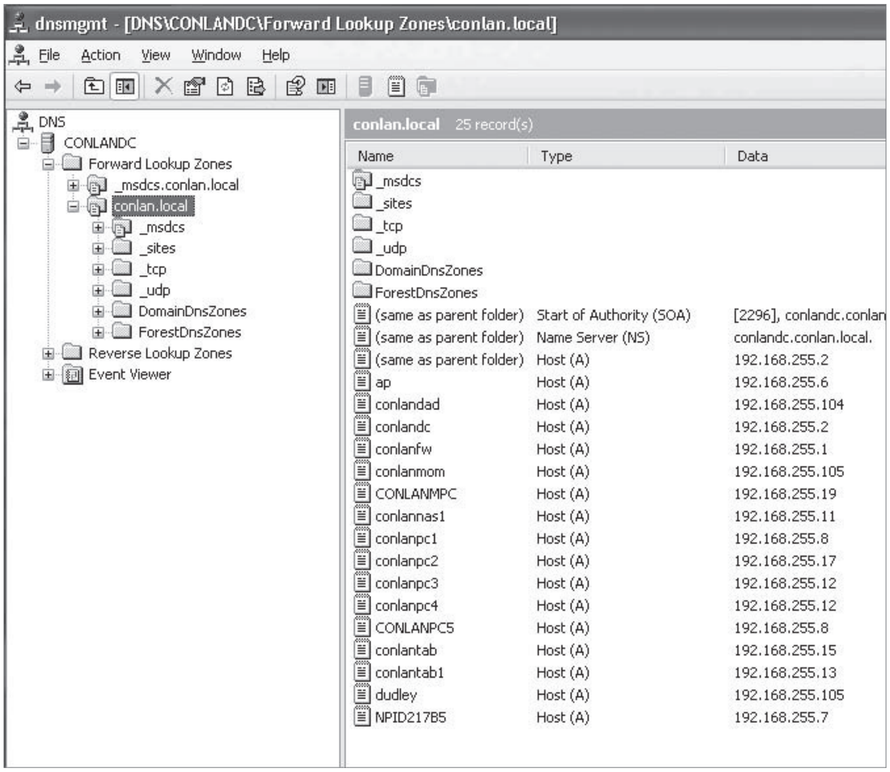
This DOS screen shows how the DNS server can resolve the human name to the IP address of the `Lammle.com` server when I ping the server by the name instead of the IP address.

It should be easy to imagine how hard life would be without DNS translating human names to IP addresses, routing your packet through the Internet, or internetwork to get to your servers. Figure 5.20 gives you an example of a Windows server configured as a DNS server.

Now the hosts can receive the IP address of this DNS server, and then this server will resolve host names to correct IP address. This is a mission-critical service in today's networks, don't you think? As shown in Figure 5.20, if I ping from a host to `conlanpc1`, the host will send the name-resolution request to the DNS server and translate this name to IP address 192.168.255.8.

Host (A) is called an A record and is what gives you the IP address of a domain or host. In IPv6, it's called a quad-A or AAAA record. As shown in Figure 5.20, you can see that each name has an A record, which is associated to an IP address. Okay, so "A" record resolve hostnames to IP addresses, but what happens if you know the IP address and want to know the hostname? There is a record for this too! It's called the pointer record (PTR).

FIGURE 5.20 A Windows DNS server



Name	Type	Data
_msdcs		
_sites		
_tcp		
_udp		
DomainDnsZones		
ForestDnsZones		
(same as parent folder)	Start of Authority (SOA)	[2296], conlandc.conlan
(same as parent folder)	Name Server (NS)	conlandc.conlan.local.
(same as parent folder)	Host (A)	192.168.255.2
ap	Host (A)	192.168.255.6
conlandad	Host (A)	192.168.255.104
conlandc	Host (A)	192.168.255.2
conlanfw	Host (A)	192.168.255.1
conlanmom	Host (A)	192.168.255.105
CONLANMPC	Host (A)	192.168.255.19
conlannas1	Host (A)	192.168.255.11
conlanpc1	Host (A)	192.168.255.8
conlanpc2	Host (A)	192.168.255.17
conlanpc3	Host (A)	192.168.255.12
conlanpc4	Host (A)	192.168.255.12
CONLANPC5	Host (A)	192.168.255.8
conlantab	Host (A)	192.168.255.15
conlantab1	Host (A)	192.168.255.13
dudley	Host (A)	192.168.255.105
NPID217B5	Host (A)	192.168.255.7

Other typical records found on DNS servers are *mail exchanger (MX) records*, which are used to translate mail records. The MX record points to the mail exchanger for a particular host. DNS is structured so that you can actually specify several mail exchangers for one host. This feature provides a higher probability that email will arrive at its intended destination. The mail exchangers are listed in order in the record, with a priority code that indicates the order in which they should be accessed by other mail-delivery systems.

If the first-priority mail exchanger doesn't respond in a given amount of time, the mail-delivery system tries the second one, and so on. Here are some sample mail-exchange records:

```
hostname.company.com.  IN    MX    10 mail.company.com.
hostname.company.com.  IN    MX    20 mail2.company.com.
hostname.company.com.  IN    MX    30 mail3.company.com.
```

In this example, if the first mail exchanger, `mail.company.com`, does not respond, the second one, `mail2.company.com`, is tried, and so on.

Another important record type on a DNS is the canonical name *(CNAME) record*. This is also commonly known as the *alias record* and allows hosts to have more than one name. For example, suppose your web server has the host name www, and you want that machine to also have the name `ftp` so that users can use FTP to access a different portion of the file system as an FTP root. You can accomplish this with a CNAME record. Given that you already have an address record established for the host name www, a CNAME record that adds ftp as a host name would look something like this:

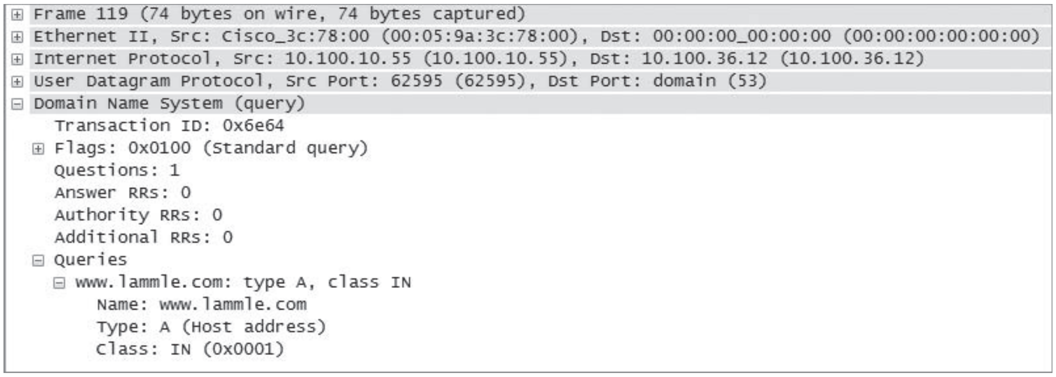
```
www.company.com.       IN    A       204.176.47.2
ftp.company.com.       IN    CNAME   www.company.com.
```

When you put all these record types together in a zone file, or DNS table, it might look like this:

```
mail.company.com.      IN    A       204.176.47.9
mail2.company.com.     IN    A       204.176.47.21
mail3.company.com.     IN    A       204.176.47.89
yourhost.company.com.  IN    MX      10 mail.company.com.
yourhost.company.com.  IN    MX      20 mail2.company.com.
yourhost.company.com.  IN    MX      30 mail3.company.com.
www.company.com.       IN    A       204.176.47.2
ftp.company.com.       IN    CNAME   www.company.com.
```

Let's take a look a tad deeper for a minute into how resolution takes place between a host and a DNS server. Figure 5.21 shows a DNS query from my host to www.lammle.com from a browser.

FIGURE 5.21 A DNS query to www.lammle.com



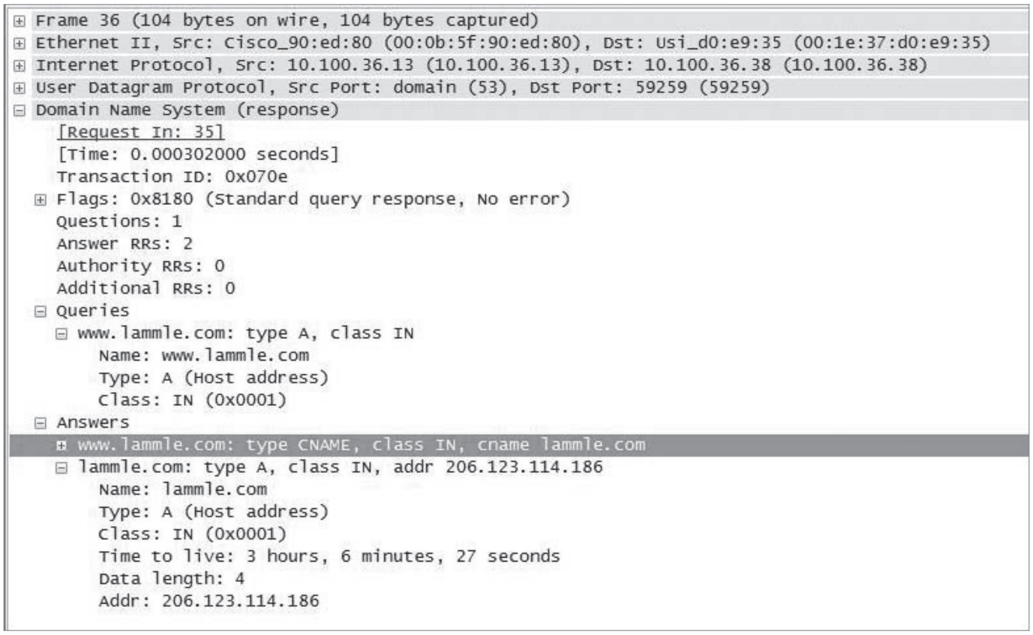
```
⊞ Frame 119 (74 bytes on wire, 74 bytes captured)
⊞ Ethernet II, Src: Cisco_3c:78:00 (00:05:9a:3c:78:00), Dst: 00:00:00_00:00:00 (00:00:00:00:00:00)
⊞ Internet Protocol, Src: 10.100.10.55 (10.100.10.55), Dst: 10.100.36.12 (10.100.36.12)
⊞ User Datagram Protocol, Src Port: 62595 (62595), Dst Port: domain (53)
⊟ Domain Name System (query)
    Transaction ID: 0x6e64
  ⊞ Flags: 0x0100 (Standard query)
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ⊟ Queries
    ⊟ www.lammle.com: type A, class IN
        Name: www.lammle.com
        Type: A (Host address)
        Class: IN (0x0001)
```

This figure shows that DNS uses User Datagram Protocol (UDP) at the Transport layer (it uses Transport Control Protocol [TCP] if it is updating its phone book pages—we call these *zone updates*), and this query is asking destination port 53 (the DNS service) on host 10.100.36.13 who the heck www.lammle.com is.

Let's take a look at the server's response. Figure 5.22 shows the DNS answer to our query for www.lammle.com.

Port 53 answered from server 10.100.36.13 with the IP address of 206.123.114.186. My host can now go to that server requesting HTTP pages using the IP address.

FIGURE 5.22 The DNS answer to our query



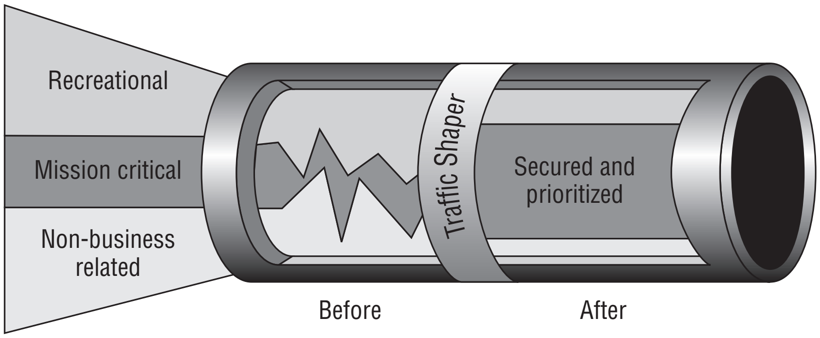
```
⊞ Frame 36 (104 bytes on wire, 104 bytes captured)
⊞ Ethernet II, Src: Cisco_90:ed:80 (00:0b:5f:90:ed:80), Dst: Usi_d0:e9:35 (00:1e:37:d0:e9:35)
⊞ Internet Protocol, Src: 10.100.36.13 (10.100.36.13), Dst: 10.100.36.38 (10.100.36.38)
⊞ User Datagram Protocol, Src Port: domain (53), Dst Port: 59259 (59259)
⊟ Domain Name System (response)
    [Request In: 35]
    [Time: 0.000302000 seconds]
    Transaction ID: 0x070e
  ⊞ Flags: 0x8180 (Standard query response, No error)
    Questions: 1
    Answer RRs: 2
    Authority RRs: 0
    Additional RRs: 0
  ⊟ Queries
    ⊟ www.lammle.com: type A, class IN
        Name: www.lammle.com
        Type: A (Host address)
        Class: IN (0x0001)
  ⊟ Answers
    ⊞ www.lammle.com: type CNAME, class IN, cname lammle.com
    ⊟ lammle.com: type A, class IN, addr 206.123.114.186
        Name: lammle.com
        Type: A (Host address)
        Class: IN (0x0001)
        Time to live: 3 hours, 6 minutes, 27 seconds
        Data length: 4
        Addr: 206.123.114.186
```

Bandwidth Shaper

Sometimes referred to as packet shaping or a traffic shaper, a *bandwidth shaper* is essentially another great tool used for optimizing a network's performance. It works by controlling computer network traffic and delaying specified packets to lower response time and maximize the network's available bandwidth.

Traffic shaping really means setting parameters on particular types of profiled data streams that delay the earmarked packets' flow through the network. Figure 5.23 provides a before-and-after snapshot of what data can look like when bandwidth shaping is applied to it.

FIGURE 5.23 Bandwidth shaping



Nice! Why would anyone choose to run a large internetwork without a bandwidth shaper? Because it's expensive, that's why. If you can't shell out the money for your large corporate network to have a bandwidth shaper, just make sure that all the typical porn sites and other popular (nonwork) sites like YouTube, MySpace, Facebook, and so on are blocked from users accessing them inside your network; then, for the most part, you'll be fine.

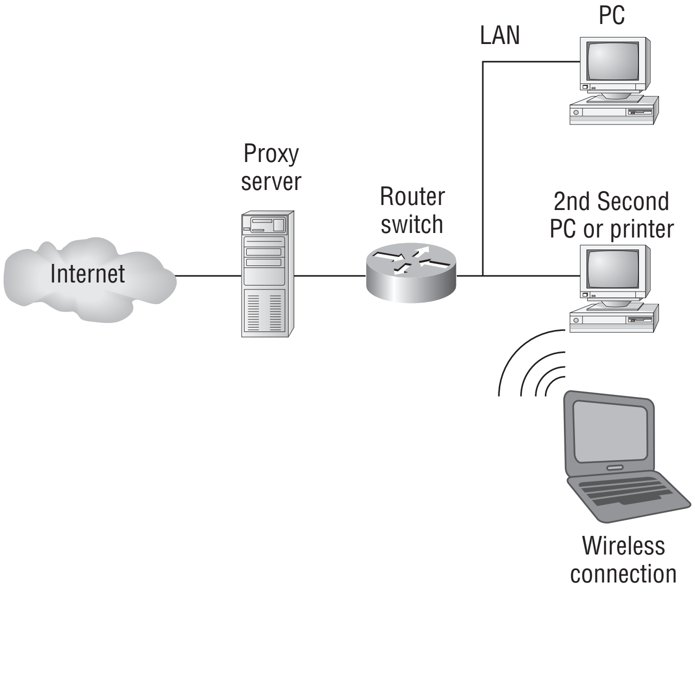
Proxy Server

A *proxy server* is basically a type of server that handles its client-machine requests by forwarding them on to other servers while allowing granular control over the traffic between the local LAN and the Internet. When it receives a request, the proxy will then connect to the specific server that can fulfill the request for the client that wants it.

Sometimes the proxy modifies the client's request or a server's response to it—or even handles the client's request itself. It will actually cache or "remember" the specific server that would have normally been contacted for the request in case it's needed another time. This behavior really speeds up the network's function, thereby optimizing its performance. However, proxy servers can also limit the availability of the types of sites that users on a LAN have access to, which is a benefit for an administrator of the network if users are constantly connected to nonwork sites and using all the WAN bandwidth.

Figure 5.24 shows where a proxy server would be typically found in a small to medium-size network.

FIGURE 5.24 A proxy server



There are two main types of proxy servers you'll typically find working in present-day networks:

Caching proxy server A caching proxy server speeds up the network's service requests by recovering information from a client's or clients' earlier request. Caching proxies keep local copies of the resources requested often, which really helps minimize the upstream use of bandwidth. These servers can greatly enhance network performance.

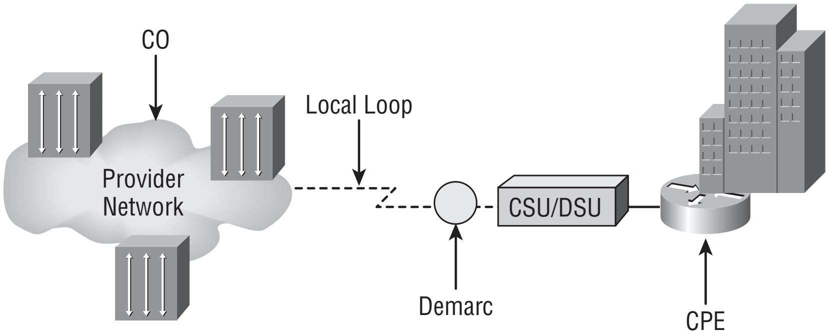
Web proxy server A web proxy server is usually used to create a web cache. You experience this when you Google a site you've visited before. The web proxy "remembers" you, and the site not only loads faster, but sometimes even recalls your personal information by automatically filling in your username—or even your billing/shipping information when you place another order.

Channel Service Unit/Data Service Unit (CSU/DSU)

The CSU/DSU is a common device found in equipment rooms when the network is connected via a T-series data connection or other digital serial technology like a T1 or Digital Data Server (DDS). It's essentially two devices in one that are used to connect a digital carrier (the T-series or DDS line) to your network equipment—usually to a router. The *Channel Service Unit (CSU)* terminates the line at the customer's premises and also provides diagnostics and remote testing, if necessary. The *Data Service Unit (DSU)* does the actual transmission of the signal through the CSU and can also provide buffering and data-flow control.

Figure 5.25 shows where a typical CSU/DSU would be used for a T1 connection

FIGURE 5.25 Typical placement of a CSU/DSU device



The CSU/DSU connects to your router on one side, and into what is called a demarcation location on the other—which connects your network to the providers WAN.

Both components of a CSU/DSU are required if you are going to connect to a digital transmission medium like a T1 line; and sometimes, one or both of these components may even be built into a router. In the latter case, you can just go ahead and plug the T1 line directly into the router. Otherwise, you'll need some Physical layer specification, like V.35, to cable the interface on the router to the external CSU/DSU.

Okay—with all that in mind, it's time to delve deeper into the particulars of the most common devices found in today's networks: hubs, switches, and routers. The next section will give examples and more detail regarding how these devices are used.

Network Segmentation

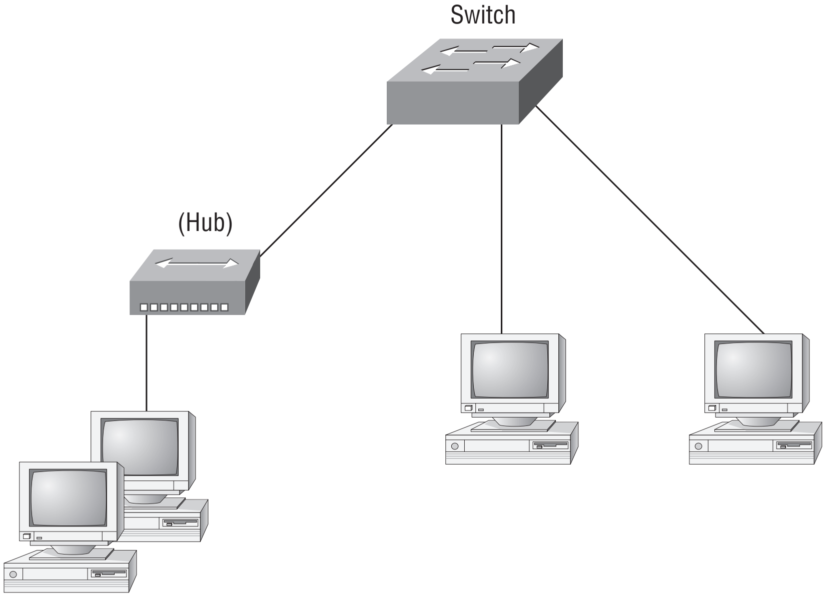
It's very likely that at some point you'll have to break up one large network into a bunch of smaller ones because user response will have dwindled to a slow crawl as the network grew and grew. With all that growth, your LAN's traffic congestion will have reached epic proportions. So, now I'm going to show you how to use the segmentation devices I have defined so far in this chapter.

Here's a list of some of the nasty things that commonly cause LAN traffic congestion:

- Too many hosts in a broadcast domain
- Broadcast storms
- Multicasting
- Low bandwidth
- Adding hubs for connectivity to the network

The answer to fixing a huge but slow network is to break it up into a number of smaller networks—something called *network segmentation*. You do this by using devices like routers and switches, which are sometimes still referred to as bridges because switches still use bridging technologies. Figure 5.26 displays a network that's been segmented with a switch so each network segment connected to the switch is now a separate collision domain. But make note of the fact that this network is actually still one *broadcast domain*——the set of all devices on a network segment that hear all the broadcasts sent on that segment.

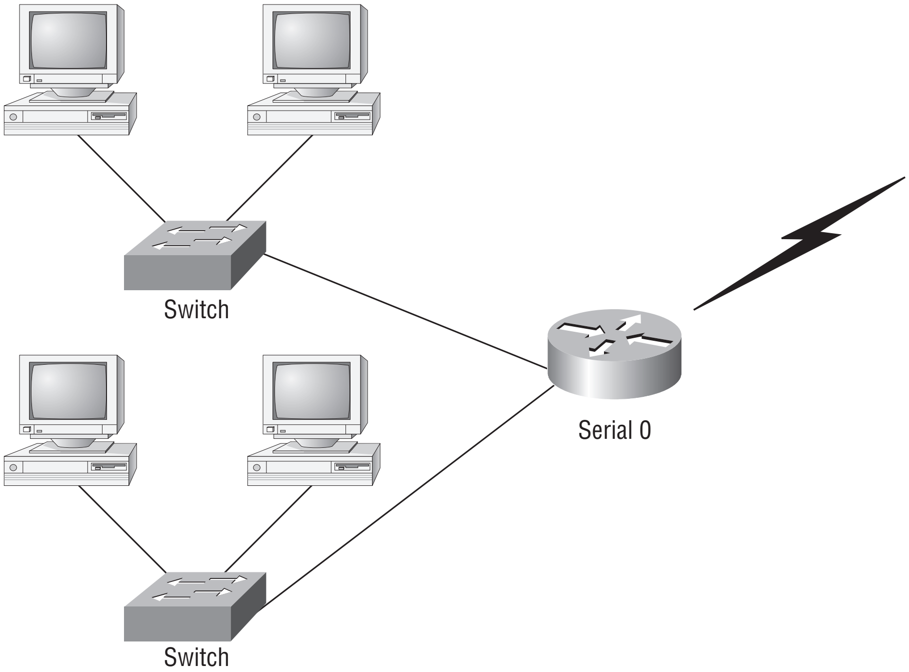
FIGURE 5.26 A switch can replace the hub, breaking up collision domains.



And keep in mind that the hub used in Figure 5.26 just extended the one collision domain from the switch port.

Routers are used to connect networks together and route packets of data from one network to another. (Cisco has become the de facto standard for routers because of its high-quality router products, great selection, and fantastic service.) Routers, by default, break up a broadcast domain. Figure 5.27 shows a router in our little network that creates an internetwork and breaks up broadcast domains.

FIGURE 5.27 Routers create an internetwork.



The network in Figure 5.27 is pretty cool. Each host is connected to its own collision domain, and the router has created two broadcast domains. And don't forget that the router provides connections to WAN services as well. The router uses something called a serial interface for WAN connections: specifically, a V.35 physical interface.

Breaking up a broadcast domain is important because when a host or server sends a network broadcast, every device on the network must read and process that broadcast—unless you've got a router. When the router's interface receives this broadcast, it can respond by basically saying, "Thanks, but no thanks," and discard the broadcast without forwarding it on to other networks. Even though routers are known for breaking up broadcast domains by default, it's important to remember that they break up collision domains as well.

There are two advantages of using routers in your network:

- They don't forward broadcasts by default.
- They can filter the network based on Layer 3 (Network layer) information (such as IP address).

Four router functions in your network can be listed as follows:

- Packet switching
- Packet filtering
- Internetwork communication
- Path selection

Remember that routers are really switches; they're actually what we call Layer 3 switches. Unlike Layer 2 switches, which forward or filter frames, routers (Layer 3 switches) use logical addressing and provide what is called *packet switching*. Routers can also provide packet filtering by using access lists; and when routers connect two or more networks together and use logical addressing (IP or IPv6), this is called an *internetwork*. Last, routers use a *routing table* (map of the internetwork) to make path selections and to forward packets to remote networks.

Conversely, switches aren't used to create internetworks (they do not break up broadcast domains by default); they're employed to add functionality to a network LAN. The main purpose of a switch is to make a LAN work better—to optimize its performance—providing more bandwidth for the LAN's users. And switches don't forward packets to other networks as routers do. Instead, they only "switch" frames from one port to another within the switched network. Okay, you may be thinking, "Wait a minute, what are frames and packets?" I'll tell you all about them later in this chapter, I promise.

By default, switches break up collision domains, as mentioned in Chapter 4, "The Current Ethernet Specifications." This is an Ethernet term used to describe a network scenario wherein one particular device sends a packet on a network segment, forcing every other device on that same segment to pay attention to it. At the same time, a different device tries to transmit, leading to a collision, after which both devices must retransmit, one at a time. Not very efficient! This situation is typically found in a hub environment where each host segment connects to a hub that represents only one collision domain and only one broadcast domain. By contrast, each and every port on a switch represents its own collision domain.

NOTE Switches create separate collision domains but a single broadcast domain. Routers provide a separate broadcast domain for each interface.

The term *bridging* was introduced before routers and hubs were implemented, so it's pretty common to hear people referring to bridges as switches. That's because bridges and switches basically do the same thing—break up collision domains on a LAN (in reality, you cannot buy a physical bridge these days, only LAN switches; but these switches use bridging technologies, so Cisco still calls them multiport bridges).

So this means a switch is basically just a multiple-port bridge with more brainpower, right? Well, pretty much, but there are differences. Switches do provide this function, but they do so with greatly enhanced management ability and features. Plus, most of the time, bridges only had two or four ports. Yes, you could get your hands on a bridge with up to 16 ports, but that's nothing compared to the hundreds available on some switches.

still need a router to provide your inter-VLAN communication, or internetworking. Don't forget that.

Obviously, the best network is one that's correctly configured to meet the business requirements of the company it serves. LAN switches with routers, correctly placed in the network, are the best network design. This book will help you understand the basics of routers and switches so you can make tight, informed decisions on a case-by-case basis.

Let's go back to Figure 5.28 again. Looking at the figure, how many collision domains and broadcast domains are in this internetwork? I hope you answered nine collision domains and three broadcast domains.

The broadcast domains are definitely the easiest to see because only routers break up broadcast domains by default. And because there are three connections, that gives you three broadcast domains. But do you see the nine collision domains? Just in case that's a no, I'll explain. The all-hub network is one collision domain; the bridge network equals three collision domains. Add in the switch network of five collision domains—one for each switch port—and you've got a total of nine.

Now, in Figure 5.29, each port on the switch is a separate collision domain and each VLAN is a separate broadcast domain. But you still need a router for routing between VLANs. How many collision domains do you see here? I'm counting 10—remember that connections between the switches are considered a collision domain.

Real World Scenario

Should I Replace All My Hubs with Switches?

You're a network administrator at a large company in San Jose. The boss comes to you and says that he got your requisition to buy a switch and is not sure about approving the expense; do you really need it?

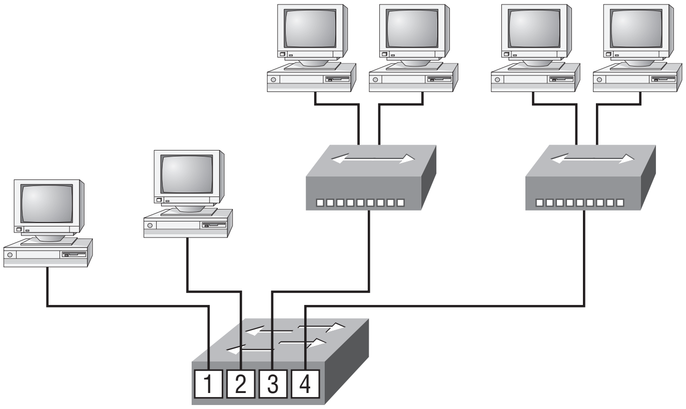
Well, if you can, sure—why not? Switches really add a lot of functionality to a network that hubs just don't have. But most of us don't have an unlimited budget. Hubs still can create a nice network—that is, of course, if you design and implement the network correctly.

Let's say that you have 40 users plugged into four hubs, 10 users each. At this point, the hubs are all connected together so that you have one large collision domain and one large broadcast domain. If you can afford to buy just one switch and plug each hub into a switch port, as well as plug the servers into the switch, then you now have four collision domains and one broadcast domain. Not great; but for the price of one switch, your network is a much better thing. So, go ahead! Put that requisition in to buy all new switches. What do you have to lose?

So now that you've gotten an introduction to internetworking and the various devices that live in an internetwork, it's time to head into internetworking models.

the frame was received on. This information (logged in the bridge's or switch's filter table) is what helps the machine determine the location of the specific sending device. Figure 5.31 shows a switch in an internetwork.

FIGURE 5.31 A switch in an internetwork



Each segment has its own collision domain.
All segments are in the same broadcast domain.

The real-estate business is all about location, location, location, and it's the same way for both Layer 2 and Layer 3 devices. Although both need to be able to negotiate the network, it's crucial to remember that they're concerned with very different parts of it. Primarily, Layer 3 machines (such as routers) need to locate specific networks, whereas Layer 2 machines (switches and bridges) need to eventually locate specific devices. So, networks are to routers as individual devices are to switches and bridges. And routing tables that "map" the internetwork are for routers as filter tables that "map" individual devices are for switches and bridges.

After a filter table is built on the Layer 2 device, it will forward frames only to the segment where the destination hardware address is located. If the destination device is on the same segment as the frame, the Layer 2 device will block the frame from going to any other segments. If the destination is on a different segment, the frame can be transmitted only to that segment. This is called *transparent bridging*.

When a switch interface receives a frame with a destination hardware address that isn't found in the device's filter table, it will forward the frame to all connected segments. If the unknown device that was sent the "mystery frame" replies to this forwarding action, the switch updates its filter table regarding that device's location. But in the event that the destination address of the transmitting frame is a broadcast address, the switch will forward all broadcasts to every connected segment by default.

All devices that the broadcast is forwarded to are considered to be in the same broadcast domain. This can be a problem; Layer 2 devices propagate Layer 2 broadcast storms that choke performance, and the only way to stop a broadcast storm from propagating through an internetwork is with a Layer 3 device—a router.

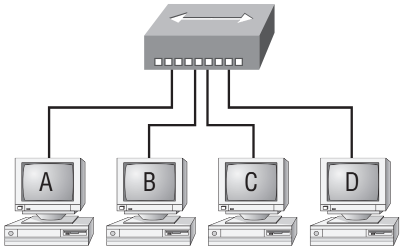
The biggest benefit of using switches instead of hubs in your internetwork is that each switch port is actually its own collision domain. (Conversely, a hub creates one large collision domain.) But even armed with a switch, you still can't break up broadcast domains. Neither switches nor bridges will do that. They'll typically simply forward all broadcasts instead.

Another benefit of LAN switching over hub-centered implementations is that each device on every segment plugged into a switch can transmit simultaneously—at least, they can as long as there is only one host on each port and a hub isn't plugged into a switch port. As you might have guessed, hubs allow only one device per network segment to communicate at a time.

Hubs at the Physical Layer

As you know, a hub is really a multiple-port repeater. A repeater receives a digital signal and reamplifies or regenerates that signal and then forwards the digital signal out all active ports without looking at any data. An active hub does the same thing. Any digital signal received from a segment on a hub port is regenerated or reamplified and transmitted out all ports on the hub. This means all devices plugged into a hub are in the same collision domain as well as in the same broadcast domain. Figure 5.32 shows a hub in a network.

FIGURE 5.32 A hub in a network



All devices in the same collision domain
All devices in the same broadcast domain
Devices share the same bandwidth.

Hubs, like repeaters, don't examine any of the traffic as it enters and is then transmitted out to the other parts of the physical media. Every device connected to the hub, or hubs, must listen if a device transmits. A physical star network—where the hub is a central device and cables extend in all directions out from it—is the type of topology a hub creates. Visually, the design really does resemble a star, whereas Ethernet networks run a logical bus topology, meaning that the signal has to run through the network from end to end.

NOTE Hubs and repeaters can be used to enlarge the area covered by a single LAN segment, although I do not recommend this. LAN switches and/or wireless APs are affordable for almost every situation.

Summary

Whew, this chapter covered quite a bit of information. In this chapter, you learned the difference between a router, a switch (bridge), and a hub and when to use each one. I also covered some devices that you might find in a network today, but not as often, such as a repeater modem and media convertors.

The information I discussed about DNS and DHCP is critical to your success on the Network+ objectives, and I highly suggest that you reread those sections. I covered how both the DNS and DHCP services works on a network.

In addition to the most common devices, I discussed the specialized network devices mentioned in the Network+ objectives. I finished the chapter by discussing collision and broadcast domains in detail as well as how you would use a router, switch, and hub in your networks today.

All of the information in this chapter is fundamental, and you must understand it before moving on to the other chapters in this book.

Exam Essentials

Understand how DHCP works and its purpose. Dynamic Host Configuration Protocol (DHCP) provides IP configuration information to hosts. It is important to know how a DHCP client requests information from a server, how a server receives this information, and also how the server responds to the client and with what type of information.

Understand how DNS works and its purpose. Domain Name Service (DNS) is used to resolve human names to binary format. Understanding how DNS resolves these names is critical, as is understanding how a DNS query is sent and how a DNS server responds.

Understand the difference between a hub, a switch (bridge), and a router. A hub just connects network segments together. A switch/bridge segments the network using MAC addresses, and a router segments the network using logical addressing (IP and IPv6).

Remember the different names for a router A router is a layer-3 hardware device, but can also be called a layer-3 switch, or a multilayer switch.

Remember the various devices used on networks today and when you would use each one and how. Understand the differences and how each device works: hubs, repeaters, modems, NICs, media convertors, WAPs, switches, routers, and DHCP.

Understand what IDS and IPS is and what each one does IDS only identifies, detects, and reports attempts of unauthorized access to the network as well as any suspicious activity, and is the best software type that would be best for identifying an attack. However, if you want to stop the attack in its track you need to add an IPS device. An Intrusion Prevention System (IPS) provides computers with security by vigilantly watching for any suspicious and potentially malicious tactics and prevents these evil activities. Unlike IDS, IPS will identify and possibly shut down a port or drop certain type of packets.

Identify the purpose, benefits, and characteristics of using a proxy service. A proxy server keeps a LAN somewhat separated from the Internet. Doing so increases security and filtering control and has the tendency to speed up Internet access through caching of recently used web pages.

Written Lab

Complete the table by filling in the appropriate layer of the OSI or hub, switch, or router device.

Description	Device or OSI Layer
This device sends and receives information about the Network layer.	
This layer creates a virtual circuit before transmitting between two end stations.	
A layer-3 switch or multilayer switch	
This device uses hardware addresses to filter a network.	
Ethernet is defined at these layers.	
This layer supports flow control and sequencing.	
This device can measure the distance to a remote network.	
Logical addressing is used at this layer.	
Hardware addresses are defined at this layer.	
This device creates one big collision domain and one large broadcast domain.	
This device creates many smaller collision domains, but the network is still one large broadcast domain.	
This device can never run full duplex.	
This device breaks up collision domains and broadcast domains.	

(The answers to the Written Lab can be found following the answers to the Review Questions for this chapter.)

Review Questions

1. Which is not a common term associated with modems?

   A. POTS

   B. DSL

   C. Cable

   D. NIC

2. What advantage does a switch have over a hub?

   A. It discards frames.

   B. Transmissions received on one port will be sent out all the other ports.

   C. It recognizes frame boundaries and destination MAC addresses of incoming frames.

   D. Any two or more devices the switch connects have the capability of causing a collision with each other.

3. Which device is used to segment a network?

   A. Hub

   B. Switch

   C. Repeater

   D. All of the above

4. What is the primary function of a bridge?

   A. Breaks up collision domains

   B. Allows a NIC or other networking device to connect to a different type of media than it was designed for

   C. Allows mobile users to connect to a wired network wirelessly

   D. None of the above

5. A network device that is used to connect multiple devices together without segmenting a network is a?

   A. Hub

   B. Wireless access point

   C. Switch

   D. Router

6. What is the function of a firewall?

   A. Protects LAN resources from attackers on the Internet

   B. Provides extra bandwidth

   C. Reduces throughput

   D. Allows access to all computers on a LAN

7. Which of the following devices can work at both Layers 2 and 3 of the OSI model?
   - A. Hub
   - B. Switch
   - C. Repeater
   - D. Bridge

8. What is an advantage of using DHCP in a network environment?
   - A. More difficult administration of the network
   - B. Static IP addressing
   - C. Can send an operating system for the PC to boot from
   - D. Assigns IP address to hosts

9. What is a benefit of a multilayer switch (MLS) over a Layer 2 switch?
   - A. Less bandwidth
   - B. Routing functions
   - C. Fewer features
   - D. Fewer ports

10. Which device should be used if you need to send incoming packets to one or more machines that are hidden behind a single IP address?
    - A. Switch
    - B. Load balancer
    - C. Hub
    - D. Repeater

11. What role does the "A" record in a Domain Name Service (DNS) server have in your network?
    - A. Translates human name to IP address
    - B. Translates IP address to human name
    - C. Enables printing, copying, and faxing from one device
    - D. Controls network packets to optimize performance

12. Which device does not aid in network segmentation?
    - A. Router
    - B. Switch
    - C. Hub
    - D. Bridge

13. What is the most common use for a web proxy?

- **A.** Web cache
- **B.** Increases throughput
- **C.** Provides administrative control
- **D.** Supports user authentication

14. Which is not an advantage of network segmentation?

- **A.** Reduced congestion
- **B.** Improved security
- **C.** Containing network problems
- **D.** Preventing broadcast storms

15. Users arrive at the office after a weekend and the hosts that were shut down over the weekend are restarted but cannot access the LAN or Internet. Hosts that were not shut down are working fine. Where can the problem be?

- **A.** The DNS server
- **B.** The DHCP server
- **C.** The Proxy server
- **D.** The Firewall

16. You need a device that detects and reports attempts of unauthorized access to your network, identifies suspicious activity, and is best for identifying an attack. Which device should you install?

- **A.** Firewall
- **B.** IDS
- **C.** IPS
- **D.** Proxy server

17. Which device creates separate collision domains and a single broadcast domain?

- **A.** Hub
- **B.** Router
- **C.** Switch
- **D.** Modem

18. Which device by default does not forward any broadcast or multicast packets?

- **A.** Repeater
- **B.** Hub
- **C.** Router
- **D.** Switch

19. Which type of server in your network uses pointer and A records?
    - A. NAT Translation server
    - B. IPS/IDS Server
    - C. DNS Server
    - D. Proxy Server

20. Users on your network are saturating your bandwidth because they are using too many nonwork related sites. What device would limit the availability of the types of sites that users on a LAN have access to while providing granular control over the traffic between the local LAN and the Internet?
    - A. Switch
    - B. DHCP server
    - C. DNS server
    - D. Proxy server

Answers to Review Questions

1. D. A modem is a device that modulates digital data onto an analog carrier for transmission over an analog medium and then demodulates from the analog carrier to a digital signal again at the receiving end. Therefore, traditional (POTS), DSL, and cable are all common types of modems. The NIC is the expansion card you install in your computer to connect, or interface, your computer to the network.

2. C. Like a hub, a switch connects multiple segments of a network together, with one important difference. Whereas a hub sends out anything it receives on one port to all the others, a switch recognizes frame boundaries and pays attention to the destination MAC address of the incoming frame as well as the port on which it was received.

3. B. Hubs don't segment a network; they just connect network segments together. Repeaters don't segment the network; they repeat a signal and allow the distance covered to be increased. So the only correct option is B, a switch.

4. A. The primary function of a bridge is to keep traffic separated on both sides of the bridge, breaking up collision domains.

5. A. Hubs create one collision domain and one broadcast domain.

6. A. Firewalls are the first line of defense for an Internet-connected network. If a network was directly connected to the Internet without a firewall, an attacker could theoretically gain direct access to the computers and servers on that network with little effort.

7. B. A switch is typically just a Layer 2 device segmenting the network by using MAC addresses. However, some higher-end switches can provide Layer 3 services.

8. D. Remember that DHCP servers assign IP addresses to hosts. Thus DHCP allows easier administration than providing IP information to each host by hand (called static IP addressing).

9. B. Multilayer switches (also called layer-3 switches) don't have any fewer features, less bandwidth, or fewer ports than a normal switch; they just allow routing functions between subnets.

10. B. A load balancer uses a little trickery and sends incoming packets to one or more machines that are hidden behind a single IP address. Modern load-balancing routers can use different rules to make decisions about where to route traffic, which can be based on least load, fastest response times, or simply balancing requests.

11. A. DNS translates human names to IP addresses for routing your packet through the Internet. Hosts can receive the IP address of this DNS server and then resolve host names to IP addresses.

12. C. Routers, switches, and bridges are all devices that help break up big networks into a number of smaller ones—also known as network segmentation. Hubs don't segment networks—they just connect network segments together.

13. A. Web cache, of course! Most proxy programs provide a means to deny access to certain URLs in a blacklist, thus providing content filtering, usually in corporate environments.

14. D. Options A, B, and C all aid in boosting network performance, so the only option left is broadcast storms. Increased traffic will increase LAN congestion.

15. B. If the DHCP server has stopped functioning, it will not hand out IP addresses to hosts that are restarted. However, the hosts that were not shut down still have an IP addresses because the lease time has not expired.

16. B. An IDS device can detect and report suspicious activity, but unlike an IPS, it does not stop attacks. IDS is best for identifying an attack.

17. C. Switches create separate collision domains but a single broadcast domain. Remember that routers provide a separate broadcast domain for each interface.

18. C. Routers don't forward any broadcast or multicast packets by default, but they do have plenty of other functions like using the logical address, using access lists, and providing Layer 2 bridging functions.

19. C. A DNS server uses many types of records. An "A" record is a hostname to IP address record and a pointer record is an IP address to hostname record.

20. D. A proxy server can provide many functions. A proxy server can use a caching engine so repeated access request for web information would accelerate repeated access for users, and they can also limit the availability of web sites.

Answers to Written Lab

Complete the table by filling in the appropriate layer of the OSI or hub, switch, or router device.

Description	Device or OSI Layer
This device sends and receives information about the Network layer.	Router
This layer creates a virtual circuit before transmitting between two end stations.	Transport
A layer-3 switch, or multilayer switch	Router
This device uses hardware addresses to filter a network.	Bridge or switch
Ethernet is defined at these layers.	Data Link and Physical
This layer supports flow control and sequencing.	Transport
This device can measure the distance to a remote network.	Router
Logical addressing is used at this layer.	Network
Hardware addresses are defined at this layer.	Data Link (MAC sublayer)
This device creates one big collision domain and one large broadcast domain.	Hub
This device creates many smaller collision domains, but the network is still one large broadcast domain.	Switch or bridge
This device can never run full duplex	Hub
This device breaks up collision domains and broadcast domains.	Router