# 1 WELCOME TO THE CLOUD

## LEARNING OBJECTIVES

**By the end of this chapter, you should be able to:**

- Discuss why networking and security are converging.
- Describe basic network terminology.
- Describe packet switching as a way to reduce long-distance transmission costs and error retransmissions. Describe physical links and data links.
- Describe the origins of Internet standards and applications, including e-mail.
- Explain why internetworking was needed.
- Explain why Kahn and Cerf's concept for internetworking resulted in duplicated network concepts. Describe the internet and transport layers.
- Explain the five basic layers of standards in the TCP/IP-OSI Hybrid Standards Architecture.
- Describe TCP/IP standards.
- Explain the evolution of the Internet from research network to commercial network.
- List TCP/IP supervisory standards.
- Describe a small home network.

## IN THE CLOUDS

### Jason Akana

Jason Akana works at the First Bank of Paradise (FBP)[1] in Honolulu, Hawai`i. FBP has 50 branches throughout the state and many more ATMs. Jason helps develop new media campaigns for bank products. His campaigns use Facebook, Twitter, YouTube, e-mail lists, and the bank's website.

Currently, the bank is developing a promotional campaign for its new *AlohaSmart* credit card, which has both a computer chip and a traditional magnetic stripe. Figure 1-1 shows the new card. Cards with computer chips are called **smart cards**. They are common in Europe, and American merchants are beginning to add smart card readers.

[1]The "First Bank of Paradise" is a composite of several banks in Hawai`i. Individual banks are obviously reluctant to have specific information about their networking and security made public.

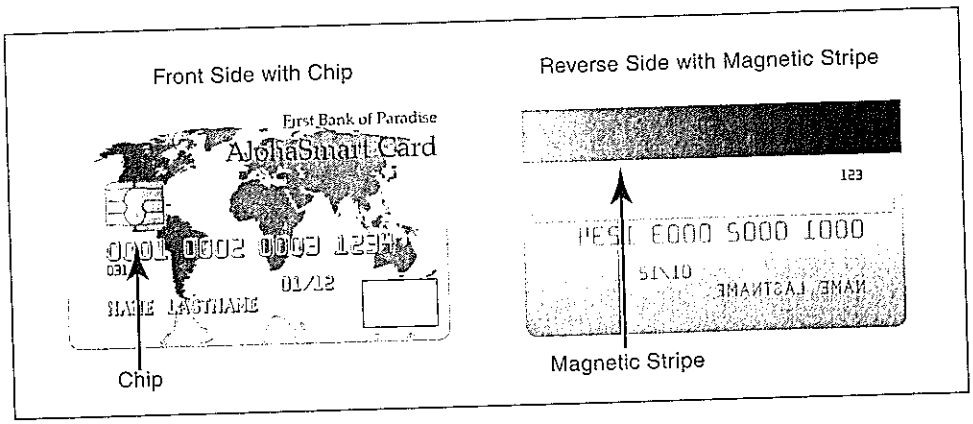


**FIGURE 1-1** *AlohaSmart* Credit Card

*Source:* © omergenc/iStockphoto

The marketing campaign will emphasize the advantages of smart cards over traditional magnetic stripe cards in terms of security and extra services. The card has a slightly higher fee than traditional magnetic stripe cards, so the bank will benefit when customers switch to the new card. Jason has developed a tentative campaign and a PowerPoint presentation to lay out how the bank can use new media to promote the card.

When Jason woke this morning, he opened his tablet and checked his mail. Yesterday, he posted his draft plan and a PowerPoint presentation to everyone in marketing. Three replied with good suggestions for changes. After breakfast, Jason turned on his desktop computer and made the changes to the presentation. He likes to develop new material on his desktop because its large screen allows him to lay out subsidiary material next to the PowerPoint presentation.

Figure 1-2 shows that when Jason finished working on his desktop, his BlueSync software uploaded the PowerPoint presentation to a BlueSync server on the Internet.

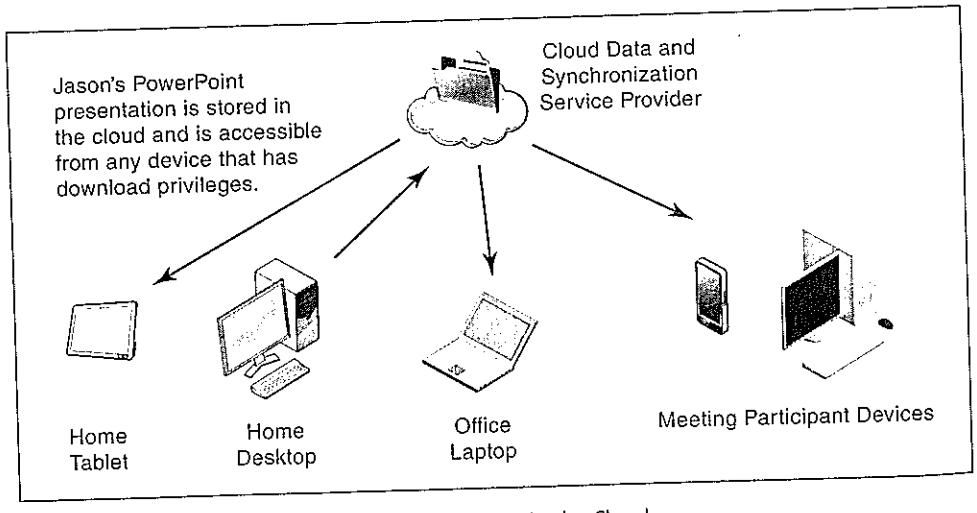


**FIGURE 1-2** File Synchronization and Distribution in the Cloud

In more colorful language, it saved the file "to the cloud." Later, when Jason turns on his tablet, it will be available on the tablet. When he turns on his laptop at work, it will be there as well. It will even be available on Jason's smartphone. In fact, on these machines, any file changed on one will be synchronized on the others. For the coming meeting to explain his plan, it will be available to the attendees as well. BlueSync is a **cloud data and synchronization service provider.**

The term *cloud* is used often in networking and other parts of IT today. The **cloud** imagery signifies that users do not have to know what is happening "inside the cloud." The details are taken care of by the organization or by an outside company. BlueSync is one of these external services.

*The cloud imagery signifies that users do not have to know what is happening "inside the cloud."*

Off to work, Jason rides from his home in Kailua to work in a shared van. It is slower than driving himself, but parking fees are horrendous downtown. On the ride, Jason goes over the presentation several times on his tablet. He also exchanges phone texts with his boss about the presentation.

At home, PowerPoint is installed on his desktop computer. On his tablet, however, Jason uses a cloud version of PowerPoint. It is only downloaded to his tablet when he needs it, and he pays a monthly fee for the service. Figure 1-3 shows that the software is managed by a **cloud software service provider**. This name comes from the fact that software is provided as a service when needed, not on a product to be purchased once and stored on the user's machine like traditional software.

At work, Jason fires up his laptop computer and takes it to the conference room. On this device too, he downloads PowerPoint only when needed. He plugs the laptop into the projector and shows the first slide on the screen. He is ready. Over the next few minutes, four local attendees walk into the room. They exchange small talk, check things on their tablets and laptops, and use their phones to make calls and send texts.

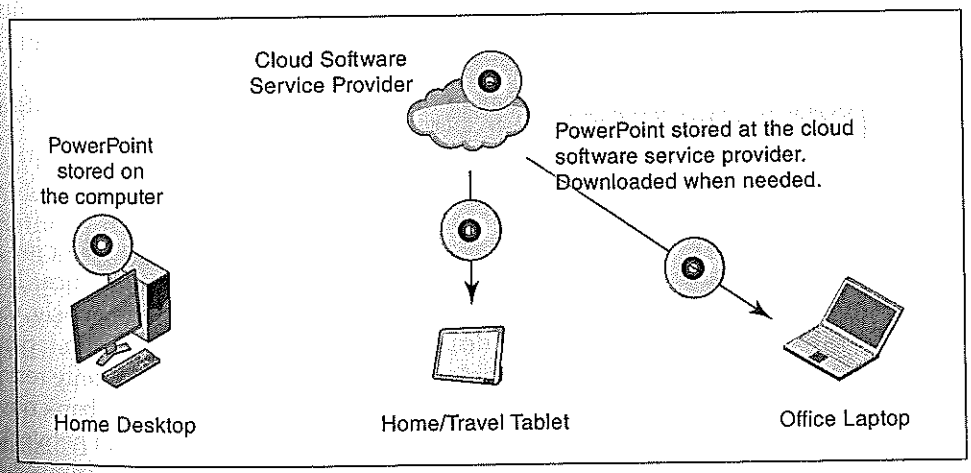


**FIGURE 1-3** Software in the Cloud

At 10:00 am, the meeting starts. In addition to being shown on the screen, slide images are on BlueSync, so everybody in the meeting can bring them up on his or her laptops and tablets. Two of the attendees are at a second site. They can see the slides on their computers, and the conference room has an audioconferencing system for voice communication between the attendees at the two sites.

After the overview presentation, Jason goes over Microsoft Word documents, detailing specific aspects of the new media campaign. The group then discusses how to integrate the new media campaign with the bank's traditional media campaign using print, radio, and television. During the discussion, one of the marketing staff members plays a high-definition television ad the department would like to run. With everyone reasonably satisfied, the marketing director gives the green light to both the new media and traditional media campaigns.

Jason's experience is far from unique. Today, user computers span a wide range of sizes. Most of the newest products have been at the small end of the spectrum, with smartphones and tablets growing rapidly. The multi-screen desktop is at the high end today, but in the future, entire tables and walls will be displays.

None of this would be possible without networks to link people to resources locally and over long distances. Networks have been around for many years, and they have always been useful. Today, they are critical. On the negative side of this criticality, if the network fails, even briefly, a great deal of corporate work is disrupted. Networks must work reliably. Networks must also work well; many applications require very high speeds, so networks must be fast, even when they use wireless transmission, which is notoriously finicky. On the positive side, great networks enable the firm to do things like Jason's meeting, which would have been very difficult to do in the past. In general, networks today must be like traditional servants—excellent and invisible.

### Test Your Understanding

1. a) Why do you think wireless is such a big concern today in networking and security? (In this book, "do you think" questions require you to go beyond what is in the text. You may not be able to answer them perfectly, but try hard because they are good learning opportunities.) b) Distinguish between cloud data storage and synchronization on the one hand and cloud software service on the other. c) What do you think are the advantages of each? d) What do you think are their disadvantages? e) Why do you think the bring your own device (BYOD) revolution has made networking more difficult? List several issues.
2. Go to YouTube and watch "A Day Made of Glass" by the Corning Corporation. List new ways of displaying information shown in the video.

## Claire Lorek

Claire Lorek works in the networking department at the First Bank of Paradise. Her job is to keep the bank's headquarters wireless networking running at peak performance. The bank has over 50 wireless access points to serve its users in the building. Previously, the wireless network had not been given a great deal of attention. It was completely independent of the bank's internal wired network. Employees were told that they could only reach the Internet and should not consider the network secure or always available. The bank now has plans to interconnect its wireless local area network with its wired local area network. The wireless network will have to mature radically in both performance and security.

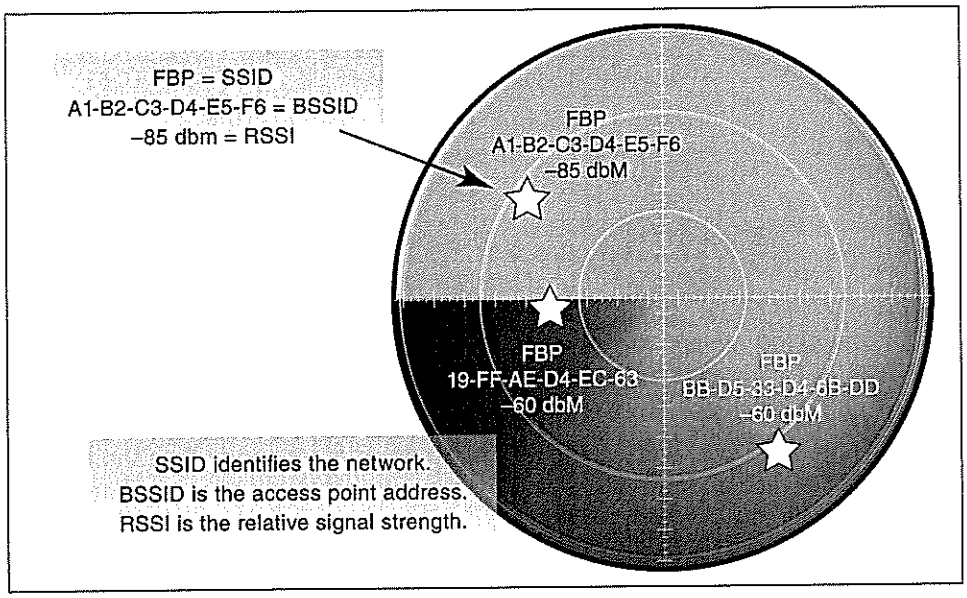


**FIGURE 1-4** The Wireless Sniffer Radar Map

Currently, the bank has no central way to manage its wireless network. Every day and sometimes several times a day, Claire walks around the building with her ultra-portable laptop. At her normal stopping points, she fires up her **wireless sniffer** program. Figure 1-4 shows that the first thing she does is look at the "radar map" which shows the relative directions of nearby access points and their signal strengths.

At this stop, three access points are in range. Each access point has a **service set ID (SSID)**, which is the name of the network. The SSID is FBP, which indicates that this is the bank's network. The access point in the upper left has the **BSSID** A1-B2-C3-D4-E5-F6. This is the address of the specific access point. Note that other access points have different BSSIDs. The **RSSI** value is the relative signal strength indicator, which shows strength in terms of decibel relative to one milliwatt. Everything looks good. The only access points are corporate-approved access points, and signal strengths look good.

Going in more depth, Claire looks at the wireless sniffer details table shown in Figure 1-5. It gives more details about the nearby access points. She checks to make sure that each is properly configured for security. All are using the security methods required by corporate policy. For encryption (so that no one can intercept and read transmissions), all three use AES-CCMP. For authentication (in which a wireless device user proves his or her identity before being allowed to use the access point), the access points all require WPA2/PEAP. We will explain these terms in Chapter 7.

At her next stop, Claire notices a signal strength problem for one of the access points. The problem, she quickly discovers, is that a group of metal filing cabinets were installed yesterday. They are partially blocking the access point's signal. There are other access points available, so this is not a problem, but she makes a note in her daily log.

| SSID | FBP | FBP | FBP |
|---|---|---|---|
| BSSID | A1-B2-C3-D4-E5-F6 | BB-D5-33-D4-6B-DD | 19-FF-AE-D4-EC-63 |
| Signal | −85 dBm | −60 dBm | −60 dBm |
| Mode | 802.11g | 802.11n | 802.11n |
| Channel | 11 | 48 | 44 |
| Encryption | AES-CCMP | AES-CCMP | AES-CCMP |
| Authentication | WPA2/PEAP | WPA2/PEAP | WPA2/PEAP |
| Vendor | Cisco | Cisco | Cisco |

**FIGURE 1-5** Wireless Sniffer Details Table

Near the end of her "walkabout," Claire finds an access point that should not be there. In security terminology, it is a **rogue access point**. Using the map function, she traces it to a conference room. However, the conference room looks clean. She checks in cabinets and finds nothing. Then she notices a network cable and power cable leading up to the ceiling. She climbs up on the table and lifts a tile in the false ceiling. There it is. She disconnects it and turns it off. She will give it to security.

*A rogue access point is an unauthorized access point.*

The bank will soon be switching over to a centralized wireless management system. Instead of having to walk around, Claire will just check a display. It will give her the information she now collects so laboriously in a single integrated database.

- It will identify rogue access points and access points outside the building automatically, sending her a text when it finds one.
- It will be able to do several things she cannot do now, such as identifying access points that are not operating during one of her walkabouts.
- She will even be able to adjust access point power levels, channels, and other parameters remotely. When she finds a problem, she will usually be able to fix it without wasting time going to the access point.
- She will even be able to adjust the access points when groups of people with smartphones, tablets, and notebooks assemble in a location that was previously quiet.

Claire will need the boost in productivity the wireless management system will bring. Wireless traffic in the building is exploding, and the growth of video applications is requiring her to give increasing attention to error rates and other performance parameters that have not been issues with traditional applications.

**Test Your Understandings**

3. a) What information could Claire learn about individual access points? b) Distinguish between SSIDs and BSSIDs. c) What is a rogue access point? d) Why do you think rogue access points are dangerous? e) Why is centralized wireless management highly desirable compared to "management by walking around" as Claire does today?

### John Lee

John Lee is a member of the First Bank of Paradise IT security staff. Like Claire Lorek, he focuses on the bank's internal wireless LAN. While Claire works to keep the network running, John's job is to keep the users safe (and to keep the bank safe from users).

One of John's projects is to develop security policies for user-owned smartphones and tablets. (There are already security policies for user-owned laptops and desktops.) The main problems that John faces are the diversity of operating systems and the rapid pace of change in technology. There are three major operating systems to keep track of—Apple's iOS, Microsoft's Windows 7 and Windows 8, and Android. Apple and Microsoft keep tight control over their operating systems. The diversity among Android operating system variants, however, is daunting. Just keeping track of security vulnerability reports on these operating systems and variants takes up much of John's day. With the technology changing at dizzying speed, operating systems and hardware are generally maturing, but while each advance seems to bring better security in general, it also brings a few new security problems. John refers to the situation as the **BYOD (bring your own device)** issue.

One issue that concerns John is the loss of smartphones and tablets. Many of these devices are loaded with private customer information, private information about other employees, and trade secrets such as price and customer lists. John wants to find secure encryption and device locking tools that will be strong enough for protection and easy enough to use for phone and tablet users to accept. Right now, however, he can only alert users to the problem and keep his fingers crossed.

**Test Your Understanding**

4. a) List major wireless LAN security issues. b) Why is BYOD security so difficult today?

### The Rogue Access Point

Remember the rogue access point that Claire discovered? John traced it to Albert Gomes, a mid-level manager. Gomes was surprised that his access point had been detected because he had set the access point to "stealth mode." Claire and John replied that their software and hacker software have no problem finding such access points. Gomes replied that he thought stealth mode would keep the company safe.

Claire and John were not interested in making this an inquisition. They asked about the problem that Gomes was trying to solve with his rouge access point. It turns out that he is doing HDTV training in his department and often has two or three people taking training at a time. Claire agreed that the department needed more speed and promised that she would provide it by the next day. She also pointed out that his access point was not capable of implementing quality of service guarantees, which his employees needed for smooth video delivery. Her system would fix that. Greatly relieved and pleased, Albert thanked them enthusiastically. Networking and security had made an important friend and had done their job of using networks to make the company work better.

### Networking and Security

Everything is changing in information technology, and it is changing at light speed. Few things are changing more rapidly than networking and security. These are fields for people who do not like boredom, and corporations almost always put these two disciplines near the top of their IT hiring lists.

Although networking and security are technically distinct fields, in practice they are converging rapidly. Networking professionals, who used to be concerned with wires and switches, now find that they need to consider security in everything they do. Security professionals, in turn, find that most of their vexing problems are concerned with networking and require extensive networking knowledge. The title of this book, *Business Data Networks and Security*, reflects this convergence.

### Test Your Understanding

5. Why does this book combine networking and security?

## BASIC NETWORK TERMINOLOGY

### Networks, Hosts, and Applications

In a book on networking, it makes sense to begin by defining what a network is. As a working definition, we will define a **network** as a system that permits applications on different hosts to work together.

---

*As a working definition, a network is a system that permits applications on different hosts to work together.*

---

Figure 1-6 illustrates this working definition.

In Chapter 2, we will extend this working definition into a formal definition using concepts that are not necessary for a broad understanding of networks.

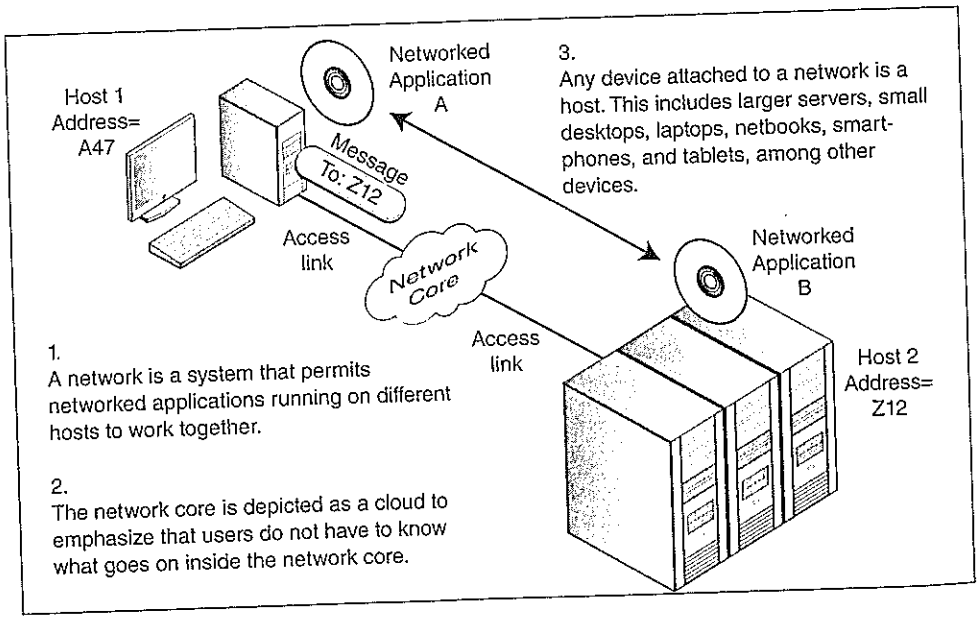


**FIGURE 1-6** Basic Network Terminology

**NETWORKED APPLICATIONS** Note that the definition focuses on *applications*. Although you will spend most of your time in this course learning the details of how networks operate, our preliminary definition takes the *user's* point of view. Networks are only valuable if they connect the applications that users need to have connected. Users have no desire to know the details of how the network operates. Nor *should* they. They simply want to use their applications as easily as they can. Applications are the only things users care about and should have to know about.

Computer applications existed long before networking. However, as networks began to connect different computers and to do so with greater reliability and speed, there was an explosion in **networked applications**, which require networks to work.

*Networked applications are applications that require networks to work.*

The 1970s brought the first networked applications, including e-mail for communication and the File Transfer Protocol (FTP) for moving large files. Networked applications continued to evolve steadily, but there was an explosion of applications in the 1990s, driven by the rapid growth and commercialization of the Internet, that brought the World Wide Web (WWW), e-commerce, social networking, and many of the other networked applications we use so extensively today.

This century has brought **Web 2.0** applications, in which users provide the content. In early (Web 1.0) Web applications, the website owner created the content. This was expensive and gave website owners control of what appeared on the World Wide Web. With Web 2.0, these barriers are gone. Users create most or all of the content. Wikipedia is a classic example of a Web 2.0 application. Although Wikipedia has some problems, this "crowd-sourced" application is by far the most comprehensive encyclopedia the world has ever known. On a less august level, there is YouTube.

*In Web 2.0 applications, users provide the content.*

Included in the Web 2.0 category are **social media applications**, such as Facebook, Twitter, and dating services, which are designed to facilitate relationships. E-mail and other communication applications have long been useful in building relationships, but social media sites have developed specialized tools to enhance social networking. Facebook works on the individual level, connecting a person with several of his or her friends. Other new applications bring together groups of people with common interests.

*Social media applications are applications designed to facilitate relationships.*

**HOSTS AND ADDRESSES** In the definition of networking, it was noted that applications run on devices called *hosts*. We use the term *host* rather than *computer* because not all devices connected to networks are computers in the traditional sense. Smartphones and tablets are good examples of this. In the future, toasters and coffee pots will be

connected to networks within individual homes and across the Internet.[2] Formally, we will define a **host** as any device connected to a network.

*A host is any device attached to a network.*

Each host needs a unique **network address**. When a source host sends a message to a destination host, the source host places the address of the destination host in the message. Based on this address, the network delivers the message to the destination host. The source host does not have to know how the message gets to the destination host—just that it does. Analogously, when you place a call, you simply dial a number and the telephone network invisibly connects you to the other telephone without your knowing how this occurred.

*Each host has a unique network address. Based on this address, the network delivers the message to the destination host.*

**THE NETWORK CORE** For users, a network is simply there. Consequently, Figure 1-6 depicts the **network core**—the central part of the network—as a cloud. Just as you cannot see the inside of a cloud, users do not have to look inside the network.

*The network core is the central part of the network.*

Of course, as an information systems student, you will need to look *deeply* inside various types of network technologies ranging from the network you may have in your home to the global Internet. Before my grandfather came to America, he was told that the streets of America were paved with gold. When he got here, he found that they were not even paved. Guess who was going to pave them?

Fortunately for you, the network "streets" in corporations today are already reasonably well paved. Your job will be to turn them into true super highways and to help your organization assimilate the ever-growing number of networked applications and novel types of applications that network capabilities will make possible. You will also help users cope with the outpouring of new applications and devices. On the negative side, you will have to deal with a large and rapidly growing number of security threats that threaten the potential benefits of networking.

**ACCESS LINKS** Users connect to networks through **access links**, which may use copper wire, optical fiber, or radio transmission. Users typically need to know a little more about access links than about the network core. For example, they may have to plug

[2]In fact, there already is an *Internet Coffee Pot Control Protocol* (RFC 2324) for remotely managing coffee brewers. The standard was created on April 1, 1998. In America, the first of April is April Fools' Day, which is a traditional day for pranks. Although the Internet Coffee Pot Control Protocol was a joke standard, the future will hold many similar but serious protocols.

in access link technology, configure it, and troubleshoot simple problems. However, required knowledge should be reduced as far as possible.

**Test Your Understanding**

6. a) Give the book's definition of *network*. b) What is a networked application? c) What are Web 2.0 applications? d) What are social media applications? e) What is a host? f) Is your laptop PC or desktop PC a host? g) Is a smartphone a host? h) Why is the network core shown as a cloud? i) Why may the user need to know more about his or her access link than about the network cloud?

## Application Interactions

Traditionally, programmers wrote applications that ran on a single computer. Today, most programmers need to be able to write applications that work with other applications on other computers on other networks. How to divide the workload between the multiple computers involved in interactions is an important concern.

**CLIENT/SERVER PROCESSING** Networked applications on different computers have to interact in a disciplined way. Today, they normally interact through **client/server processing**, in which a **server program** on a **server host** provides service to a **client program** on a **client host**.

*In client/server processing, a server program on a server host provides services to a client program on a client host.*

You use client/server processing every day because this is how the World Wide Web (WWW) operates. Figure 1-7 illustrates how client/server processing works when you use the Web. Here the application protocol is the Hypertext Transfer Protocol (HTTP). Your PC is the client host, and the webserver is the server host. The webserver host runs webserver application software, while your PC runs a client application program, namely your browser. The webserver application provides service to your browser by sending you the webpages you request.

Although browser–webserver applications are very common, the server is not always a webserver, and the client program is not always a browser. In database

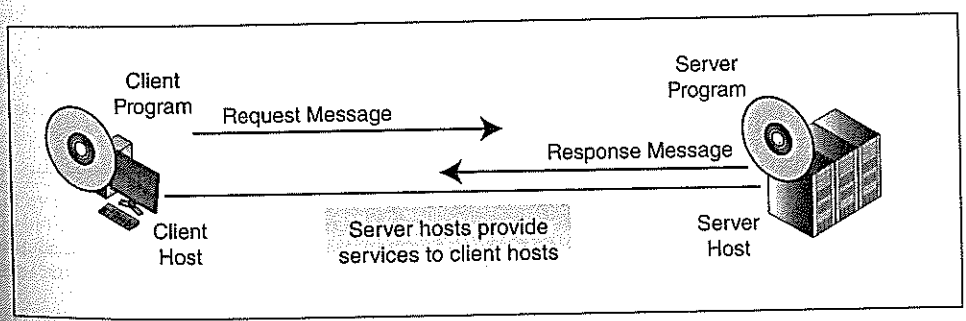


**FIGURE 1-7** Client/Server Processing

processing, of course, the server is a database server. The client PC, in turns, often runs a dedicated database client program.

Client/server processing typically operates using a **request–response cycle.** The client sends a **request message** to the server. The server sends back a **response message.** For instance, when you type a URL or click on a link, your browser sends a request message to the webserver. This request message specifies a file to be downloaded. The webserver sends back a response message containing the specified file (or an error message). If the user downloads another webpage a few seconds or minutes later, this is a separate request–response cycle.

Client/server computing is enormously important today, and most programmers will spend much of their careers writing them. Writing client/server programs effectively requires a good understanding of how networks work in order to minimize interaction times and to troubleshoot problems.

**PEER-TO-PEER PROCESSING** Although client/server processing is important, it is not the only way to allocate work among computers. An emerging way for applications to interact is **peer-to-peer (P2P) processing,** in which client hosts provide services directly to other client hosts. Figure 1-8 shows a P2P interaction between two nearby mobile phones. In this case, the two phones communicate directly. There is no network between them.

*In peer-to-peer (P2P) processing, client hosts provided services directly to other client hosts.*

Peer-to-peer processing can also work over a network. Many people illegally share music and movies through P2P services on the Internet, and this has given networked P2P processing a bad name. However, companies are beginning to recognize that their client PC desktops represent an enormous mine of unused processing and storage power. Companies are developing ways to use this power instead of buying more servers. We will look at P2P applications in Chapter 11.

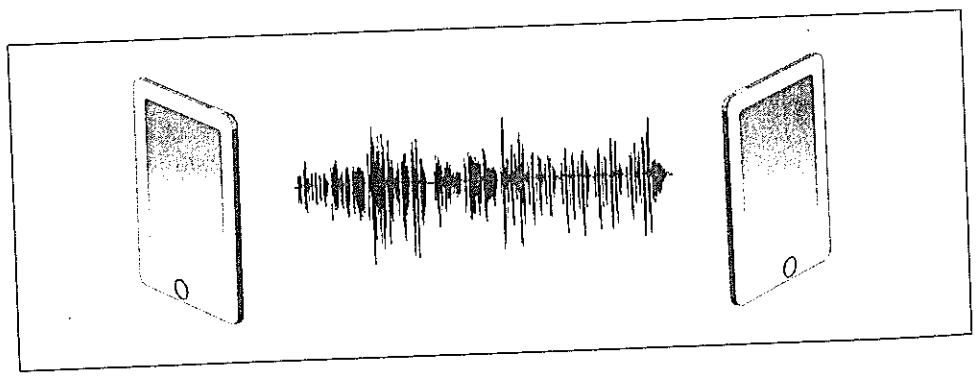

**FIGURE 1-8** Peer-to-Peer Computing

| Transmission Speed Measurements | | |
|---|---|---|
| Bits per second (bps) | | |
| Usually not bytes per second (Bps) | | |
| **Metric Suffixes** | | |
| Kilobits per second | kbps (lowercase k) | 1,000 bits per second (not 1,024) |
| Megabits per second | Mbps | 1,000 kbps |
| Gigabits per second | Gbps | 1,000 Mbps |
| Terabits per second | Tbps | 1,000 Gbps |

**FIGURE 1-9** Transmission Speeds

## Speed

The first question people ask about a newborn baby is, "Is it a boy or girl?" The first question people ask about a network is, "How fast is it?" Network speeds[3] are measured in **bits per second (bps)**. Note that this is *bits* per second, not *bytes* per second. Occasionally, when you download files, your software will tell you how many bytes per second are being downloaded. However, that is an exception. In such cases, the B usually is capitalized, so that *bytes per second* is abbreviated as **Bps**.

*Network speeds are measured in bits per second (bps), not bytes per second (Bps).*

Figure 1-9 shows that, in increasing factors of one thousand, there are **kilobits per second (kbps)**, **megabits per second (Mbps)**, **gigabits per second (Gbps)**, and **terabits per second (Tbps)**.

In the metric system, numbers are expressed in **base units**, such as bps, preceded by a **metric prefix** that multiplies the base unit. Without a metric prefix, five million bits per second would be 5,000,000 bps. With the metric prefix for mega (M), it is 5 Mbps. To give another example, 7.2 kbps without its metric prefix is 7,200 bps.

Note in particular that metric prefixes for speeds increase in factors of 1,000—not 1,024. Factors of 1,024 are used for computer memory and storage.

Note also that the abbreviation for kilobits per second is *kbps*, not *Kbps*. In the metric system, kilo is a lowercase k. (Capital K is for Kelvins, which is a measure of temperature.)

How fast does a network *need* to be? The answer depends upon the sizes of messages transmitted. Figure 1-10 shows download times at various transmission speeds for some applications you might use yourself.

[3] Strictly speaking, speed is a poor description because it implies velocity. Signals always travel at the speed of light through propagation media, regardless of how many bits are sent. We really should say rate, which is like the amount of water flowing through a pipe. Looked at another way, velocity is like running faster. More bits per second is like talking faster.

| Applications | 10 kbps | 100 kbps | 1 Mbps | 5 Mbps | 10 Mbps | 100 Mbps | 1 Gbps |
|---|---|---|---|---|---|---|---|
| File Transfers | | | | | | | |
| Text e-mail message (250 words) | 1.5 s | 0.15 s | 0 s | 0 s | 0 s | 0 s | 0 s |
| Photograph (5 MB), E-Mail with 5 MB attachment, or media-rich webpage | 83 m | 8 m | 1 m | 10 s | 5 s | 1 s | 0.1 s |
| Download 1 Hr. HDTV Video (10 Mbps) | 42 d | 4 d | 10 h | 2 h | 1 h | 6 m | 36 s |
| Backup/File Synchronization (10 GB) | 116 d | 12 d | 28 h | 6 h | 3 h | 17 m | 2 m |
| Live or streaming media | | | | | | | |
| MP3 Song (10 kbps) | OK | OK | OK | OK | OK | OK | OK |
| Standard-quality TV (2 Mbps) | | | | OK | OK | OK | OK |
| HDTV (10 Mbps) | | | | | OK | OK | OK |

**FIGURE 1-10** Download Times for Various Applications at Various Throughput Rates

- An e-mail message of 250 words will be downloaded instantly at any current network transmission speed.
- A photograph or e-mail with a large attachment may be 5 GB in size. It is not until 10 Mbps that download becomes only a few seconds, and it even takes a second at 100 Mbps.
- How long will it take to download a one-hour HDTV video? Even at 5 Mbps downstream throughput, it will still take 2 hours to download. In fact, download time does not fall to a reasonable 36 seconds until a gigabit per second.
- We would like to do backup and file synchronization quickly, to encourage people to do it. Assuming 10 GB per backup or synchronization, things are still slow at 100 Mbps and are still 2 minutes at 1 Gbps.
- For an MP3 music file compressed to 10 kbps, all current transmission speeds are sufficient.
- TV requires even faster download speeds. Even with normal-quality television, which requires about 2 Mbps for real-time streaming, transmission speed has to be better than 1 Mbps for good screening.
- For high-definition TV, 10 Mbps is a typical transmission requirement for real-time streaming. Given HDTV's sensitivity to adequate throughput, even faster speeds are important.

How much speed do users need, then? Most applications work well at 1 Mbps, but 10 Mbps or higher is necessary for common downloads or streaming HDTV. For downloading HDTV videos and movies and for very rapid backup and file synchronization, speeds of 100 Mbps to 1 Gbps become highly desirable. Of course, these are speed to individual users. To deliver such speeds, speeds in the network core must be far higher.

Test Your Understanding

7. a) Are network speeds usually measured in bits per second or bytes per second?
   b) How many bits per second (without a metric prefix) is 20 kbps? Use commas.
   c) How many bits per second (without a metric prefix) is 7 Mbps? Use commas.
   d) How many bits per second (without a metric prefix) is 320 kbps? Use commas.
   e) Is the metric prefix for kilo k or K? f) Express 27,560 bps with a metric prefix.

## PACKET SWITCHING AND THE ARPANET

So far, we have been looking at networks as they exist today. In this section, we will begin looking at how we got here.

### Larry Roberts Has a Burstiness Problem

During the 1960s, the U.S. Department of Defense's **Advanced Research Projects Agency (ARPA)**[4] funded a great deal of basic research across the United States. While some of this research had direct military application, most funding was for basic science and technology.

Dr. Larry Roberts was head of ARPA's Information Processing Technology Office (IPTO). IPTO funded a great deal of software research. In those days, software usually was not portable from one computer to another. If someone needed to use a piece of software, he or she would need a computer terminal and a transmission line to the host computer running the software.

With transmission lines from the telephone company, you have to pay by the minute for service, regardless of whether you are using the line or not. Figure 1-11 shows that this is not a problem for human conversations because long gaps in telephone conversation are rare. Typically, only one side talks at a time, and there are always brief silences. However, normal voice conversations use the capacity of a two-direction telephone line 30 percent to 40 percent of the time.

In contrast to telephone conversations, data transmission is **bursty**, which means that there are brief bursts of traffic followed by long silences. To see this, consider what happens when you visit a website. To download a webpage, you send a request and get back a response containing the webpage. All of this usually takes a second or less. Now, you probably will look at the webpage for about 30 seconds. (Count it sometime!) This means that you are only using one-thirtieth of your circuit—about 3 percent. So while paying by the minute is only somewhat wasteful for voice, it is *very* wasteful for data.

*Data transmission is bursty, which means that there are bursts of traffic separated by long silences. This is very wasteful if you are using reserved-capacity circuits.*

In addition, Figure 1-11 shows data traffic with large boxes, to indicate that data bursts need to be large. When you download a webpage, you do not want it to dribble

[4] Is it ARPA or DARPA? It depends on the year. It was born ARPA in 1958. In 1972, it became DARPA to emphasize its status as a Department of Defense agency. In 1993, it went back to ARPA. Then it went back to DARPA in 1996. DARPA, "ARPA-DARPA: The Name Chronicles," undated. http://www.darpa.mil/About/History/ARPA-DARPA__The_Name_Chronicles.aspx. Last viewed January 2012.

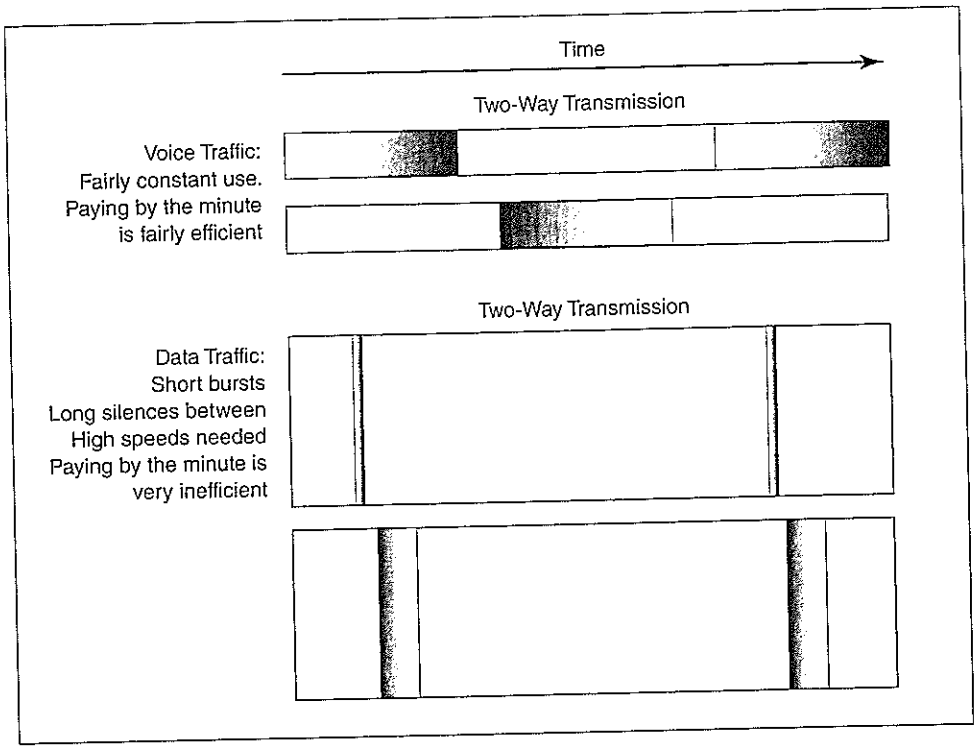


**FIGURE 1-11** Data Burstiness

down. You want it right now. This requires a much faster line for data than for voice. Of course, the per-minute cost of a very fast transmission line is much higher per minute than an ordinary voice telephone line. This exacerbates the problem of burstiness.

### Test Your Understanding

8. a) Why is paying for a transmission line by the minute not too bad for voice conversations? b) For what two reasons is paying for a transmission line by the minute bad for data transmissions?

## Packet Switching Presents a Possible Solution

**PACKET SWITCHING** During the 1960s, several researchers identified a solution for the inefficiency of pay-by-the-minute transmission lines for bursty data transmission. This solution was called packet switching. Figure 1-12 shows how packet switching works.

In the figure, an application on Host A wishes to send an application message (Original Message AC) to an application on Host C. The figure shows that Host A fragments the message into many smaller segments and sends each in a separate message called a **packet**. A typical packet is about 100 bytes long. Even sending very brief e-mail messages may require two or three packets. Longer documents, graphics files, audio messages, and video files may be sent in hundreds or thousands of packets. However many packets are sent, the network delivers them to Host C. The destination host reassembles the packets and passes them to the destination application program.

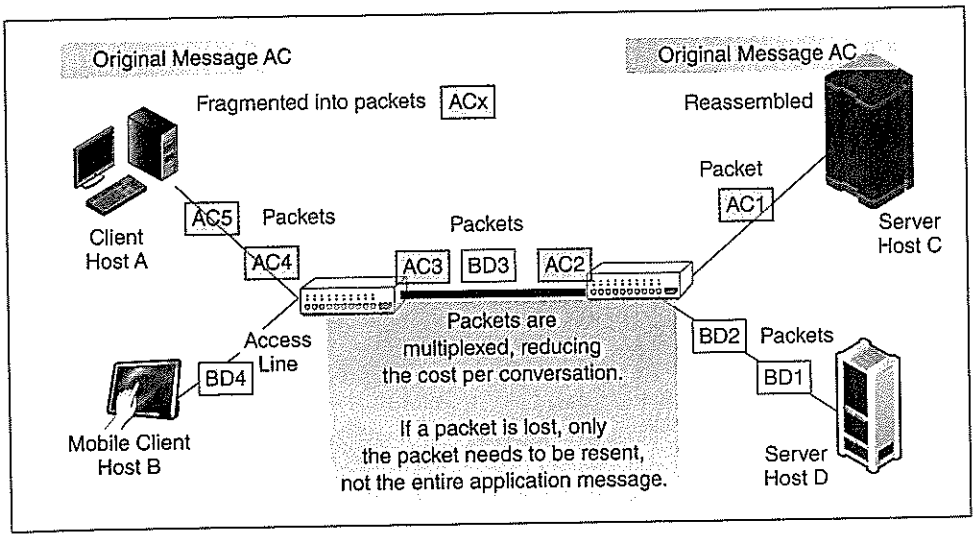


**FIGURE 1-12** Packet Switching and Multiplexing

*In packet switching the source host fragments the application message into many smaller pieces called packets; the network delivers the packets to the destination host, which reassembles the application message.*

Figure 1-12 shows that the packets of multiple conversations can be **multiplexed** (mixed) on long-distance circuits. If each conversation between hosts is using only 3 percent of capacity, then, roughly speaking, about 30 conversations can be multiplexed onto the circuit. Packet switching thus saves money by multiplexing multiple conversations over expensive circuits. Bursty traffic has to pay only for the capacity it actually uses.

*Packet switching saves money by multiplexing multiple conversations over expensive circuits.*

The figure lists another benefit. If there is a transmission error that destroys a packet, only the lost packet needs to be resent. Early transmission lines had substantial error rates, and when whole messages were sent, messages might have to be resent several times before being received correctly.

**PACKET SWITCH SWITCHING DECISIONS** Figure 1-13 looks at the packet switches that make this work. Here, there are six packet switches, imaginatively labeled A through F. The source host transmits a packet to Packet Switch A—the switch to which the source host connects directly. Packet Switch A has to make a **switching decision**. It must decide where to send the packet next. It can either forward the packet to Packet Switch B or send it to Packet Switch C. Packet Switch A decides that Packet Switch B is a better choice for getting the packet to Destination Host Y, so Packet Switch A will forward the packet to Switch B.

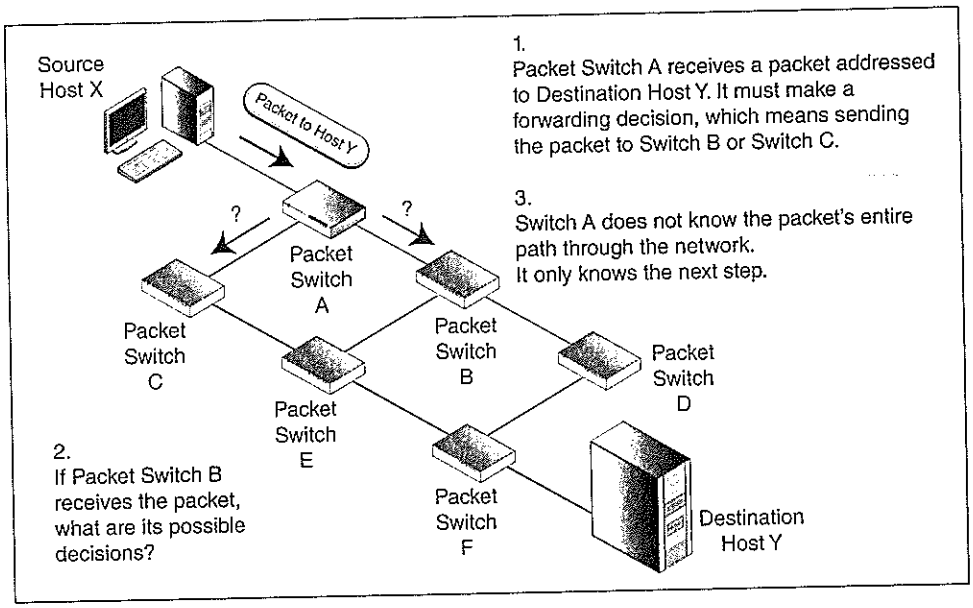


**FIGURE 1-13** Sequential Switch Switching Decision

*Each packet switch makes a switching decision, which means deciding where to send the packet next.*

Now the packet is at Packet Switch B. Packet Switch B has to make its own switching decision. It can forward the packet to Packet Switch D or to Packet Switch E. In this case, it decides to forward the packet to Packet Switch D. Again, it made this decision knowing only about the switches to which it directly connected—Switch D and Switch E.

Note that neither Packet Switch A nor Packet Switch B knows the entire path the packet takes through the network. Each makes a local decision; it only decides where to send the packet *next*. It does not matter how many hops there are across packet switches from the source host to the destination host.

*Individual packet switches do not know the packet's entire path through the network. They only make a local switching decision in which they decide where to send the packet next.*

Why did Packet Switch A decide to forward the packet to Packet Switch B rather than Packet Switch C? Figure 1-14 gives the answer. It shows that switches have switching tables. Each row specifies a destination host and the next hop for the packet. For Destination Host Y, the next-hop column specifies Packet Switch B. The switch therefore sends the packet on to Packet Switch B.

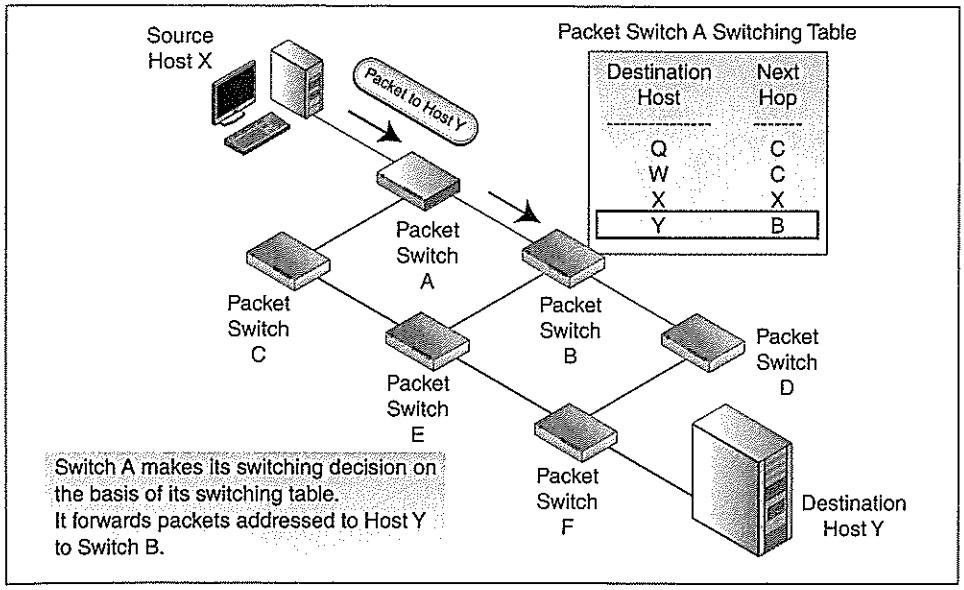


**FIGURE 1-14** Address-Based Packet Switch Forwarding Decision

### Test Your Understanding

9. a) In packet switching, what does the source host do? b) About how long is a packet? c) Why is fragmentation done? d) Where is reassembly done? e) What are the two benefits of multiplexing? f) When a packet switch receives a packet, what decision does it make? g) Do packet switches know a packet's entire path through a network? h) If Packet Switch A receives a packet addressed to Destination Host W, where will it send the packet?

## Physical Links and Data Links

In a packet-switched network, there are two types of links or transmission paths. Figure 1-15 illustrates these links. First, there are **physical links** between hosts and their switches and between switches. These physical links along the path the packet takes between the two hosts may use different technologies.

There also needed to be a name for the path across switches that a packet takes between the source host and the destination host. The term **data link** was selected for this path. When one host transmits a packet to another host, there may be multiple physical links along the way, but there is only one data link.

### Test Your Understanding

10. a) In Figure 1-15, how many physical links are there between the source host and the destination host along the indicated data link? b) How many data links are there between the source host and the destination host? c) If a packet passes through eight switches between the source and destination hosts, how many physical links will there be? (Careful!) d) How many data links will there be?

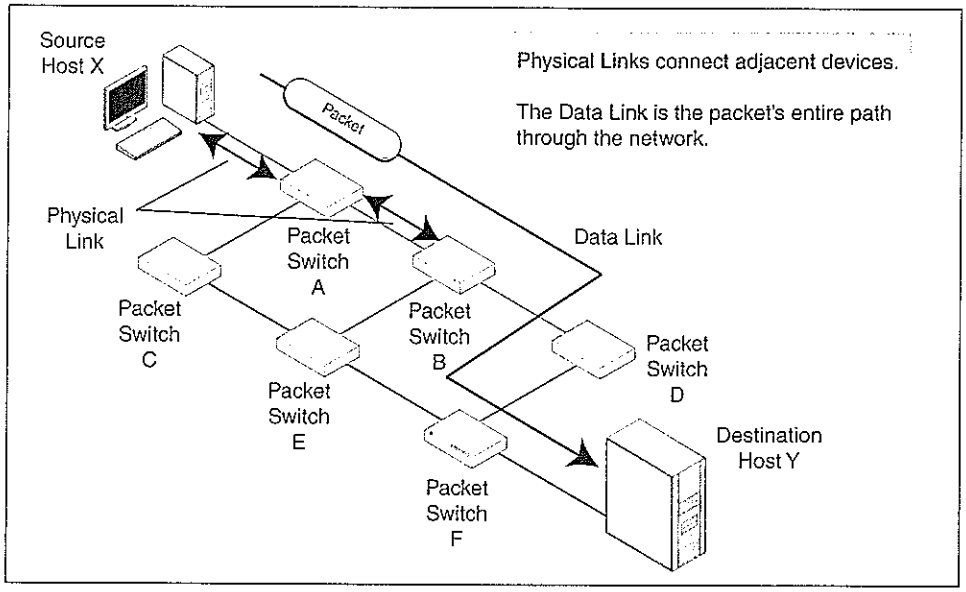


**FIGURE 1-15** Physical Links and Data Links

## Larry Roberts Builds a Solution

Larry Roberts saw packet switching as a way to reduce the cost of connecting remote terminals to the hosts that ran software funded by ARPA. He also saw it as a research project. Packet switching seemed to have great potential. To see if that potential was real, someone had to build a packet-switched network.

Roberts funded the creation of a packet-switched network called the ARPANET, which Figure 1-16 illustrates. Most of the network chores were handled by minicomputers called **interface message processors (IMPs)**. An IMP received a message from a host attached to it, broke the message into small pieces, and placed each piece in its own packet. The IMPs then acted as packet switches, forwarding the packet to an IMP nearer to the destination host. The final IMP reassembled the original message and passed the message to the destination host. Note in the figure that an IMP can serve multiple hosts. A small team at Bolt Beranek and Newman programmed the IMPs.

Each host ran software called the **Network Control Program (NCP)**. This handled details of host-to-host interactions above the level of packetization, delivery, and reassembly. For instance, if the source host transmitted too rapidly for the destination host to process, the NCP on the destination host could tell the NCP on the source host to slow down.

In 1969, the first two IMPs were installed at UCLA in Los Angeles and at Stanford Research Institute (now called SRI International[5]). As soon as that link was established, the

[5]Protests during the Vietnam War caused Stanford University to sever ties with Stanford Research Institute. The institute decided to call itself SRI, but there was already a small consultancy with that name, so Stanford Research Institute became SRI International. However, faculty and graduate students continued to move frequently between the university and the institute. (In one case, a PhD student got a contract worth about $200,000 in today's dollars to do his dissertation.)

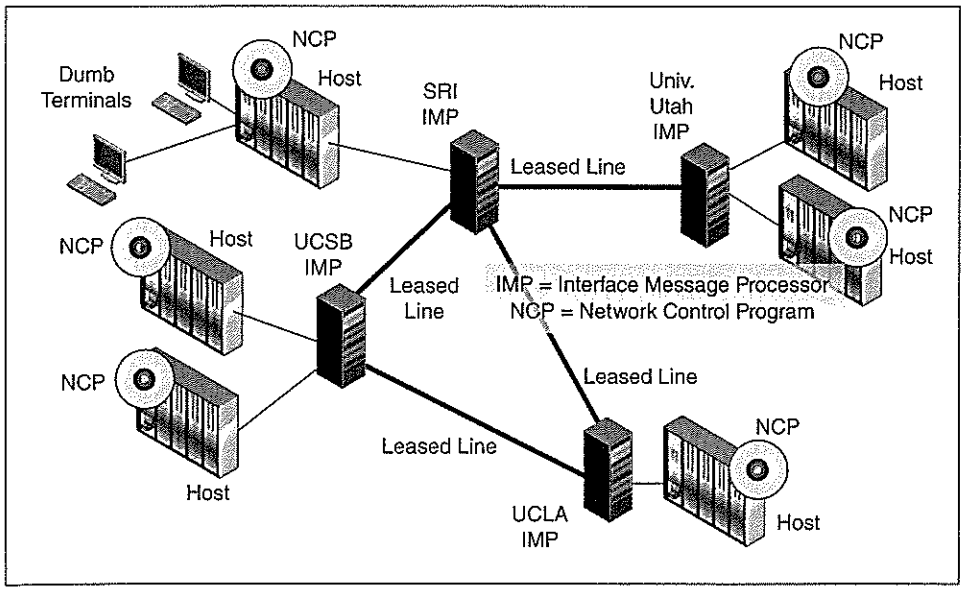


**FIGURE 1-16** The ARPANET

University of California at Santa Barbara and the University of Utah were added. When it was clear that NCP and the IMPs were working as intended on these four sites, other sites were added, and some sites began to connect multiple computers to their IMPs.

Although the ARPANET was officially "born" in 1969, its performance was flakey until the early 1970s. More fundamentally, there were no applications at all beyond those that ran on hosts for terminal users. Even in 1973, when the first author used the ARPANET, reliability and services were quite limited. Over time, however, reliability improved and more applications appeared. Users stopped calling the ARPANET the "notwork."

### Test Your Understanding

11. a) On the ARPANET, explain the functions of IMPs. b) How is this like what packet switches do today? c) How is it more than packet switches do today?

## The Network Working Group

Each of the first four sites had to devise its own way to connect its hosts to its IMP. This had to change. To make the new ARPANET useful, fundamental standards were needed for host–IMP connections. There also needed to be standards for applications, such as e-mail.

Nobody was filling this vacuum for standards, so graduate students from the four initial schools got together to create an informal **Network Working Group**. Led by Steve Crocker at UCLA, the Network Working Group began to create necessary standards. The group had no charter to develop standards, so the group members decided not to call their designs *standards* but rather **Requests for Comment (RFCs)**. Steve Crocker wrote RFC 1 to describe needs for software on two hosts to communicate. Over time, the Network Working Group grew considerably, but it kept its informal flavor. Today,

this work is carried on by the **Internet Engineering Task Force (IETF)**, which creates standards for the Internet. It still calls its standards RFCs.

**Test Your Understanding**

12. a) What organization sets Internet standards today? b) What does the IETF call its standards?

## E-Mail

Even before the ARPANET, users on the same host computer had e-mail. When you logged in, you received any messages that other users on that computer had left for you. When the ARPANET appeared, Ray Tomlinson at Bolt, Beranek, and Newman saw the opportunity for users to send mail to other users on different machines. He decided to adapt the local SNDMSG program on his local host for message delivery over the ARPANET. To send a message, you would type SNDMSG and hit Enter. You would then enter a value for To: and then type the body.

On a single computer, your e-mail address was simply your username. However, across multiple computers, usernames were not unique. To be unique, addresses needed a combination of username and host designation. For example, to get an e-mail to Ra3y, you would have to specify that this is a username on the host Office1. Tomlinson looked at his keyboard and saw a character that was hardly ever used. This was the "at" sign, @. So to reach Ray Panko at the Office1 computer at Stanford Research Institute, you sent the mail to Ra3y@Office1.

Now, Tomlinson had to write the e-mail program for host computers. He was not being paid to do this, so he did it on his own time, over a weekend. Networked e-mail was born. However, his program, SNDMSG, was only a message *sending* program. The receiver could only read incoming messages one at a time. One intense early e-mail user was Larry Roberts. He often received dozens of messages per day. Initially, he had to go through his messages one at a time. On his own time, he wrote the first useful e-mail *reading* program, which he called RD. Soon, other ARPANET users created better e-mail reading programs, such as bananard. (It was the 1970s.) This kind of do-it-yourself spirit and an absence of any profit motive made the ARPANET a great development environment.

ARPANET e-mail was also a social networking tool. ARPA funded a great deal of computer science research in the 1960s and early 1970s. People in different organizations funded by ARPA often knew each other well. Individual e-mail messages and computer conferencing message groups increased the cohesion of this research community. For example, the Message Service Group was a mailing list for people who engaged in discussions about issues in e-mail.

In 1974, a young researcher was visiting ARPA. In a meeting, he asked about the cost of an e-mail message. Nobody gave an answer. Later that day, a senior official pulled the visitor into an office and told him to stop asking about the price of e-mail. He said that e-mail's cost was about $60 per message, and e-mail made up 75 percent of the ARPANET's traffic. Government auditors would scream if they knew this, and ARPA would have to come down hard against e-mail use. The researcher said that that $60 per message was impossible. They sat down and poured over the data for several hours. It turned out that the average message cost about as much as a postage stamp to deliver across the network. In the next few months, ARPA's concerns about e-mail relaxed, and ARPA began to advertise its success in connecting people by e-mail.

Test Your Understanding

13. a) How did Ray Tomlinson extend e-mail? b) How did he change e-mail addresses?

## THE INTERNET

### Bob Kahn Has a Problem

By the early 1970s, the ARPANET was (reasonably) stable. In addition, packet switching was proving itself in wireless environments. The ALOHANET project at the University of Hawai`i demonstrated that packet switching could be done over satellite circuits, despite the long time lags in upward and downward transmission. Terrestrial (earth-bound) packet radio projects also began to appear, using backpack radios.

Dr. Bob Kahn, who was by now in charge of IPTO at ARPA, was happy to see packet switching blossoming. However, he also saw a deep problem. Users on the ARPANET, the PRNET packet radio network, and the SATNET packet-switched satellite network could not communicate with each other. Packet-switched networking was becoming a Tower of Babel.

Test Your Understanding

14. What problem did Bob Kahn face?

### Bob Kahn and Vint Cerf Find a Solution

Kahn discussed the problem with a young Stanford professor, Vint Cerf. Together, Kahn and Cerf explored various solutions to the problem of internetworking. Their final solution, which Figure 1-17 illustrates, was to use special devices called **gateways** to connect different networks together into an "internet." Today, we call these devices **routers**.

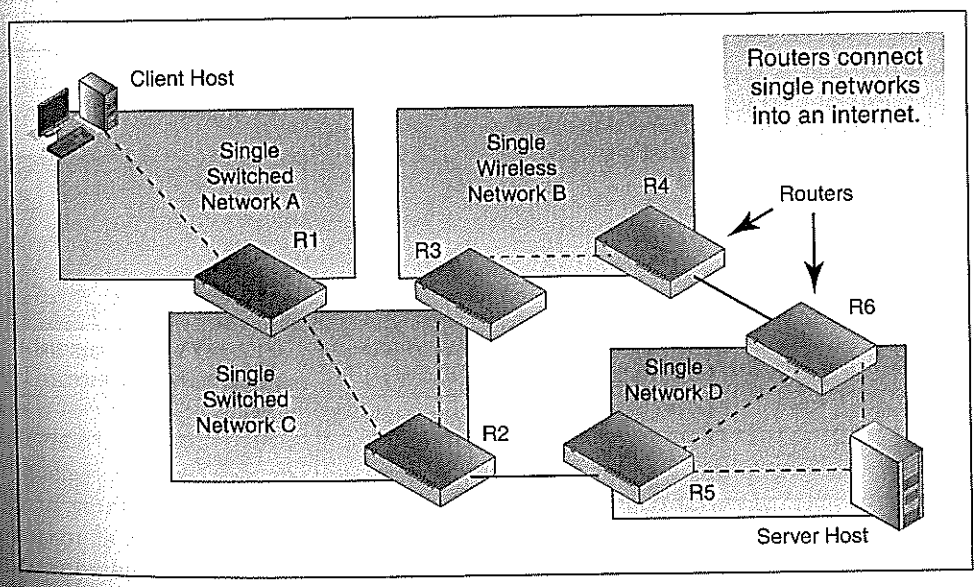


**FIGURE 1-17** Internetworking

*Routers connect different networks together into an internet.*

Test Your Understanding

15. a) What device connects different networks into an internet? b) What is the old name for this device?

## A Second Layer of Networking

**CAUTION:** The following material is difficult because the creation of a second level of networking creates many pairs of concepts that are similar but different.

One problem in connecting different types of networks is that they use incompatible data link technologies. They organize packets in different ways, and they have different and deeply incompatible addressing systems. To overcome such problems, Kahn and Cerf essentially created a higher layer of networking over all of these individual incompatible networks technologies. We will call this higher layer the **internet layer**. Figure 1-18 shows that their solution needed to do more than just create a second layer of networking. It also had to duplicate the concepts of addresses, packet switches, packets, and paths. Unfortunately, this causes a great deal of confusion for students.

Another problem with having two levels of network concepts is that terminology for packet switching matured in confusing ways. Figure 1-18 shows this confused terminology.

**WARNING:** The following distinctions are critical. If you don't get them down firmly, your life in this class will be absolutely miserable. They are a bit confusing, but they are not at all difficult.

| Component | Generic Terminology (Not Used in Actual Networks) | Single Networks | Internets |
|---|---|---|---|
| Addresses | | Vary by network technology | 32-bit IPv4 Addresses and 128-bit IPv6 Addresses |
| Packets are called | Packets | Frames | Packets |
| Packet switches are called | Switches | Switches | Routers |
| End-to-end routes are called | | Data links | Routes |

**FIGURE 1-18** Two Layers of Networking

- The generic term packet switching, with messages called packets and forwarding devices called switches, is not actually used in real networks.
- In single networks, packets are called frames, and forwarding devices are called switches.
- In internets, packets are called packets, but forwarding devices are called routers.

Overall, then, there are frame switches and packet routers, but there are no packet switches per se in the real world.

**POINT OF TERMINOLOGY** "Internet" is both a general concept and a name for the largest internet of all, the global Internet. To make distinctions, capitalization will be reserved for the global *Internet* (and of course when internet is capitalized in titles and at the start of sentences). An *internet* (in lowercase) is any internet. Lowercase is also used to refer to the *internet layer*.

*Capitalization will be reserved for the global Internet (and of course when internet is capitalized in titles and at the start of sentences). An internet (in lowercase) is any internet. Lowercase is also used to refer to the internet layer.*

**A SECOND LEVEL OF ADDRESSES** Different network technologies use different syntaxes for addresses. For example, Ethernet uses 48-bit MAC addresses, while Frame Relay uses 10-bit DLCI addresses. In addition, these addresses were not always assigned uniquely. Nearly all Frame Relay networks, for example, had a host represented by DLCI 1.

For the second layer of networking, Kahn and Cerf created a new globally unique address, the **Internet Protocol (IP) address**. This address was 32 bits long. Routers work with 32-bit IP addresses directly. However, it is difficult for humans to remember strings of thirty-two 1s and 0s. So for inferior biological entities, IP addresses are usually expressed in **dotted decimal notation**. In this notation, the IP address is broken into four 8-bit segments. Each segment's bits are treated as a binary number and are converted into a decimal number. The four segment decimal numbers are written out with dots (periods) between them. So 128.171.17.13 is a typical IP address in dotted decimal notation.

*Computers have no problems dealing with 32-bit strings, so only inferior biological entities use dotted decimal notation as a memory aid.*

In a later innovation, a system was given for dividing up the billions of possible IP addresses in a way that would make each IP address globally unique. For example, the University of Hawai`i was given control over all IP addresses beginning with 128.171. This is 16 bits. No other organization could use IP addresses beginning with these bits. The University of Hawai`i then assigned the remaining 16 bits internally in a way that preserved uniqueness. For example, the university assigns one host the IP address 128.171.17.13.

If you think about it, telephone numbers are assigned this way. Each country gets to assign its internal telephone numbers any way it chooses, as long as each phone has a phone number and no two phones have the same number. Countries give non-overlapping blocks of numbers to individual telephone carriers and let the carriers assign them.

**IPV4 AND IPV6** In talking about 32-bit Internet Protocol (IP) addresses, we have been describing the address length in the current dominant version of the Internet Protocol, **IP Version 4 (IPv4)**. (There were no Versions 1, 2, and 3.) As we will see in Chapter 2, and later in this book, **IP Version 6 (IPv6)** is beginning to grow in popularity. IPv6 has 128-bit addresses instead of 32-bit addresses. We will see in Chapter 8 that IPv6 uses a much more complex scheme than dotted decimal notation to represent binary addresses for human reading.

**A SECOND LEVEL OF PACKETS** Adding a second layer of networking creates of several things besides addresses. We now had two levels of packets: one at the data link layer for single networks and one at the internet layer for internets. In time, terminology emerged to handle these differences. Packets at the data link layer are called **frames**, and packets at the internet layer are called **packets**.

*Packets at the data link layer are called frames, and packets at the internet layer are called packets.*

**THE RELATIONSHIP BETWEEN PACKETS AND FRAMES** Figure 1-19 shows the relationship between packets and frames. Here, there are three networks connected by routers. The packet sent by the source host travels all the way to the destination host. The packet is addressed to the IP address of the destination host. Within each network, however, the packet travels in a frame specific to that network. In other words, packets always travel inside frames. In this case, there is one packet, but there are three frames along the way, one for each network. If 10 networks separated the source host from the destination host, there would have been one packet traveling in 10 different frames.

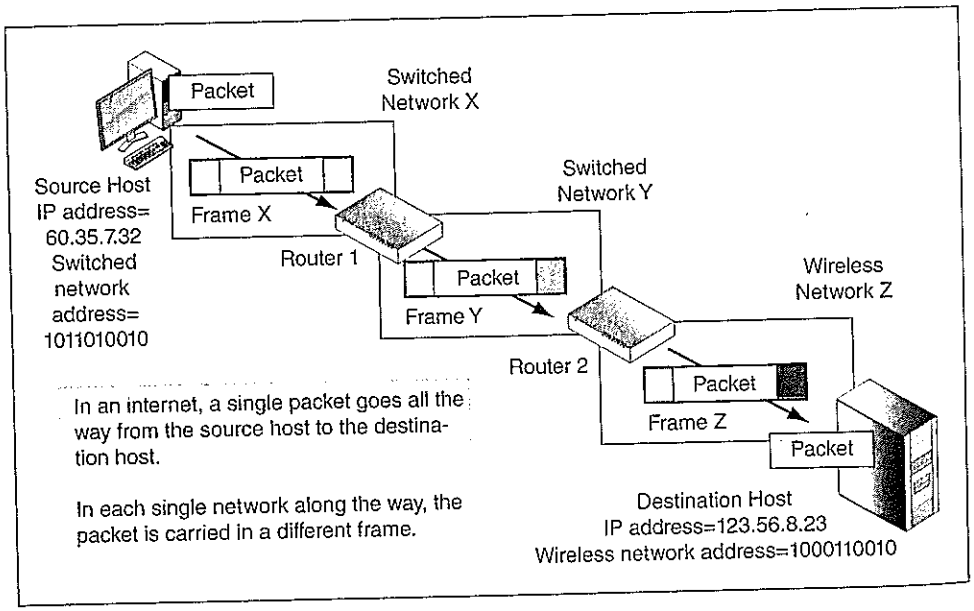


**FIGURE 1-19** Packets and Frames

**SWITCHES AND ROUTERS** Packet switches are needed at both the network and internet levels. Figure 1-20 shows that at the network level, packet switches are simply called *switches*. It also shows that at the internet layer, packet switches are called *routers*.[6]

**PHYSICAL LINKS, DATA LINKS, AND ROUTES** We also need two types of network links above the physical layer. We saw that the path a packet (now called a frame) takes across its single network is called its data link. Figure 1-19 shows that the path that a

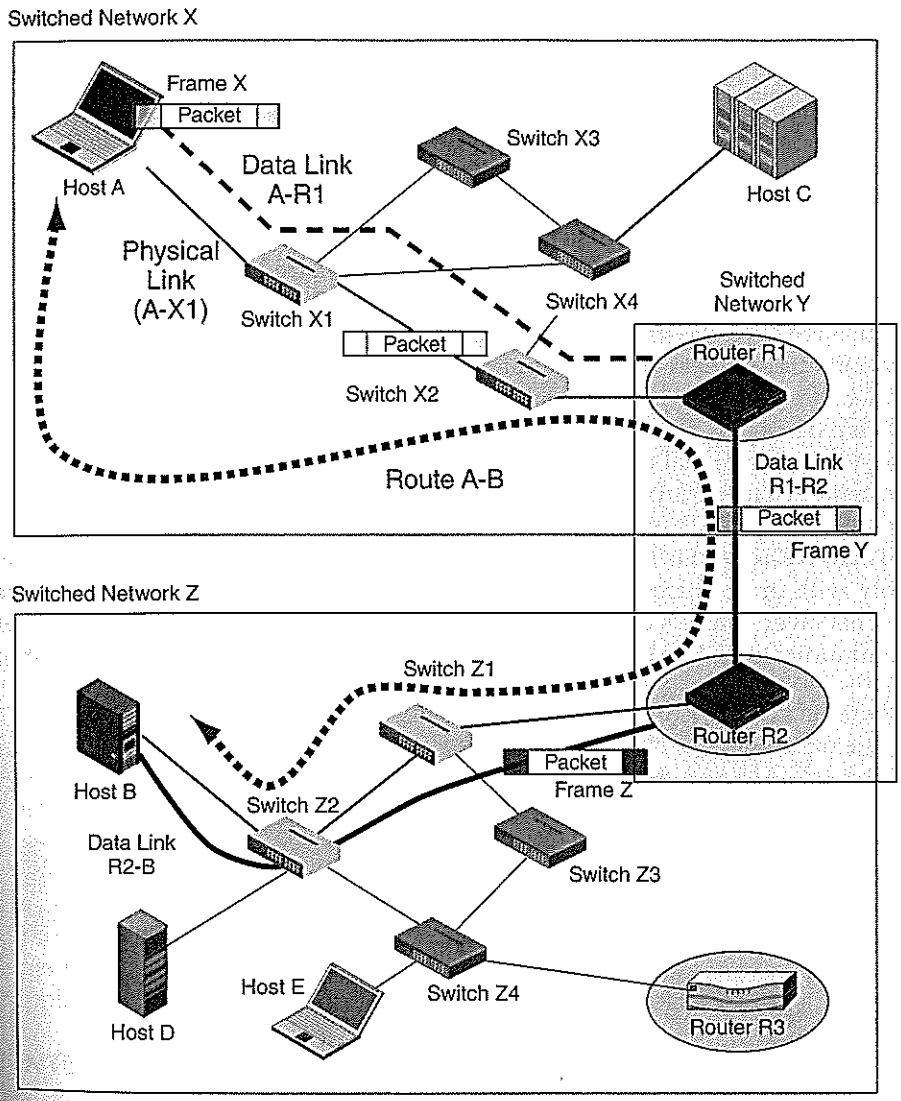


**FIGURE 1-20** Physical Links, Data Links, and Routes

[6] An odd side effect of this terminology is that you have packet routers and frame switches, but there are no packet switches per se. A really dirty question is, "At what layer do you find packet switches?"

*packet* takes at the internet layer across an entire internet is called its **route**. If there are 10 networks between the source host and the destination host, there will be 10 data links along the way but only one route.

---

*The path that a packet takes at the internet layer across the entire internet is called its route.*

---

**THE TRANSPORT LAYER** Actually, internetworking also required the creation of a fourth level of standards, the **transport layer**. Figure 1-21 shows the relationship between the internet layer and the transport layer.

The internet layer is concerned with moving packets across an internet, across a series of routers. It governs what happens on each router along the way. It also governs packet formats and addressing.

The **transport layer**, which is above the internet layer, is only concerned with what happens on the source host and destination hosts. We saw earlier that original application messages are fragmented and that the fragments are placed in packets. This is actually done at the transport layer, and the application layer fragments are called segments. On the destination host, the transport layer collects all of the arriving segments, places them in order, and passes them to the application program. The transport layer has a number of other functions. For example, it typically provides error correction, which means that packets that are damaged or lost in transmission are retransmitted. Overall, the internet layer is a best-effort service that tries to get packets through but may fail to do so. The transport layer usually is a fix-up layer that supplies the functionality that the internet layer lacks.

**THE APPLICATION LAYER** The four lowest layers get a packet to the destination host, possibly with error correction and other services. The fifth layer is the **application layer**. This layer controls communication between the two application programs that are communicating. For example, when browsers talk to webservers, this requires

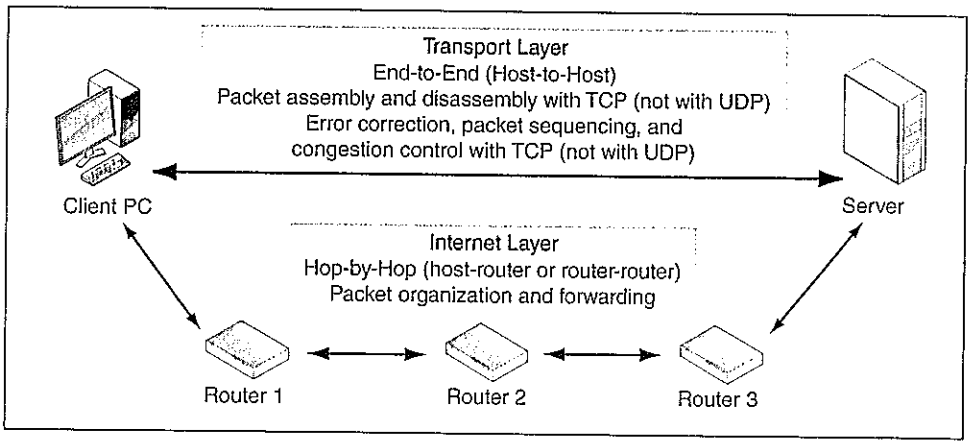


**FIGURE 1-21** Internet and Transport Layers

| Layer Number | Layer Name | Specific Purpose | Broad Purpose |
|---|---|---|---|
| 5 | Application | Communication between application programs | Application Communication |
| 4 | Transport | Application message fragmentation and reassembly. Ordering of packets, error correction, and congestion reduction | Internet Transmission |
| 3 | Internet | Connection across an internet. Defines packet formats and router operation | |
| 2 | Data Link | Connection across a single network. Defines frame formats and switch operation | Single-Network Transmission |
| 1 | Physical | Physical connections between adjacent devices | |

**FIGURE 1-22** Five Networking Layers

an application layer standard (HTTP) to specify the communication. There are many applications, and many of them have their own standards. Consequently, there are more application layer standards than there are standards at other layers.

**FIVE LAYERS** Overall, network functionality is described fairly well by thinking of the five layers we have seen in this chapter. Figure 1-22 shows that these are the physical layer, data link layer, internet layer, transport layer, and application layer. Each layer provides services to the layer above it. The bottom two layers provide transmission through single networks. The internet and transport layers provide host-to-host transmission through an internet. Finally, the application layer provides application–application communication.

### Test Your Understanding

16. a) Distinguish between *internet* with a lowercase i and *Internet* with an uppercase I. b) Why are many networking concepts duplicated in switched networks and internets? c) What are the two levels of addresses? d) How long are IP addresses? e) How are IP addresses usually expressed for humans? f) Distinguish between packets and frames. g) A host transmits a packet that travels through 47 networks. How many packets will be there along the route? h) How many frames will be there along the route? i) Are frames carried inside packets? j) Distinguish between switches and routers. k) Distinguish between physical links, data links, and routes. l) Distinguish between what happens at the internet and transport layers. m) Do IPv4, IPv6, or both use dotted decimal notation for human reading? n) Why are application layer standards needed? o) List the numbers and names of the five layers.

## THE INTERNET EVOLVES

### The TCP/IP Standards

Kahn and Cerf realized that their new approach would need new standards. Today, these are called the **TCP/IP standards**. As noted earlier, these standards are maintained today by the Internet Engineering Task Force (IETF). Figure 1-23 lists the main TCP/IP standards.

| Layer | Standard(s) | |
|---|---|---|
| Transport Layer | Transmission Control Protocol (TCP)<br>Fragmentation<br>Error Correction<br>Congestion Control | User Datagram Protocol (UDP)<br>No Fragmentation<br>No Error Correction<br>No Congestion Control |
| Internet Layer | Internet Protocol (IP)<br>IPv4 and IPv6 | |

**FIGURE 1-23** Core TCP/IP Standards

**IPV4 AND IPV6** For the internet layer, the **Internet Protocol (IP)** was created. This layer deals with addresses, so addresses on an internet are called **IP addresses**. As we saw earlier, the dominant version of IP today is Version 4 (IPv4). The emerging version is IPv6.

**TRANSMISSION CONTROL PROTOCOL (TCP)** At the transport layer, TCP/IP has two alternatives. For a transport layer protocol with *high functionality*, the transport standard was the **Transmission Control Protocol (TCP)**. TCP fragments application messages. Each fragment is carried in a TCP message, which is called a **TCP segment**. TCP also re-orders information in packets if the packets arrive out of order, corrects errors, and reduces the likelihood of congestion.

**USER DATAGRAM PROTOCOL (UDP)** The second transport layer alternative is for applications that do not need or cannot use high transport layer functionality. This is the **User Datagram Protocol (UDP)**. UDP messages are called **datagrams**. UDP does *not* do fragmentation, so the application message must be able to fit inside a single UDP datagram. In addition, UDP does no error correction. If there are any errors, the application program must handle them.

This may not make much sense to you. Why have a layer standard that essentially does nothing? The answer is that while it does nothing, it does so very cheaply. In Chapter 2, we will see that UDP is much simpler than TCP and places a much smaller burden on the network. For applications that do not need the high functionality of TCP, UDP offers lower network costs.

**THE TCP/IP STANDARDS** The original standards specified three main protocols: IP, TCP, and UDP. However, they became known collectively as the **TCP/IP standards**. Later, many more standards were created to add functionality missing from TCP, UDP, and IP. However, this growing family of standards usually is still called the TCP/IP standards family. It is the dominant standards family for the Internet.

*TCP/IP is a family of standards including IP, TCP, UDP, and many other standards.*

### Test Your Understanding

17. a) What are the roles of the Internet Protocol? b) What are the roles of the Transmission Control Protocol? c) What are the limitations of the User Datagram Protocol? d) Why is UDP used sometimes? e) What is TCP/IP?

## The Internet Is Born—Slowly

When did the ARPANET end and the Internet start? (The equivalent question is when did ARPANET standards end and the TCP/IP standards begin? The TCP/IP standards are the defining feature of the Internet.) The answer is surprisingly difficult because the Internet did not appear all at once. For several years after the TCP/IP standards were created, hosts were allowed to transition gradually from NCP to TCP/IP. During this period, both NCP and TCP/IP standards were supported. Finally, on January 1, 1983, hosts were required to stop using NCP. Although there were a few loud protests,[7] most hosts had long since transitioned TCP/IP. In that sense, the Internet was born on January 1, 1983. At the same time, it had effectively existed for several years.

### Test Your Understanding

18. a) In what sense is January 1, 1983, the birthday of the Internet? b) In what sense is it not?

## The Internet Goes Commercial

Initially, ARPA funded ARPANET and Internet transmission. In 1986, the National Science Foundation created NSFNET, which two years later became the backbone of the Internet. NSFNET brought higher speeds to the Internet backbone. NSFNET also brought the **Acceptable Use Policy** to the Internet. Basically, the Acceptable Use Policy explicitly forbade the use of the Internet for commercial purposes such as buying, selling, and advertising. The Internet was to be a pure research network, although the e-mail systems of commercial networks were eventually allowed to use the Internet.

On April 30, 1995, the NSF discontinued the NSFNET as the core of the Internet. Before this, a number of companies called Internet Service Providers (ISPs) had formed to connect users to the Internet. These ISPs were also connected to one another. Consequently, when NSF pulled the plug on NSFNET support for the Internet, the Internet continued with no visible change to users.

Figure 1-24 shows today's commercial Internet. Technologically, the Internet is simply a large collection of routers. However, these routers are owned by different ISPs. The ISPs are interconnected at **Network Access Points (NAPs)**, which allow the ISPs to exchange packets.

To use the Internet today, you *must* connect to it via an ISP. The ISP sends your packets into the Internet, to destination hosts. Your ISP also delivers reply packets to you. You need an access line from your home or place of business to your ISP's nearest office. Corporations, which have far larger volumes of Internet traffic than individuals, need much faster access lines.

Who pays for Internet transmission? The answer is that *you* do. You pay money each month to your ISP. Corporations also pay money to their ISPs. While you pay ten to a hundred dollars a month, large corporations pay tens of thousands of dollars per month for service. Given the number of people and corporations on the Internet, there is enough money to pay for all of the Internet's transmission volume.

[7] To nudge organizations along, NCP functionality was made nonfunctional for a brief time and then for a slightly longer time as the deadline approached. This was not publicized at the time, but it was effective.

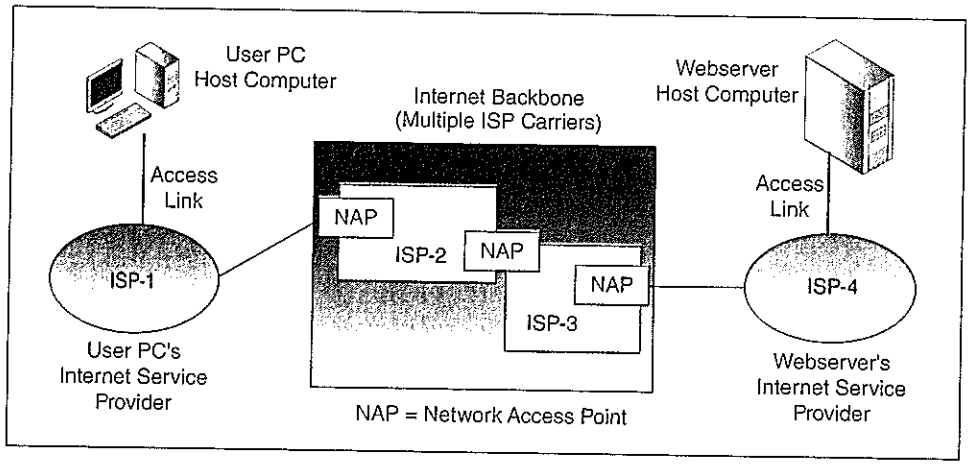


**FIGURE 1-24** The Commercial Internet

Although Internet service did not change when the Internet backbone was transferred to commercial ISPs, the vanishing of NSF support led to the vanishing of NSF's acceptable use policy. From that point on, commercial activities were perfectly fine.

The termination of the NSFNET did not come as a surprise to anyone. NSF announced the change far ahead of time, and companies were ready to offer commercial service. Immediately after the NSFNET was disconnected, companies began to offer e-commerce service on a massive scale. In a few months, it was hard to believe that e-commerce was not several years old. Many Internet-only companies began to sell stock through initial public offerings. A vast land rush of dot-com companies appeared, many based on absurd business models. In 2000 and 2001, the bottom dropped out of dot-com stocks. However, e-commerce did not collapse. After one year of stagnant growth in the middle of a recession, e-commerce continued to grow very rapidly.

Most e-commerce activity relied upon the World Wide Web, which was invented at CERN by Tim Berners-Lee. Although he created the HTML and HTTP standards in 1991, WWW protocols were just beginning to reach widespread use on the Internet when commercial activity became possible.

### Test Your Understanding

19. a) What was the Acceptable Use Policy in place on the Internet before 1995? b) Why did commercial activities on the Internet become acceptable in 1995? c) What do we call the carriers that provide Internet service? d) Why do they need to be interconnected? e) At what locations do ISPs interconnect?

## Supervisory Applications

TCP, IP, and UDP are sufficient for delivering packets over an internet, but, as noted earlier, the TCP/IP family of standards today is much larger than these three. Many TCP/IP standards are user application standards, such as standards for e-mail and the Web. Many other TCP/IP standards, however, are **supervisory standards** that keep an

internet working. We will look at two TCP/IP supervisory application standards in this section. We will see many more throughout this book.

**THE DYNAMIC HOST CONFIGURATION PROTOCOL (DHCP)** Servers must have permanent IP addresses. Otherwise, clients could not find them. (Imagine what it would be like if your favorite store kept changing its street address.) Unchanging addresses are called **static IP addresses.**

Client PCs normally get their IP addresses a different way. When a client PC boots up, it realizes that it has no IP address. Figure 1-25 shows that the PC sends[8] a **Dynamic Host Configuration Protocol (DHCP)** request message. This message asks for an IP address.

When the DHCP server receives the message, the server picks an available IP address from its address database. It then sends this IP address in a DHCP response message. The client PC uses that IP address as its IP address. This is called a **dynamic IP address.**

The figure indicates that DHCP does more than give the client PC an IP address. As "Configuration" in the name suggests, it sends general configuration information.

- This includes the IP address of a default router. When the PC needs to send a packet to a host that is not on the same network, it sends the packet to this default router.
- Configuration information also includes the IP addresses of local Domain Name System (DNS) servers, which we will see next.
- It includes other information such as a subnet address mask, which we will see later in this book.

It would be possible to simply enter this information in every corporate PC just once, manually. Afterward, the client PC would not have to use DHCP. However, consider what would happen if the firm later changed the IP addresses of its Domain

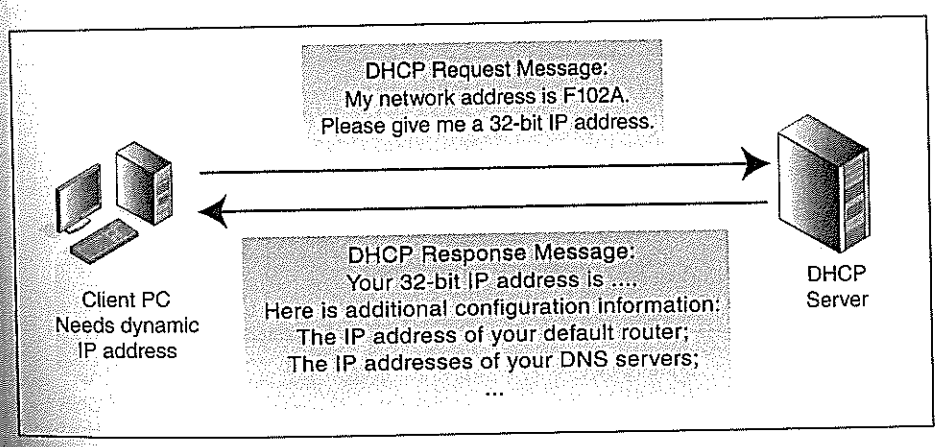


**FIGURE 1-25** Dynamic Host Configuration Protocol (DHCP)

[8] Actually, it broadcasts the message because it does not know the IP address of the DHCP server. If more than one DHCP server replies, the PC picks one of them.

Name System servers. The firm would have to reconfigure every client PC manually. That would be painfully expensive. With DHCP, every client PC automatically gets hot fresh information every time it boots up. If configuration information changes, all client PCs will be updated automatically.

**THE DOMAIN NAME SYSTEM (DNS)** To send a packet to another host, a source host obviously needs to know the IP address of the destination host. However, people are not good at memorizing 32-bit IP addresses, even in dotted decimal notation. Consequently, servers are often given host names, such as Voyager.shidler.hawaii.edu. Host names are much easier to memorize than IP addresses, so when we type URLs, we use host names instead of IP addresses. (In fact, you probably didn't even know that you could use IP addresses instead of host names in URLs.)

For the ARPANET, there was a Network Information Center (NIC) at Stanford Research Institute that maintained a file containing the host names and host ARPANET address of every named host on the ARPANET. The NIC was run by a single person, Elizabeth (Jake) Feinler. When the ARPANET grew, she eventually got an assistant.

As the ARPANET and then the Internet grew, this centralized manual approach to maintaining host names and associated addresses became impossible to continue. Consequently, the IETF created the **Domain Name System (DNS)** in 1984. In this system, each organization with a second-level domain name, such as Panko.com or Hawaii.edu, must maintain one or more DNS servers. ISPs that provide service directly to customers also need to maintain DNS servers for their customers.

Figure 1-26 shows that when a source host needs to know the IP address of a destination host, the source host sends a DNS request message to its local DNS server. The DNS request message contains the host name of the target host.

As the figure shows, the DNS server looks up the IP address associated with the host name in its DNS table. It then sends back a DNS response message that contains the IP address.

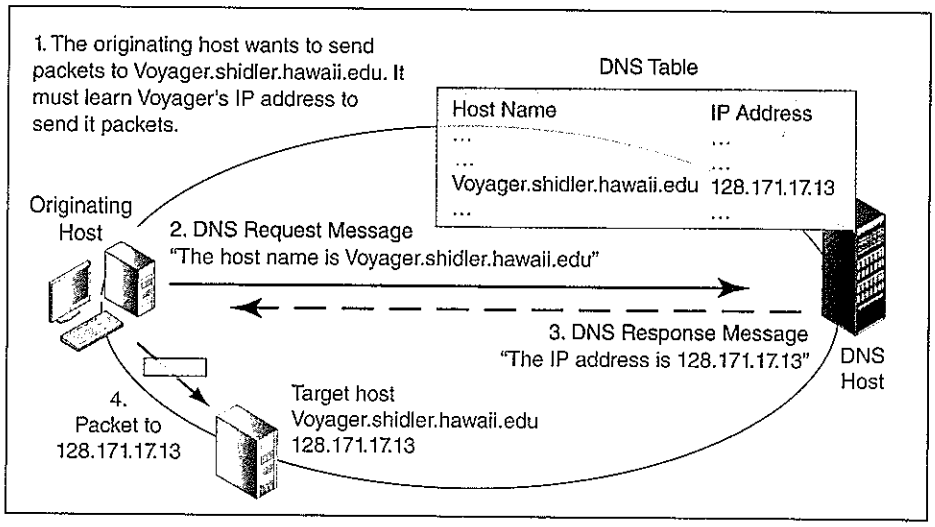


**FIGURE 1-26** Domain Name System (DNS) Service

Now the source host knows the IP address of the target host. It has no more need for the DNS server. The source host simply sends packets to the IP address of the target host.

What happens if the local DNS server does not know the IP address of the host name? In that case, the local DNS server will contact other DNS servers until it finds the correct IP address (or until it gives up).

### Test Your Understanding

20. a) Why do servers need static IP addresses? b) What protocol provides a client PC with its dynamic IP address? c) What other configuration information does this protocol provide? d) Why should PCs get their configuration information dynamically instead of manually?
21. a) To send packets to a target host, what must the source host know? b) If the source host knows the host name of the target host, how can it get the target host's IP address?
22. Both DHCP servers and DNS servers send a host an IP address. These are the IP addresses of what hosts?

## A SMALL HOME NETWORK

We have looked at networking *principles* so far. This box looks at a real, although very small network—a network in a residential home. This is a network on the family's premises, so by definition, it is a local area network. Although this is a small network, it has most of the elements you have studied in this chapter.

### Components

Figure 1-27 illustrates the basic hardware devices in a typical home computer network.

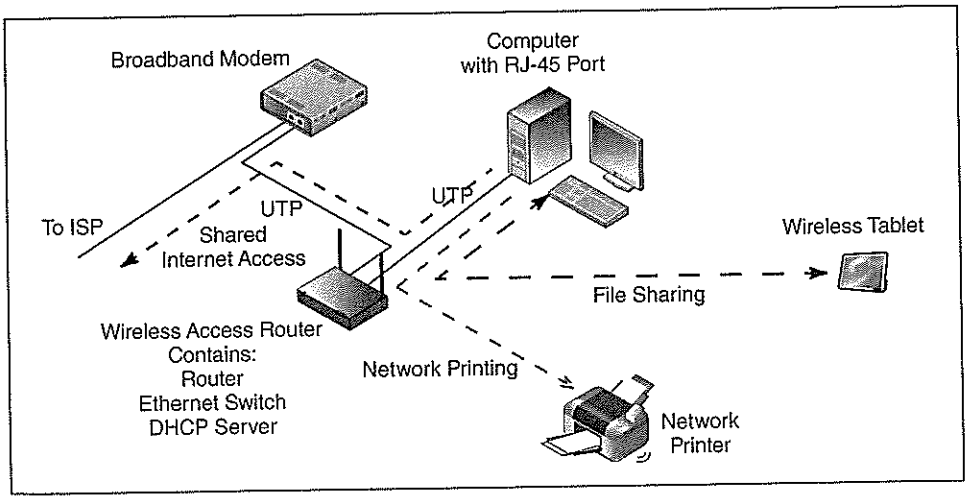


**FIGURE 1-27** A Small Home Network

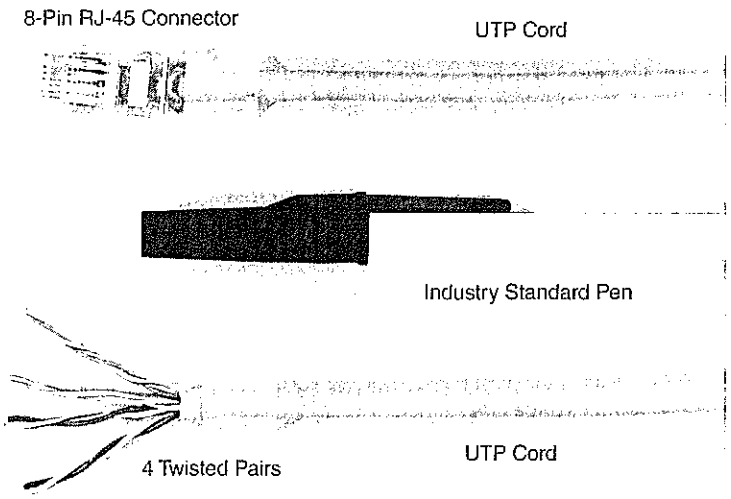


**FIGURE 1-28** Unshielded Twisted Pair (UTP) Wiring

*Source:* Courtesy of Raymond R. Panko

- The heart of the network is a wireless access router. We will look at this device in more detail a little later.
- A broadband modem connects the home network to an Internet Service Provider via a wired connection or a wireless connection. We will look at wired ISP connections in Chapter 10.
- There are two client hosts. One is connected to the wireless access router using a 4-pair unshielded twisted pair (UTP) cable. Figure 1-28 shows that 4-pair UTP consists of eight copper wires arranged in pairs, with each pair twisted around each other several times per inch. 4-pair UTP looks like a fat home telephone wire. It terminates in an RJ-45 connector. We will see more about 4-pair UTP in Chapter 5.
- The other client is a tablet with wireless capability. It connects to the wireless access point via radio signals. With wireless connections, there is no need to buy UTP cables and run them to each computer. However, as we will see in Chapter 7, wireless transmission is not always reliable or as fast as UTP transmission. We will also see in Chapter 7 that the main standard for wireless LAN (WLAN) transmission is 802.11.
- The final element is a wireless **network printer**. This printer also communicates with the wireless access router via 802.11. An increasing number of printers are network printers, which communicate with the access router via UTP, 802.11, or both.

### The Wireless Access Router

The **wireless access router** deserves special attention because it contains several important hardware functions.

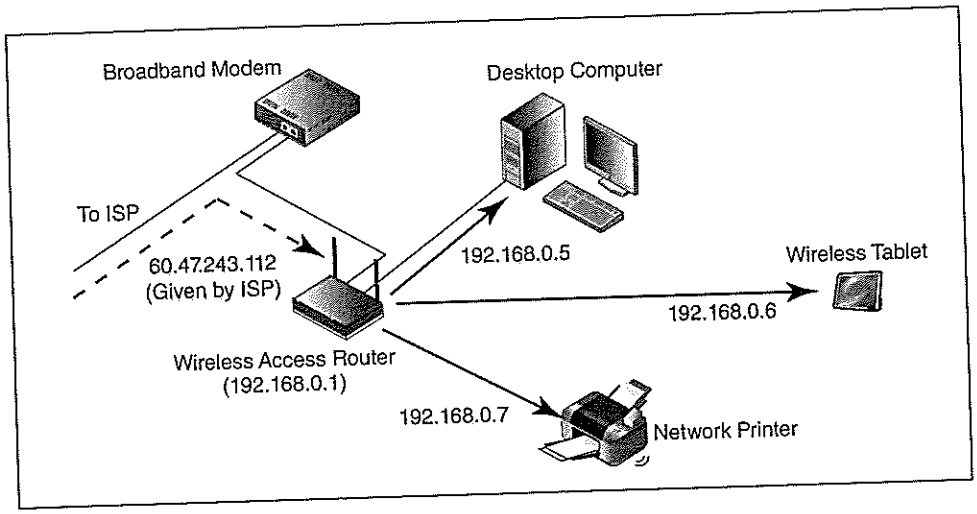


**FIGURE 1-29** DHCP in a Small Home Network

- First, although it is more than just a router, the access router really is a router. Routers connect two networks. The access router connects the home network to the network of the ISP that provides Internet access.
- Second, the router has a built-in Ethernet switch. Most home access routers have at least four RJ-45 ports for UTP connections. The router in the figure has another RJ-45 port to connect to the broadband modem.
- The router has a wireless access point to connect by radio to wireless hosts within the house. Not all routers have built-in wireless access points.
- In this chapter, we saw that client PCs get their IP addresses from DHCP servers. Somewhat amazingly, the wireless access router has a built-in DHCP server. Figure 1-29 shows that the DHCP server gives IP addresses to the two clients (192.168.0.5 and 192.168.0.6) and to the network printer (192.168.0.7). The wireless access router's DHCP server also gives the wireless access router its own IP address (192.168.0.1).
- The wireless access router also provides **network address translation (NAT)**, which translates between the internal IP addresses and the single IP address the ISP gives to the household (60.47.243.112). As Figure 1-30 illustrates, the ISP's DHCP server only gives the household a single IP address. When an internal device transmits, NAT converts the IP source addresses in its packets to the ISP's single allocated IP address. It then sends the packet on to the ISP. When packets arrive from the ISP, all have the IP address provided by the ISP as their destination addresses. The NAT function in the wireless access router places the internal IP address of the internal PC or the network printer into the packet's destination IP address field.

## Services

Once the network is set up, the users can focus on the services their home network provides. Three of these services dominate today:

- **Shared Internet access** allows the two client PCs to use the Internet simultaneously, as if each was plugged directly into the broadband modem.

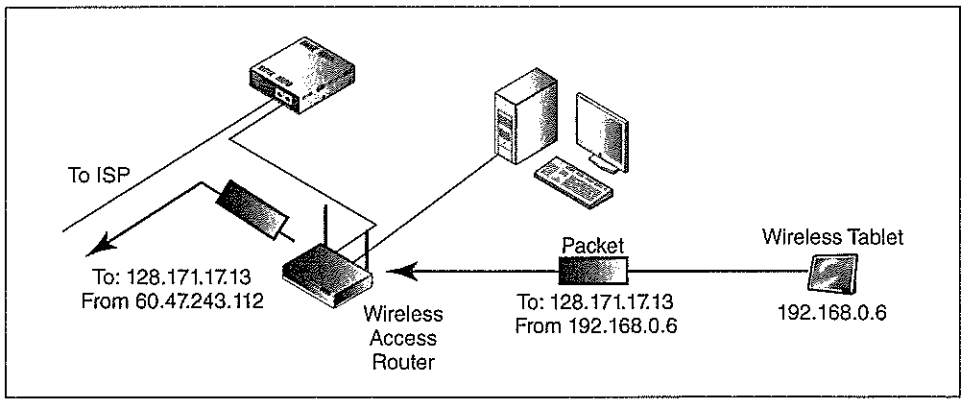


**FIGURE 1-30** Network Address Translation (NAT)

- **File sharing** allows the two client PCs to share files on each other's hard drives. For example, one PC may have the family budget. A user at the other PC can access the budget file at any time.[9]
- **Printer sharing** allows either client PC to print to the printer. Both can even print at the same time. If they do, the printer will store one job in the print queue until it finishes printing the other.

## Configuration

Shared Internet access is completely automatic and requires no setup. However, file sharing and printer sharing require some setup on each of the client PCs. The process varies between operating systems and between versions of each operating system (e.g., Vista versus Windows 7).

The wireless printer and wireless access point also have to be configured. They do not have displays and keyboards to allow configuration. Consequently, they are configured from one of the PCs. They are first connected to the network. The PC doing the configuration finds them and does the configuration work.

### Test Your Understanding

23. a) List the hardware elements in the small home network described in this section. b) For wired connections, what transmission medium is used? c) What is its connector standard? d) What is the standard for wireless PCs and printers to connect to a wireless access point? e) What are the five hardware functions in a wireless access router? f) Why is the DHCP function necessary? g) Why is NAT necessary? h) What three services does this network provide to the desktop PC and the wireless tablet? i) Which devices need to be configured? (List them.)

[9]Network hard drives can be attached to the network like a network printer. Networked PCs can access them directly.

## HOW THIS BOOK IS ORGANIZED

Figure 1-31 shows how the rest of the book is organized. Chapter 2 looks at network standards in much more depth than we have seen in this chapter. Network standards are critical in networking. They allow products from different vendors to interoperate (work together) regardless of who produced them. Standards create vendor competition with all of the benefits that competition brings in terms of price reductions and the drive to increase features to avoid producing commodity products.

The next two chapters bring us into security. We have not talked about security in this chapter. In fact, in the first two chapters, we will discuss networking as if security was not a problem. Of course, security is a very real and important problem. You can rarely pick up a newspaper or scan the news online today without seeing something about the latest security incident. However, it seems easiest for you to focus on how networks operate in general before looking at how to set up defenses against attackers. To understand network security, you must first understand networks. Just as the networking concepts you will see in the first two chapters will reappear constantly throughout the book, we will look at security concerns in almost everything we cover in the rest of the book.

In this chapter, we introduced the five layers in network standards: the physical, data link, internet layers, transport, and application layers. Beginning in Chapter 5, we will

| Chapter | Topics |
|---|---|
| 1 | Welcome to the Cloud |
| 1a | Hands-On: Windows Networking |
| 2 | Network Standards |
| 2a | Hands-On: Wireshark Packet Capture |
| 3 | Network Security |
| 4 | Network Management |
| 4a | Hands-On: Microsoft Office Visio |
| 5 | Wired Ethernet LANs |
| 5a | Hands-On: Cutting and Connectorizing UTP |
| 5b | Hands-On: Ethernet Switching |
| 6 | Wireless LANs I |
| 6a | Hands-On: Using the Xirrus Wi-Fi Inspector |
| 7 | Wireless LANs II |
| 8 | TCP/IP Internetworking I |
| 9 | TCP/IP Internetworking II |
| 10 | Wide Area Networks |
| 11 | Networked Applications |

**FIGURE 1-31** Organization of the Book

move up through these five layers, beginning with the lowest layers and moving up to applications in the last chapter. There are three chapters on local area networks (LANs)—one on wired Ethernet LANs and two on the complex world of wireless LANs (WLANs). Chapters 8 and 9 deal with the internet and transport layers and with the TCP/IP standards that dominate at these layers. Chapter 10 discusses the messy but vital world of wide area networks. Finally, Chapter 11 covers how applications work over networks.

The book has several "a" and "b" chapters that provide hands-on exercises in networking. Students tend to like hands-on exercises, but keep in mind that networking is primarily a conceptual game. Anything you learn may be critical in any networking task. In networking, it is the things you don't know that kill you.

## CONCLUSION

### Synopsis

We began this chapter (and this book) with a look at what user experiences probably will be like in the future. Networking drives all of these innovations, but the network is invisible to the users in this scenario. This is the way it should be. Networking professionals, of course, must understand how networks operate in great detail. Other IT professionals also have to understand networking. For example, programmers increasingly write programs that talk to other programs over networks.

Afterward, we looked at basic network terminology from the user's viewpoint. We defined a network as a system that permits applications running on different hosts to work together. Networked applications, quite simply, are applications that require a network to work (e.g., e-mail). A host, in turn, is any device attached to a network. A host can be a client PC, a server, a laptop, a netbook, a mobile smartphone, a tablet, or any other networked device. The network core is the central part of the network. Access lines connect hosts to the network core. We showed the core of the network as a cloud to indicate that users do not have to "look inside the cloud" to know how it operates.

We then looked at three ways applications can work with each other over networks. In client/server computing, a client program on a client computer receives services from a server program on a server computer. Client/server computing works through a request–response cycle in which the client sends a request to the server and the server sends back a response. Peer-to-peer (P2P) computing, in which client hosts provide service to each other, is growing rapidly. Cloud computing, in which servers, applications, or both are invisible in a cloud, is also beginning to grow.

We noted that speeds are measured in *bits* per second (bps), not *bytes* per second (Bps). Speeds are expressed in the metric system, with kilobits per second (kbps), megabits per second (Mbps), gigabits per second (Gbps), and terabits per second (Tbps). Metric measurements increase in factors of 1,000, not 1,024.

We pay for telephone lines by the minute. This is fine for voice because not too much transmission capacity is needed and because human conversations use telephone lines continuously, wasting little capacity. In contrast, data transmission usually consists of enormous traffic bursts separated by long silences. To pay by the minute for a great deal of capacity and barely use it would not make sense economically.

Packet switching addresses this inefficiency by dividing application messages into small segments and sending each segment in an individual packet. The packets of multiple conversations can share leased line circuits. This is called multiplexing. It uses

capacity efficiently. Packet switching also makes error correction easier because only a single packet has to be retransmitted, not the entire original application message.

When a packet arrives at a packet switch, the switch may have several alternatives for forwarding the packet to another packet switch. The switch makes a switching decision and sends the packet on to the next packet switch. That switch makes its own switching decision to another packet switch. These sequential decisions eventually get the packet all the way to the destination host. Links between hosts and packet switches and between pairs of packet switches are called physical links. The path a packet takes across a packet-switched network is called a data link.

The ARPANET was the first major packet-switched network. The first ARPANET standards were created by an informal Network Working Group. This group of graduate students had no formal permission to create standards, so they called their standards requests for comments (RFCs). One of the first applications on the ARPANET was e-mail, but there were soon many more. Later, the job of creating standards for the successor of the ARPANET—the Internet—was taken over by the Internet Engineering Task Force (IETF).

As more networks began to appear, there was a need to interconnect them. Bob Kahn and Vint Cerf came up with an approach for doing this. Their idea was to create internets by connecting individual networks together with routers. They essentially created a second layer of networking. Consequently, internet transmission involves two types of addresses, two types of packet switches (switches and routers), two types of packets (frames and packets), and two types of paths—data links across individual switched networks and routes all the way from the source to the destination address. This is very confusing, but you need to spend the time to understand it clearly or you will have problems throughout the book.

A new set of standards was created for internetworking. These are generically called the TCP/IP standards. The basic work of internetworking was divided into two layers. The internet layer governs packet organization and how packet switches forward packets to their destination. The main standard at this layer is the Internet Protocol (IP). IPv4 addresses are 32 bits long and are often represented as dotted decimal notation, in which there are four numbers separated by dots. An example would be 128.171.17.13. In Chapter 2, we will see that IPv6 addresses are more complex.

The transport layer lies above the internet layer. The TCP standard at the transport layer provides application message fragmentation and reassembly, placing packets in their correct sequence, handling error correction, and bringing congestion avoidance, among other things. The UDP standard at the transport layer does none of these things. UDP is good when an application cannot use the services of TCP.

In addition to these three core protocols, TCP/IP has many application protocols. It also has many supervisory protocols to keep an internet operating (including the global Internet). We briefly looked at two supervisory protocols. DHCP provides your client with an IP address every time you connect to the Internet, as well as other configuration information. DNS allows your computer to learn another computer's IP address if you only know its host name.

The global Internet grew out of the original ARPANET. In 1995, the National Science Foundation stopped paying for the operation of the Internet backbone. Its role was taken over by commercial Internet Service Providers (ISPs), which interconnect at Network Access Points (NAPs). Once government money was no longer used, the Internet could be used for commercial activities. E-commerce was born immediately and grew meteorically.

The chapter closed with a box that looked at a small PC network in a home. These networks are familiar to most students. Although they are small, they encompass most major concepts in networking.

## END-OF-CHAPTER QUESTIONS

### Thought Questions

1. a) In Figure 1-15, when Host X transmits a packet along the data link shown, how many physical links are there along the data link shown? b) How many data links?
2. a) In Figure 1-20, how many physical links, data links, and routes are there along the way when Host A sends a packet to Host B? b) When Host E sends to Host C? (Assume that hops will be minimized across switches and routers.) c) When Host D sends to Host E? (Assume that hops will be minimized across switches and routers.)
3. In a certain network, there are nine routers between Host R and Host S. a) How many data links will there be along the way when Host R transmits a packet to Host S? (Hint: Draw a picture.) b) How many routes? c) How many frames?
4. Why does it make sense to make only the transport layer reliable? This is not a simple question.
5. a) What does it mean that data transmission is bursty? b) Why is burstiness bad if you pay for a transmission line by the minute?
6. What layer fragments application messages so that each fragment can fit inside an individual packet?

### Case Study

1. A friend of yours wishes to open a small business. She will sell microwave slow cookers. She wishes to operate out of her house in a nice residential area. She is thinking of using a wireless LAN to connect her four PCs. What problems is she likely to run into? Explain each as well as you can. Your explanation should be directed to her, not to your teacher. This is not a trivial problem.

### Perspective Questions

1. What was the most surprising thing for you in this chapter?
2. What was the most difficult thing for you in this chapter?

# 1a HANDS-ON: WINDOWS NETWORKING

## LEARNING OBJECTIVES

**By the end of this chapter, you should be able to:**

- Use basic networking commands in Microsoft Windows.
- Discuss network concepts in Chapter 1 with better understanding.

## HANDS-ON NETWORKING TOOLS

This chapter introduces you to basic hands-on networking tools in Microsoft Windows. It assumes that you have read Chapter 1.

### Binary and Decimal Conversions Using the Microsoft Windows Calculator

It is relatively easy to convert 32-bit IP addresses into dotted decimal notation if you use the Microsoft Windows Calculator. As shown in Figure 1a-1, go to the *Start* button, then to *Programs* or *All Programs*, then to *Accessories*, and then click on *Calculator*. The Windows Calculator will pop up. Initially, it is a very simple calculator. Choose *View* and click on *Scientific* to make Calculator an advanced scientific calculator.

**BINARY TO DECIMAL** To convert eight binary bits to decimal, first divide the 32 bits into four 8-bit segments. Click on the *Bin* (binary) radio button and type in the 8-bit binary sequence you wish to convert. Then click on the *Dec* (decimal) radio button. The decimal value for that segment will appear.

Note that you cannot convert the whole 32-bit IP address at one time. You have to do it in four 8-bit segments.

Once you have the four decimal segment values, write them in order with dots between them. It will look something like 128.171.17.13. You have now converted the 32-bit IP address to dotted decimal notation.

**DECIMAL TO BINARY** To convert decimal to binary, go to *View* and choose *Scientific* if you have not already done so. Click on *Dec* to indicate that you are entering a decimal number. Type the number. Now click on *Bin* to convert this number to binary.

One subtlety is that Calculator drops initial zeros. So if you convert 17, you get 10001. You must add three initial zeros to make this an 8-bit segment: 00010001.

Another subtlety is that you can convert only one 8-bit segment at a time.

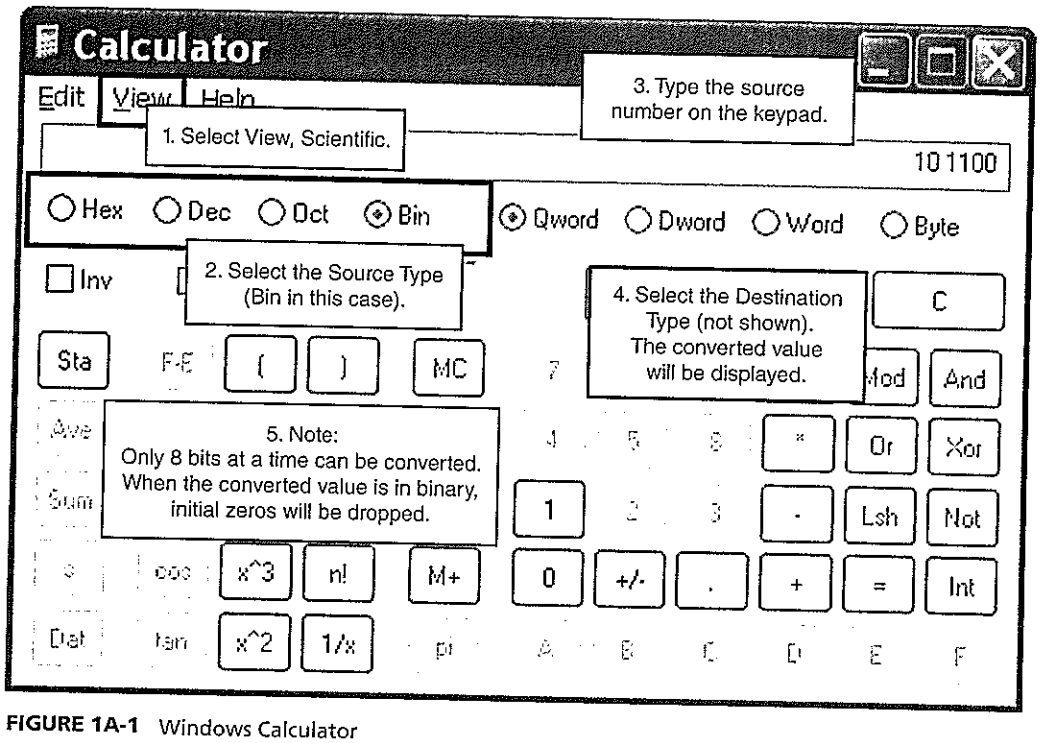


**FIGURE 1A-1** Windows Calculator

*Source:* Screenshot © 2012 by Microsoft Corporation. Used with permission from Microsoft.

1. a) What is 11001010 in decimal? b) Express the following IP address in binary: 128.171.17.13. (*Hint:* 128 is 10000000. Put spaces between each group of 8 bits.) c) Convert the following address in binary to dotted decimal notation: 11110000 10101010 00001111 11100011. (Spaces are added between bytes to make reading easier.) (*Hint:* 11110000 is 240 in decimal.)

### Test Your Download Speed

How fast is your Internet connection? Test your download speed at two sites. The following is a list of sites offering free download scans. If you can, test your bandwidth during periods of light and heavy use.

http://www.speakeasy.net/speedtest/

http://reviews.cnet.com/internet-speed-test/

http://www.speedtest.net/

2. a) What kind of connection do you have (telephone modem, cable modem, LAN, etc.)? b) What site did you use for your first test? c) What did you learn? d) What site did you use for your second test? e) Did you get different results?

### Working with the Windows Command Line

Windows offers a number of tools from its command line prompt. Network professionals need to learn to work with these commands.

**GETTING TO THE COMMAND LINE** To get to the command line, click on the *Start* button and choose *Run*. Type either *cmd* or *command*, depending on your version of Windows, and then hit *OK*.

**COMMAND LINE RULES** At the command line, you need to type carefully because even a single-letter error will ruin the command. You also need to hit *Enter* at the end of each line. You can clear the command line screen by typing *cls* and then pressing *Enter*.

3. Go to the command line. Clear the screen.

**SEE YOUR CONFIGURATION** In Windows, you can find information about your own computer by typing *ipconfig/all Enter* at the command line. This will give you your IP address, your physical address (your Ethernet address), and other information.

4. Use ipconfig/all or winipconfig. a) What is your computer's IP address? b) What is its Ethernet address? c) What is your default router (gateway)? d) What are the IP addresses of your DNS hosts? e) What is the IP address of your DHCP server? f) When you get a dynamic IP address, you are given a lease specifying how long you may use it. What is the starting time and ending time of your lease?

## Ping and Tracert

PING To find out whether you can reach a host and to see how much latency there is when you contact a host, use the **ping** command. You ping an IP address or host name much as a submarine pings a target to see whether it exists and how far away it is. To use the command, type *ping <hostname>* and press *Enter*, or type *ping <IPaddress>* and press *Enter*. Ping may not work if the host is behind a firewall, because firewalls typically block pings.

5. Ping a host whose name you know and that you use frequently. What is the latency? If this process does not work because the host is behind a firewall, try pinging other hosts until you succeed.

**PING 127.0.0.1 (PC, CALL HOME)** Ping the address 127.0.0.1. This is your computer's **loopback address**. In effect, the computer's network program sends a ping to itself. If your PC seems to have trouble communicating over the Internet, type *ping 127.0.0.1* and *Enter*. If the ping fails, you know that the problem is internal and you need to focus on your network software's configuration. If the ping succeeds, then your computer is talking to the outside world at least.

6. Ping 127.0.0.1. Did it succeed?

**TRACERT** The Windows **tracert** program is like a super ping. It lists not only latency to a target host, but also each router along the way and the latency to that router. Actually, tracert shows three latencies for each router because it tests each router three times. To use tracert, type the *tracert <hostname>* and press *Enter*, or type the tracert *IP address* and press *Enter*. Again, hosts (and routers) behind firewalls will not respond.

7. Do a tracert on a host whose name you know and that you use frequently. You can stop the tracert process by hitting Control-C. a) What is the latency to the

destination host? b) How many routers are there between you and the destination host? If this does not work because the host is behind a firewall, try reaching other hosts until you succeed.

8. Distinguish between the information that ping provides and the data that tracert provides.

### Nslookup

The **nslookup** command allows you to send a DNS request message to your local name server. At the command prompt, type *nslookup <hostname>* [*Enter*], where *<hostname>* is a host name for which you wish to know the IP address. The information shown after you type your command is the IP address.

Your local DNS host may send you a non-authoritative IP address. Each DNS server has a group of IP addresses and host names for which it is the authoritative DNS server. For instance, the DNS servers for Hawaii.edu are authoritative for all host names ending in Hawaii.edu. Sometimes, a DNS server will happen to know the IP addresses for host names in other domains. This is a non-authoritative IP address.

9. Find the IP address for Microsoft.com and Apple.com.

### RFCs

In networking, you frequently have to look up RFCs. Google, Bing, or other search tools let you do this.

10. a) Look up RFC 1149. b) In layperson's terms, what does this RFC specify? c) What are its sections? (This is a serious question. You should learn how RFCs are structured.) d) On what day was it created?