PART 3

TECHNOLOGIES FOR ELECTRONIC COMMERCE

CHAPTER 8

WEB SERVER HARDWARE AND SOFTWARE

LEARNING OBJECTIVES

In this chapter, you will learn about:

- Web server basics
- Software for Web servers
- E-mail management and spam control issues
- Internet and Web site utility programs
- Web server hardware

INTRODUCTION

As you learned in earlier chapters, **Lands' End** was one of the most successful clothing retailers on the Web before it was acquired by Sears in 2003. Now, as a division of Sears, Lands' End continues to be a leader in adding features that attract customers to its Web site and keep those customers coming back. Behind the scenes at Lands' End, a team of experienced technology professionals implements new Web page features and performs the many regular maintenance tasks that are necessary to keep the Lands' End Web site running smoothly.

Lands' End closely monitors the performance of its Web site to make sure that customers have a consistent experience each time they visit the site. The Web site's technical team works hard to make sure that site visitors do not notice the Web site's operating characteristics. This goal has not always been easy

to attain because the site's traffic volume has, on average, doubled each year since the site opened. Also, regular major improvements to the Lands' End site keep the Web team busy.

Lands' End's specific goals for performance change as Web technologies improve. For example, the site management team has a target for the time it takes one of the site's Web pages to load on a visitor's computer. In the early days of the site, that target was 15 seconds. Today, the target is under 3 seconds. The Web site's technical team has always taken a conservative approach to operating the site so that the site can meet its performance goals more easily. For example, the technical team specifies the maximum and average sizes of Web pages and graphics files that the content team can use. In addition, the technical team must complete all major changes to the site (including thorough testing) before November 1 each year, prior to the holiday selling season. Lands' End makes more than 40 percent of its total annual sales in November and December and does not want to take any chances with Web site changes during that time period.

The server hardware at Lands' End is a mix of **Sun** and **IBM** computers that are managed by another computer that allocates incoming Web traffic. Some of the Web site's advanced features, such as the graphics-intensive My Virtual Model, are created on a separate set of computers. These computers are all located at the Lands' End division headquarters in a small town near Madison, Wisconsin. The computers run a UNIX-based operating system from Sun called Solaris and a version of the Apache Web server software, about which you will learn more in this chapter. Although the Lands' End technical team writes some of the software that it uses to monitor the Web site's performance, the company also uses the services of Keynote Systems. Keynote can measure how fast particular pages load or how rapidly transactions are completed at various times of the day. Keynote can make these measurements at a number of locations around the world.

By paying close attention to the details, the technical team at Lands' End keeps the Web site operating at or above expected levels. When customers become so absorbed in the shopping experience that they do not notice the operation of the site, the technical team has done its job.

WEB SERVER BASICS

This chapter provides background information on the basic technologies used to build Web sites that can support online business operations. It includes a discussion of server software and hardware. It also includes an introduction to software that these sites use to perform utility functions such as site maintenance, diagnostics, and e-mail management. In later chapters, you will learn about software that accomplishes specific electronic commerce functions, such as order entry and processing, content management and delivery, user verification and security, and payment processing.

The main job of a Web server computer is to respond to requests from Web client computers. The three main elements of a Web server are the hardware (computers and related components), operating system software, and Web server software. All three of these elements must work together to provide sufficient capacity in a given situation.

After most companies have decided on the goals they want to accomplish with their Web sites, they begin developing their sites by estimating the number of visitors they expect to have, how many pages those visitors will view during an average visit, how large those pages will be (including graphics and other page elements), and the likely maximum number of simultaneous visitors. The next step is to determine the hardware and software combination that will work best to meet the needs of site visitors.

Types of Web Sites

An important first step in planning a Web server is to determine what the company wants to accomplish with the server. The company must estimate how many visitors will be connecting to the Web site and what types of files (graphics, multimedia, or text) will be delivered through the site. The company must also assess its existing information technology staff. Some companies have a large staff with a depth of experience, while others have a small or relatively inexperienced staff.

Companies create Web sites for a wide variety of reasons and in a wide variety of forms. Each has a different purpose, requires different computer hardware and software, and requires different monetary and personnel resources. Decisions about server hardware and software should be driven by the volume and type of Web activities expected. Types of sites include:

- *Development sites*: Simple sites that companies use to evaluate different Web designs with little initial investment. A development site can reside on an existing PC running Web server software. Multiple testers access the site through their client computers on an existing LAN.

- *Intranets*: Corporate networks that house internal memos, corporate policy handbooks, expense account worksheets, budgets, newsletters, and a variety of other corporate documents.
- *Extranets*: Intranets that allow certain authorized parties outside the company (such as suppliers or strategic partners) to access certain parts of the information stored in the system.
- *Transaction-processing sites*: Commerce sites such as business-to-business and business-to-consumer electronic commerce sites that must be available 24 hours a day, seven days a week. These sites must have spare server computers for handling high traffic volumes that occur periodically. In addition to requiring fast and reliable hardware, transaction-processing sites must run Web and commerce software that is efficient and easily upgraded when site traffic increases.
- *Content-delivery sites*: Sites that deliver content such as news, histories, summaries, and other digital information. Visitors must be able to locate articles quickly with a fast and precise search engine. The content must be presented rapidly on the visitor's screen. In general, these sites must be available 24 hours a day, seven days a week, just like transaction-processing sites. Hardware requirements for content sites are also similar to those of transaction-processing commerce sites.

Web Clients and Web Servers

When people use their Internet connections to become part of the Web, their computers become Web client computers on a worldwide client/server network. Client/server architectures are used in LANs, WANs, and the Web. In a client/server architecture, the client computers typically request services, such as printing, information retrieval, and database access, from the server, which processes the clients' requests. The computers that perform the server function usually have more memory and larger, faster disk drives than the client computers they serve. Recall from Chapter 2 that Web browser software (for example, Microsoft Internet Explorer or Netscape Navigator) is the software that makes computers work as Web clients. This software is also called Web client software.

The Internet connects many different types of computers running different types of operating system software. Because Web software is platform neutral, it lets these computers communicate with each other easily and effectively. This platform neutrality has been (and continues to be) a critical ingredient in the rapid spread and widespread acceptance of the Web. Figure 8-1 shows how the Web's platform neutrality provides multiple interconnections among a wide variety of client and server computers.

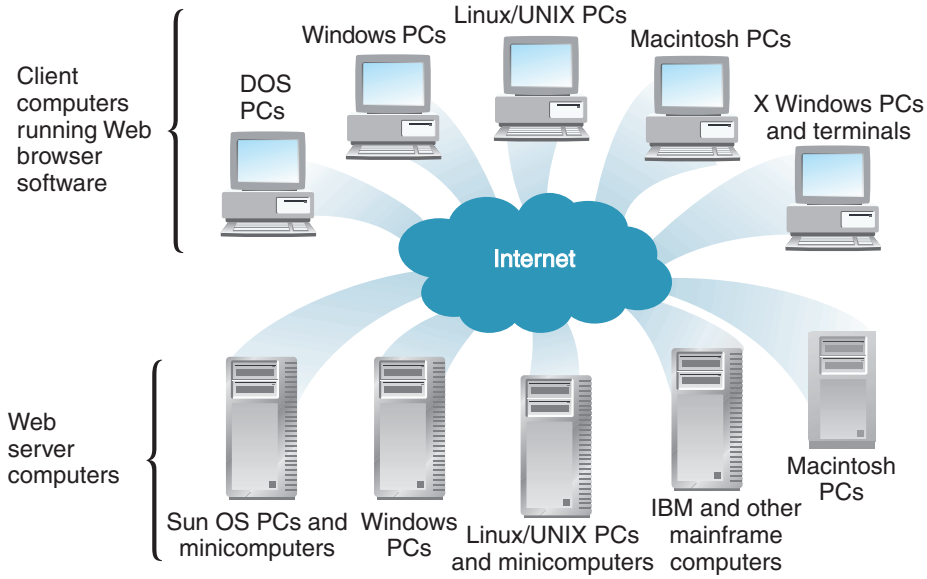


FIGURE 8-1 Platform neutrality of the Web

Dynamic Content

The mix and type of Web pages a system is likely to deliver in response to client requests can affect Web server performance. A **dynamic page** is a Web page whose content is shaped by a program in response to user requests, whereas a **static page** is an unchanging page retrieved from disk. A server delivering mostly static Web pages performs better than the same server delivering dynamic Web pages because static page delivery requires less computing power than dynamic page delivery. The largest performance differences between competing Web server products appear when servers deliver dynamic pages.

Dynamic content is nonstatic information constructed in response to a Web client's request. For example, if a Web client inquires about the status of an existing order by entering a unique customer number or order number into a form, the Web server searches the customer information (or sends a query to the back-end database in a higher tier) and generates a dynamic Web page based on the customer information it found (or that the database management software provided), thus fulfilling the client's request. Assembled from back-end databases and internal data on the Web site, a dynamic page is a specific response to the requester's query.

On a Web site that is a collection of HTML pages, the content on the site can be changed only by editing the HTML in the pages. This is cumbersome and does not allow customized pages to be produced in response to specific queries from site visitors. To create customized pages on the fly, Web sites use one of two basic approaches: server-side scripting or a dynamic page-generation technology.

Server-Side Scripting

The first Web sites to provide dynamic Web pages used an approach called server-side scripting. In **server-side scripting** (also called **server-side includes** or, more generally, **server-side technologies**), programs running on the Web server create the Web pages before sending them back to the requesting Web clients as parts of response messages.

Most server-side technologies are slow, so large Web sites used by online businesses today tend to use dynamic page-generation technologies.

Dynamic Page-Generation Technologies

Microsoft developed a now widely used dynamic page-generation technology called **Active Server Pages (ASP)**. Sun Microsystems developed a similar technology called **JavaServer Pages (JSP)**, and the open-source Apache Software Foundation sponsored a third alternative called **PHP**: **Hypertext Preprocessor (PHP)**. Yet another alternative is available from Macromedia in its **Cold Fusion** product.

In these page-generation technologies, server-side scripts are mixed with HTML-tagged text to create the dynamic Web page. For example, ASP allows Web programmers to use their choice of programming languages, such as VBScript, Jscript, or Perl. Java, a programming language created by Sun, can be used to produce dynamic pages. Such server-side programs are called **Java servlets**.

The Future of Dynamic Web Page Generation

Many critics of dynamic page-creation technologies note that these approaches do not really solve the problem of dynamic Web page generation. They argue that these dynamic page-creation approaches merely shift the task of creating dynamic pages from people who write HTML code to ASP (or JSP or PHP) programmers.

Several initiatives are under way that are directed at a more comprehensive solution to the dynamic Web page-creation problem. The **Apache Cocoon Project** is one of these initiatives. Apache Cocoon is a Web-development framework that allows programmers to query data that is in XML format and generate output in multiple formats, including HTML. The HTML output option makes Cocoon a useful tool for generating dynamic Web pages. In this approach, the content is stored with XML tags that describe the semantics (meaning) of each content item. The information request is handled by a Java servlet that can read the XML file and select the requested content items using the XML tags in the content file. Instead of creating a Web page, Cocoon can produce a response tailored to the request by applying a style sheet to the data. If a site visitor requests, for example, an Adobe Portable Document Format (PDF) file or a Wireless Markup Language (WML) file for display on a wireless handheld device, a Web site using Cocoon technology can generate the results in those file formats from its XML content files.

More recently, the Apache Cocoon Project has outlined a more complex model of the Web page-generation process that identifies four areas of concern (logic, content, style, and management). The latest version of Cocoon lets Web page developers divide the work into these four areas of concern and limits the interactions among the areas to the five specific contracts between the areas of concern shown in Figure 8-2.

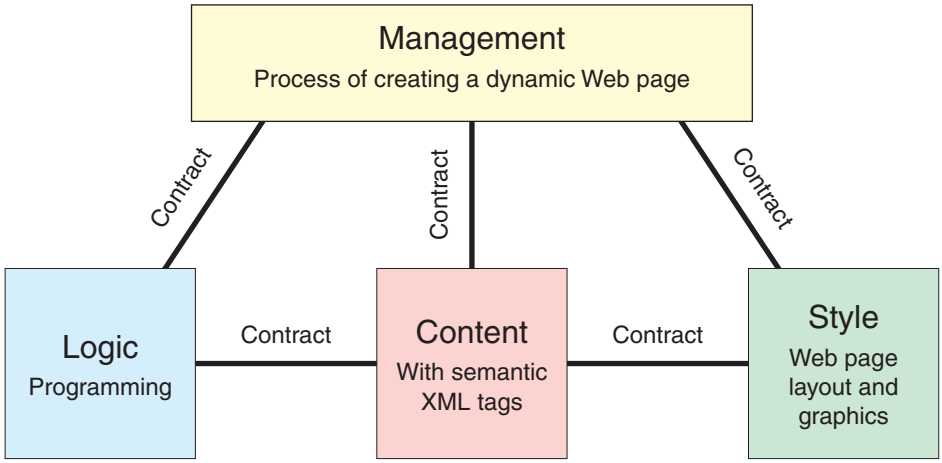


FIGURE 8-2 Apache Cocoon Project dynamic Web page generation model

Note that this model breaks the direct connection between logic and style. By separating the logic (the work of programmers) and style (the work of graphic artists) that is combined in the structure of HTML, Web designers believe that using the Cocoon model could make dynamic Web page design easier in the future. Many industry experts believe that the Apache Cocoon Project and similar development efforts by Microsoft (the **Microsoft.NET** Framework) and Oracle will provide better ways to generate dynamic Web pages in the future.

Various Meanings of "Server"

All computers that are connected to the Internet and contain documents that their owners have made publicly available through their Internet connections are called Web servers. Unfortunately, the term "server" is used in many different ways by information systems professionals. These multiple uses of the term can be confusing to people who do not have a strong background in computer technology. You are likely to encounter a number of different uses of the word "server."

A **server** is any computer used to provide (or "serve") files or make programs available to other computers connected to it through a network (such as a LAN or a WAN). The software that the server computer uses to make these files and programs available to the other computers is sometimes called **server software**. Sometimes this server software is included as part of the operating system that is running on the server computer. Thus, some information systems professionals informally refer to the operating system software on a server computer as server software, a practice that adds considerable confusion to the use of the term "server."

Some servers are connected through a router to the Internet. As you learned in Chapter 2, these servers can run software, called Web server software, that makes files on those servers available to other computers on the Internet. When a server computer is connected to the Internet and is running Web server software (usually in addition to the server software it runs to serve files to client computers on its own network), it is called a Web server.

Similar terminology issues arise for server computers that perform e-mail processing and database management functions. Recall that the server computer that handles incoming and outgoing e-mail is usually called an e-mail server, and the software that manages e-mail activity on that server is frequently called e-mail server software. The server computer on which database management software runs is often called a **database server**.

Thus, the word "server" is used to describe several types of computer hardware and software, all of which might be found in a typical electronic commerce operation. The only way to determine which server people are talking about when they use the term is from the context or by asking a clarifying question. If you hear a computer technician say, "The server is down today," the problem might be in the hardware, the software, or a combination of the two!

Web Client/Server Communication

In Chapter 2, you learned how the Web is software that runs on the Internet. In this section, you will learn more about how Web client and Web server software work. When a person uses a Web browser to visit a Web site, the Web browser (also known as a Web client) requests files from the Web server at the company or organization that operates the Web site. Using the Internet as the transportation medium, the request is formatted by the browser using HTTP and sent to the server computer. A moment later, when the server receives the request, it retrieves the file containing the Web page or other information that the client requested, formats it using HTTP, and sends it back to the client over the Internet.

When the requested information—a file containing the text and markup tags of a Web page, in this instance—arrives at the client computer, the Web browser software determines that the information is an HTML page. It displays the page on the client machine according to the directions defined in the page's HTML code. This process repeats as the client requests, the server responds, and the client displays the result. Sometimes, a single client request results in dozens or even hundreds of separate server responses to locate and deliver information.

A Web page containing many graphics and other objects can be slow to appear in the client's Web browser window because each page element (each graphic or multimedia file) requires a separate request and response.

Two-Tier Client/Server Architecture

The basic Web client/server model is a two-tier model because it has only one client and one server. All communication takes place on the Internet between the client and the server. Of course, other computers are involved in forwarding packets of information across the Internet, but the messages are created and read only by the client and the server computers in a **two-tier client/server architecture**. Figure 8-3 shows how a Web client and a Web server communicate with each other in a two-tier client/server architecture.

The message that a Web client sends to request a file or files from a Web server is called a **request message**. A typical request message from a client to a server consists of three major parts:

- Request line
- Optional request headers
- Optional entity body

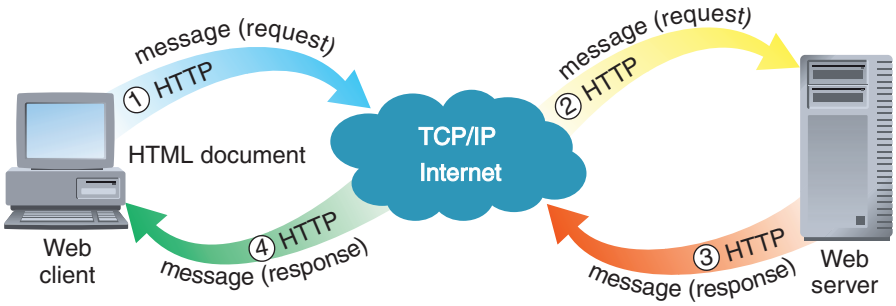


FIGURE 8-3 Message flows in a two-tier client/server network

The **request line** contains a command, the name of the target resource (a filename and a description of the path to that file on the server), and the protocol name and version number. Optional **request headers** can contain information about the types of files that the client will accept in response to this request. Finally, an optional **entity body** is sometimes used to pass bulk information to the server.

When the server receives the request message, it executes the command included in the message (in this case, it sends a particular Web page file back to the client). The server does this by retrieving the Web page file from its disk (or another disk on a network to which it is connected) and then creating a properly formatted **response message** to send back to the client. A server's response consists of three parts that are identical in structure to a request message: a response header line, one or more response header fields, and an optional entity body. In the response, however, each part has a slightly different function than it does in the request. The **response header line** indicates the HTTP version used by the server, the status of the response (whether the server found the file that the client wanted), and an explanation of the status information. Response header fields follow the response header line. A **response header field** returns information describing the server's attributes. The entity body returns the HTML page requested by the client machine.

Three-Tier and N-Tier Client/Server Architectures

Although the two-tier client/server architecture works well for the delivery of Web pages, a Web site that delivers dynamic content and processes transactions must do more than respond to requests for Web pages. A **three-tier architecture** extends the two-tier architecture to allow additional processing (for example, collecting the information from a database needed to generate a dynamic Web page) to occur before the Web server responds to the Web client's request. Higher-order architectures—that is, those that have more than three tiers—are usually called **n-tier architectures**. The third tier often includes databases and related software applications that supply information to the Web server. The Web server can then use the output of these software applications when responding to client requests, instead of just delivering a Web page. Architectures that have four, five, or even more tiers include software applications (just as the three-tier systems), but they also include the databases and database management programs that work with the software

applications to generate information that the Web server can turn into Web pages, which it then sends to the requesting client.

A good example of services supported by a database in an n-tier architecture is a catalog-style Web site with search, update, and display functions. Assume that a user requests a display of a company's exotic fruit selections. The client request is formulated into an HTTP message by the Web browser, sent over the Internet to the Web server, and examined by the Web server. The Web server analyzes the request and determines that responding to the request requires the help of the server's database. The server sends a request to the database management software to search for, retrieve, and return all information about exotic fruit in the catalog database. The database information flows back through the database management software system to the server, which formats the response into an HTML document and sends that document inside an HTTP response message back to the client over the Internet.

Three-tier and n-tier systems can track customer purchases stored in shopping carts, look up sales tax rates, keep track of customer preferences, query inventory databases, and keep the company catalog current. Figure 8-4 shows an overview of information flows in a three-tier architecture. Numbers on the flow arrows indicate the order in which the messages flow over the indicated paths.

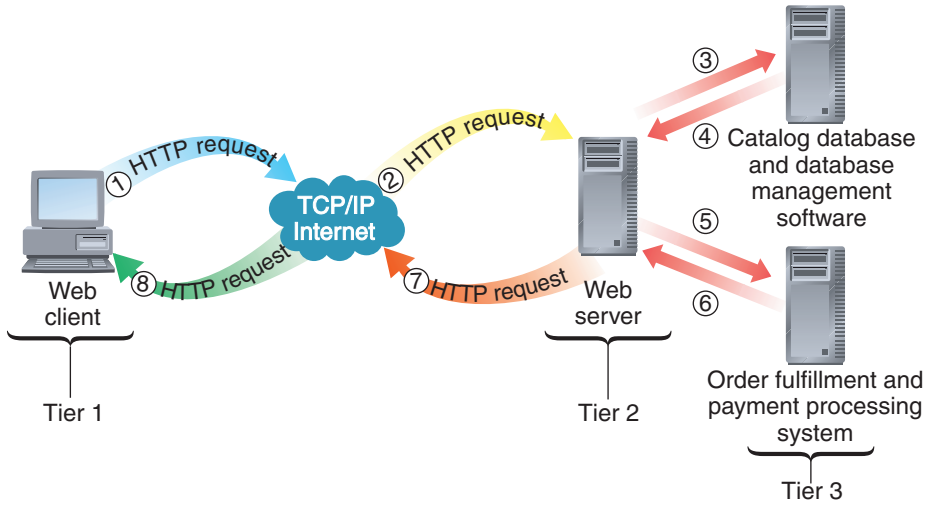


FIGURE 8-4 Message flows in a three-tier client/server network

SOFTWARE FOR WEB SERVERS

Some Web server software can run on only one computer operating system, while some can run on several operating systems. In this section, you will learn about the operating system software used on most Web servers, the Web server software itself, and other programs, such as Internet utilities and e-mail software, that companies often run on Web servers or other computers as part of electronic commerce operations.

Operating Systems for Web Servers

Operating system tasks include running programs and allocating computer resources such as memory and disk space to programs. Operating system software also provides input and output services to devices connected to the computer, including the keyboard, monitor, and printers. A computer must have an operating system to run programs. For large systems, the operating system has even more responsibilities, including keeping track of multiple users logged on to the system and ensuring that they do not interfere with one another.

Most Web servers run on computers that use one of the following operating systems: Microsoft Windows NT Server, Microsoft Windows 2000 Server or Server 2003 products, Linux, or one of several UNIX-based operating systems, such as Solaris or FreeBSD. Many companies believe that **Microsoft server products** are simpler for their information systems staff to learn and use than UNIX-based systems. Other companies worry about the security weaknesses caused by the tight integration between application software and the operating system in Microsoft products. UNIX-based Web servers are more popular, and many users believe that UNIX is a more secure operating system on which to run a Web server.

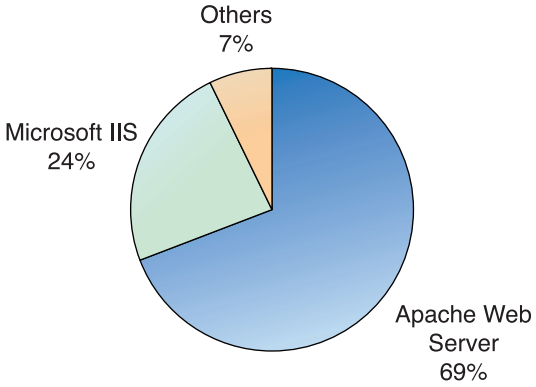
Linux is an open-source operating system that is fast, efficient, and easy to install. **Open-source software** is developed by a community of programmers who make the software available for download at no cost. Other programmers then use the software, work with it, and improve it. Those programmers can submit their improved versions of the software back to the community. You can learn more about open-source software at the **Open Source Initiative** Web site.

An increasing number of companies that sell computers intended to be used as Web servers include the Linux operating system in default configurations. Although Linux can be downloaded free from the Web, most companies buy it through a commercial distributor. These commercial distributions of Linux include useful additional software, such as installation utilities, and a support contract for the operating system. Commercial Linux distributors that sell versions of the operating system with utilities for Web servers include **Caldera**, **Mandrake**, **Red Hat**, and **SuSE**. **Sun Microsystems** sells Web server hardware along with its UNIX-based operating system, **Solaris**.

Web Server Software

This section describes the most commonly used Web server programs today: Apache HTTP Server, Microsoft Internet Information Server (IIS), and Sun Java System Web Server (JSWS) (often called by its former names, Sun ONE, iPlanet Enterprise Server, and Netscape Enterprise Server). These popularity rankings were accumulated through surveys done by **Netcraft**, a networking consulting company in Bath, England, known throughout the world for its Web server survey. Netcraft continually conducts surveys to tally the number of Web sites in existence and measure the relative popularity of Internet Web server software.

Figure 8-5 shows the use of Web server software by active sites in October 2005. You can click the **Netcraft Surveys** link in the Online Companion to check out the latest Web survey results.



Adapted from NetcraftWeb Surveys, October 2005, http://www.netcraft.com.

FIGURE 8-5 Web server software used by active sites, October 2005

The Netcraft Web server surveys show that the market share of Web server software has stabilized in recent years. Apache generally holds between 65 and 70 percent of the market, and Microsoft's IIS usually holds between 20 and 25 percent of the market. Sun, along with the **National Center for Supercomputing Applications (NCSA)** Web server and a few other products such as Zeus, account for the remainder of the market. The NCSA Web server was one of the first Web servers developed in the United States. Because it was developed with U.S. government research funds, the NCSA software is available at no cost.

According to a *PC Magazine* survey (see the Alwang reference listed in the For Further Study and Research section at the end of the chapter), the market share percentages for intranet Web servers are quite different than for public Web servers. Although the Web server software packages described in this chapter are all top selections among intranet servers, Microsoft IIS and the various Sun servers (JSWS, Sun ONE, iPlanet, Netscape) together account for 75 percent of installed intranet server programs. Other recent surveys show that large company Web sites and high traffic Web sites show a greater use of the Sun servers than the Netcraft Surveys report.

The performance of one Web server differs from that of another based on workload, operating system, and the size and type of Web pages served. *PC Magazine* evaluates computer products regularly. In its tests, some Web server software fared well when delivering static HTML pages, but other Web server software performed better when delivering dynamic Web page content. The differences among servers can be significant; picking the right server for each different business need is critical (see the Machrone reference listed in the For Further Study and Research section). The sections that follow contain descriptions of the main Web server systems used in various electronic commerce applications.

Apache HTTP Server

Apache is an ongoing group software development effort. Rob McCool developed Apache while he was working at the University of Illinois at the **NCSA** in 1994. Several Webmasters from around the world created their own extensions to the server and formed an e-mail group so that they could coordinate their changes (known as "patches") to the

system. The system consisted of the original core system with a lot of patches—thus, it became known as "a patchy" server, or simply, "Apache." The Apache Web server is currently available on the Web at no cost as open-source software.

Apache HTTP Server has dominated the Web since 1996 because it is free and performs very efficiently. It is powerful enough that IBM includes it in its WebSphere application server package. Other Web server products, such as **Zeus**, are based on the Apache open-source code. Currently, Apache is used on 65 to 70 percent of all Web servers, which means it is more widely used than all other Web server software packages combined. Apache runs on many operating systems (including FreeBSD-UNIX, HP-UX, Linux, Microsoft Windows, SCO-UNIX, and Solaris) and the hardware that supports them.

Microsoft Internet Information Server

Microsoft Internet Information Server (IIS) comes bundled with current versions of Microsoft Windows Server operating systems. IIS is used on many corporate intranets because many companies have adopted Microsoft products as their standard products. Small sites running personal Web pages use IIS, as do some of the largest electronic commerce sites on the Web. Most current surveys estimate that about 20 to 25 percent of all Web servers run some version of IIS. In recent years, the number of Web sites running IIS has been decreasing. Most industry observers believe this decrease has occurred because IIS has been the victim of several well-publicized security breaches. These security breaches allowed Web servers running IIS to be attacked successfully and defaced. You will learn more about Web server security threats and countermeasures in Chapter 10.

IIS, as a Microsoft product, was originally designed to run only on the Windows NT and Windows 2000 operating systems. It has been released for Microsoft Windows Server 2003 and runs on the Windows XP operating system, but it is not included as a standard part of Windows XP. IIS supports the use of ASP, ActiveX Data Objects, and SQL database queries. IIS also includes the Microsoft FrontPage Web site development tool and other reporting tools. IIS's inclusion of ASP provides an application environment in which HTML pages, ActiveX components, and scripts can be combined to produce dynamic Web pages.

Sun Java System Web Server (Sun ONE, iPlanet, Netscape)

A descendant of the original NCSA Web server program, Sun Java System Web Server (JSWS) was formerly sold under the names Sun ONE, Netscape Enterprise Server, and iPlanet Enterprise Server. When AOL (now Time Warner) purchased Netscape in 1999, the company formed a partnership with Sun Microsystems to support and continue to develop Netscape server products. This partnership was named iPlanet and was operated under a three-year agreement that expired in March 2002. When the partnership ended, iPlanet became a part of Sun because the Web server and electronic commerce software that iPlanet sells are more closely related to Sun's businesses than to Time Warner's businesses.

Sun JSWS is not free, but its licensing fee is reasonable. The fee varies with the processing power of the server on which it is installed, but most Web sites pay between $1400 and $5000 for their licenses. The Sun software runs on many operating systems, including HP-UX, Solaris, and Windows. According to recent estimates, Sun JSWS runs on about 1 percent of all Web servers. However, some of the busiest and best-known sites on the Internet, including BMW, Dilbert, E*TRADE, Excite, Lycos, and Schwab, run (or have run)

some version of Sun JSWS. Reports from consulting firms such as Gartner, Inc. show that Sun JSWS is in use at more than 40 percent of all public Web sites and at more than 60 percent of the top 100 enterprise Web sites.

Like most other server programs, Sun JSWS supports dynamic application development for server-side applications. Sun JSWS provides connectivity to a number of database products as well.

Finding Web Server Software Information

People who want to know the type of operating system and Web server software that a Web site is running can visit the **Netcraft** Web site. On Netcraft's home page is a link named "What's that site running?" that leads to a page with a search function. Visitors can use that search function to find out what operating system and what Web server software a specific site is now running and what the site ran in the past.

ELECTRONIC MAIL (E-MAIL)

Although the Web, with its interactions between Web servers and clients, is the most important technology used in electronic commerce today, many buyers and sellers also use e-mail to gather information, execute transactions, and perform other tasks related to electronic commerce. E-mail originated in the 1970s on the ARPANET. Although the goals of the ARPANET were to control weapons systems and transfer research files, general communications uses emerged on the network. As you learned in Chapter 2, in 1972, ARPANET researcher Ray Tomlinson wrote a program that could send and receive messages over the network. Today, e-mail is the most popular form of business communication—far surpassing the telephone, conventional mail, and fax in volume.

E-Mail Benefits

Not only was e-mail one of the first Internet applications, it was also one reason that many people were originally attracted to the Internet. E-mail conveys messages from one destination to another in a few seconds. Messages can contain simple ASCII text, or they can contain character formatting similar to word-processing programs.

One useful feature of e-mail is that documents, pictures, movies, worksheets, or other information can be sent along with the message itself. These **attachments** are frequently the most important part of the message. A business e-mail message attachment might contain an invoice, a 200-page wholesale catalog, or a set of Web pages that describe the company's products. Many electronic commerce sites use e-mail to confirm the receipt of customer orders and then the shipment of items ordered. Software vendors can also use e-mail to send information about a purchase to the buyer.

E-Mail Drawbacks

Despite its many benefits, e-mail does have some drawbacks. One annoyance associated with e-mail is the amount of time that businesspeople spend answering their e-mail today. Researchers have found that most managers can deal with e-mail messages at an average rate of about five minutes per message. Some messages can be deleted within a few seconds, but those are balanced by the e-mails that require the manager to spend much more

time finding facts, checking files, making phone calls, and doing other tasks as part of answering e-mail. Researchers have found that most people (not including those people who answer e-mails as a full-time job) begin to resent the time that e-mail consumes when they start getting more than 20 or 30 messages a day. At that point, the average person is spending about two hours a day answering e-mail.

A second major irritation brought by e-mail is the **computer virus**, more simply known as a **virus**, which is a program that attaches itself to another program and can cause damage when the host program is activated. Recall that e-mail messages can carry attachments. Although many of these attached files contain useful information, attached files can contain a virus program or other security threat. Using virus protection software and dealing with e-mailed security threats is a cost that all must bear for the convenience of using e-mail. You will learn more about computer viruses and other threats that can be transmitted through e-mail (and how to control them) in Chapter 10.

As you learned in Chapter 2, the most frustrating and expensive problem associated with e-mail today is the issue of unsolicited commercial e-mail, also known as UCE or **spam**. This nagging problem is discussed in the next section.

Spam

Figure 8-6 shows the rapid increase in the proportion of all e-mail entering business e-mail servers that is spam. The sheer magnitude of the spam problem is hard to believe. During one 24-hour period in 2005, researchers estimated that 106 billion spam e-mail messages were sent. Many researchers who track the growth in spam believe that current trends will continue and that more than 90 percent of all e-mail messages (including messages transmitted to both business and personal users) will be spam before any effective technical solutions can be implemented. Other researchers believe that the growth of spam is showing signs of leveling out.

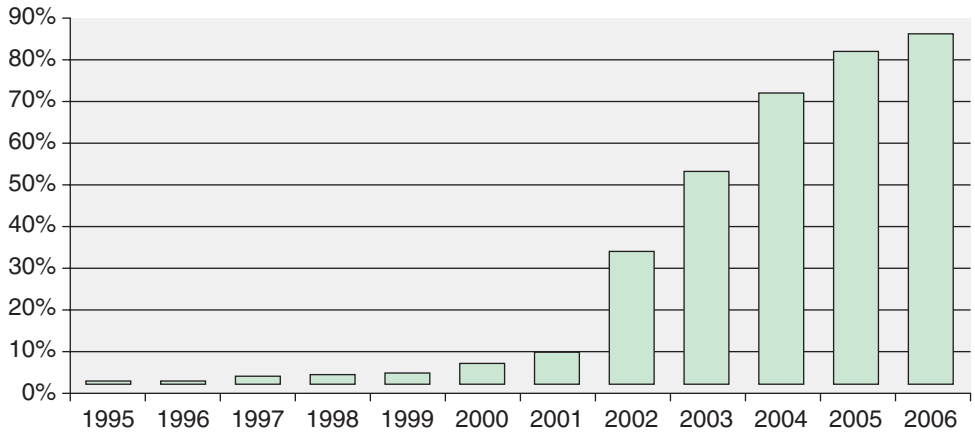


FIGURE 8-6 Growth of spam as a proportion of all business e-mail

Many grassroots and corporate organizations have decided to fight spam aggressively. America Online, for example, has taken an active role in limiting spam through legal channels. In 2005, America Online reported that the amount of spam passing through its servers declined for the first time in five years. The company attributed this success to the effects of lawsuits it has filed against spammers and its improved technical abilities to detect and delete spam e-mails before they are distributed to its users. A number of companies now offer software that organizations can run on their e-mail server computers to limit the amount of spam that gets through to their employees. Although individual users can install client-based spam-filtering programs on their computers or set filters that might be available within their e-mail client software, most companies find it more effective and less costly to eliminate spam before it is downloaded to user computers. These antispam efforts and software products can help limit the annoyance and cost of spam.

Solutions to the Spam Problem

As spam has grown to become a serious problem for all users of e-mail, an increasing number of approaches have been devised or proposed. Some of these approaches require the passing of laws, and some require technical changes in the mail-handling systems of the Internet. Other approaches can be implemented under existing laws and with current technologies, but only if large numbers of organizations and businesses cooperate. A few tactics that reduce spam can be undertaken by individual e-mail users. In the sections below you will learn more about each of these approaches to controlling the spam problem.

Individual User Antispam Tactics

One way individuals can limit spam is to reduce the likelihood that a spammer can automatically generate their e-mail addresses. Many organizations create e-mail addresses for their employees by combining elements of each employee's first and last names. For example, small companies often combine the first letter of an employee's first name with the entire last name to generate e-mail addresses for all employees at small companies. (Larger companies often must use more complex algorithms as they are likely to have both a Jane Smith and a Judy Smith working for them.) Any spam sender able to obtain an employee list can generate long lists of potential e-mail addresses using the names on the list. If no employee list is available, the spam sender can simply generate logical combinations of first initials and common names. The cost of sending e-mail is so low that a spammer can afford to send thousands of e-mails to randomly generated addresses in the hope that a few of them are valid. By using an e-mail address that is more complex, such as "xq7yy23@mycompany.com," individuals can reduce the chances that a spammer can randomly generate his or her address. Of course, such an address is hard to remember, which somewhat defeats the purpose of e-mail as a convenient way to communicate.

A second way to reduce spam is to control the exposure of an e-mail address. Spammers use software robots to search the Internet for character strings that include the "@" character (which appears in every e-mail address). These robots search Web pages, discussion boards, chat rooms, and any other online source that might contain e-mail addresses. Again, the spammer can afford to send thousands of messages to e-mail addresses gathered in this way. Even if only one or two people respond, the spammer can earn a profit because the cost of sending e-mail messages is so low.

Some individuals use multiple e-mail addresses to thwart spam. They use one address for display on a Web site, another to register for access to Web sites, another for shopping accounts, and so on. If a spammer starts using one of these addresses, the individual can stop using it and switch to another. Many Web hosting services include a large number (often 100–200) of e-mail addresses as part of their service, so this can be a useful tactic for people or small businesses with their own Web sites.

These three strategies focus on limiting spammer's access to or use of an e-mail address. Other approaches use one or more techniques that filter e-mail messages based on their contents.

Basic Content Filtering

All content-filtering solutions require software that identifies content elements in an incoming e-mail message that indicate the message is (or is not) spam. The content-filtering techniques differ in which content elements they examine, whether they look for indications that the message is spam or that it is not spam, and how strictly they apply the rules for classifying messages. Most basic content filters examine the e-mail headers (From, To, Subject) and look for indications that the message might be spam. The software that performs the filtering task can be placed on individual users' computers (called **client-level filtering**) or on mail server computers (called **server-level filtering**). Server-level filtering can be implemented on an ISP's mail server, an individual company's mail server, or both. Also, many individuals that have ISP and/or company mail servers that filter their e-mail also install client-level filters on their computers. Spam that gets through one filter can be trapped by another filter.

The most common basic content-filtering techniques are black lists and white lists. A **black list spam filter** looks for From addresses in incoming messages that are known to be spammers. The software can delete the message or put it into a separate mailbox for review. A black list spam filter can be implemented at the individual, organization, or ISP level. Several organizations, such as the **Mail Abuse Protection System** and the **Open Relay Database**, collect black lists and make them available to ISPs and company e-mail administrators. Other groups, such as the **Spamhaus Project**, track known spammers and publish lists of the mail servers they use. Some of these are free services, others charge a fee. The biggest drawback to the black list approach is that spammers frequently change their e-mail servers, which means that a black list must be continually updated to be effective. This updating requires that many organizations cooperate and communicate information about known spammers. In addition to its black list, the Spamhaus Project maintains a list of known spammers on its site. These are individuals and companies who have had their services terminated by an ISP for spam-related violations of an acceptable use policy more than three times. The Spamhaus Project provides detailed information about those on this list to law enforcement agencies.

A **white list spam filter** examines From addresses and compares them to a list of known good sender addresses (for example, the addresses in an individual's address book). A white list filter is usually applied at the individual user level, although it is possible to do the filtering at the organization level if the e-mail administrator has access to all individuals' address books (some companies mandate such access for security purposes). The main drawback to this approach is that it filters out any messages sent by unknown parties, not just spam. Because the number of **false positives** (messages that are rejected but should not

have been) can be very high for white list filters, the rejected e-mails are always placed into a review mailbox instead of being deleted.

White list and black list approaches can be used in client-level or server-level filters, but both have serious drawbacks. To overcome these drawbacks, the two approaches are often used together or with other content-filtering approaches to achieve an acceptable level of filtering without an excessive false positive rate.

Challenge-Response Content Filtering

One content-filtering technique uses a white list as the basis for a confirmation procedure. This technique, called **challenge-response**, compares all incoming messages to a white list. If the message is from a sender who is not on the white list, an automated e-mail response is sent to the sender. This message (the challenge) asks the sender to reply to the e-mail (the response). The reply must contain a response to a challenge presented in the e-mail.

These challenges are designed so that a human can respond easily, but a computer would have difficulty formulating the response. For example, a good challenge might include a picture of a fruit bowl and would ask the sender to respond with the number of apples in the bowl. This prevents a spammer from setting up a computer that receives challenges and answers them (the program would have difficulty identifying and counting the number of apples). It would be inefficient for a spammer to hire a human to respond to thousands of challenges. To learn more about this technique, you can visit the **CAPTCHA Project** site at Carnegie Mellon University.

One major drawback to challenge-response systems is that they can be abused. For example, a perpetrator could send out thousands of e-mails to recipients that use challenge-response systems. If the perpetrator includes the victim's e-mail as the From address in those e-mails, the victim will be bombarded by the automated challenges sent out by the challenge-response systems of the recipients. What is worse, the potential damage of this tactic becomes greater as more e-mail servers install challenge-response systems.

Another issue with challenge-response systems will arise if they become widespread. Most mail that any individual receives from unknown senders is spam. A challenge-response system sends a challenge message to every unknown sender. That is, for every spam message received, a second e-mail is sent. A challenge-response system thus doubles the amount of useless e-mail messages that must be handled by the Internet's infrastructure. If everyone were to use a challenge-response system, the Internet capacity wasted by spam would approximately double. Because of the drawbacks associated with challenge-response systems, most industry experts agree that they are, at best, a limited short-term solution.

Advanced Content Filtering

Advanced content filters that examine the entire e-mail message can be more effective than basic content filters that only examine the message headers or the IP address of the e-mail's sender. Creating effective content filters can be challenging. For example, a company might want to delete any e-mail message that includes the word "sex." If the company deletes all e-mails containing that character string, they will unintentionally delete all e-mailed orders from customers in the town of Essex.

Many advanced content filters operate by looking for spam indicators throughout the e-mail message. When the filter identifies an indicator in a message, it increases that message's spam "score." Some indicators increase the score more than others. Indicators can be words, word pairs, certain HTML codes (such as the code for the color white, which makes part of the message invisible in most e-mail clients), and information about where a word occurs in the message. Unfortunately, as soon as spam filter vendors identify a good set of indicators, spammers stop including those indicators in their messages.

One type of advanced content filter that is based on a branch of applied mathematics called Bayesian statistics shows some promise of staying one step ahead of the spammers. **Bayesian revision** is a statistical technique in which additional knowledge is used to revise earlier estimates of probabilities. In software that contains a **naïve Bayesian filter** (the most common type in use today), the software begins by not classifying any messages. The user reviews messages and indicates to the software which messages are spam and which are not. The software gradually learns (by revising its estimates of the probability that a message element appears in a spam message) to identify spam messages.

After seeing a few dozen messages classified, the naïve Bayesian filter can successfully classify spam messages about 80 percent of the time. As the filter continues to work, the user reviews its classifications and tells the software when it makes a mistake. After classifying a few hundred messages (and being corrected by the user when it errs), a naïve Bayesian filter typically reaches correct spam classification rates above 95 percent. Although these filters are highly effective and have low false positive rates, they must be trained, which takes time. The training is best done by each individual user because one person's spam can be another person's important message. Having users train their own filters provides the most rapid training and the best results. Most organizations do not currently use naïve Bayesian filters because they require attention by individual e-mail users. However, naïve Bayesian filters can be installed on some client computers (such as those used by people who receive large amounts of e-mail) in organizations that also use other techniques (such as white list or black list filters) at the server level. Most industry observers expect to see naïve Bayesian filtering used more widely in the future as the spam problem worsens and as more e-mail clients include such filters.

A number of researchers presented papers on the development of naïve Bayesian filters in mid-2002. Later that year, an open-source software development project led by John Graham-Cumming released one of the first functional Bayesian filter products for individual users, **POPFile**. POPFile is a program that installs on individual client computers and works with many different e-mail clients (including Microsoft Outlook, Pegasus, and Qualcomm Eudora) to provide content filtering. Because it is open-source software, POPFile is free (although the project team welcomes donations). POPFile does require that e-mail be retrieved using a POP (Post Office Protocol) connection, so it cannot be used with most Web-based e-mail accounts such as Yahoo! or Hotmail. The latest releases of some e-mail client software, such as Qualcomm Eudora and Mozilla Thunderbird, now include naïve Bayesian filtering tools that work the same way POPFile does.

Figure 8-7 shows the training screen from an installation of POPFile. Each message is listed and classified into categories that the software calls buckets. These buckets are configurable by the user; in this case, the user has created two buckets, "spam" and "OK." The user reviews each message and changes the classification if necessary. Each time the user changes a classification, POPFile revises its internal database using a naïve Bayesian algorithm and uses the revised rules to classify new e-mail messages.

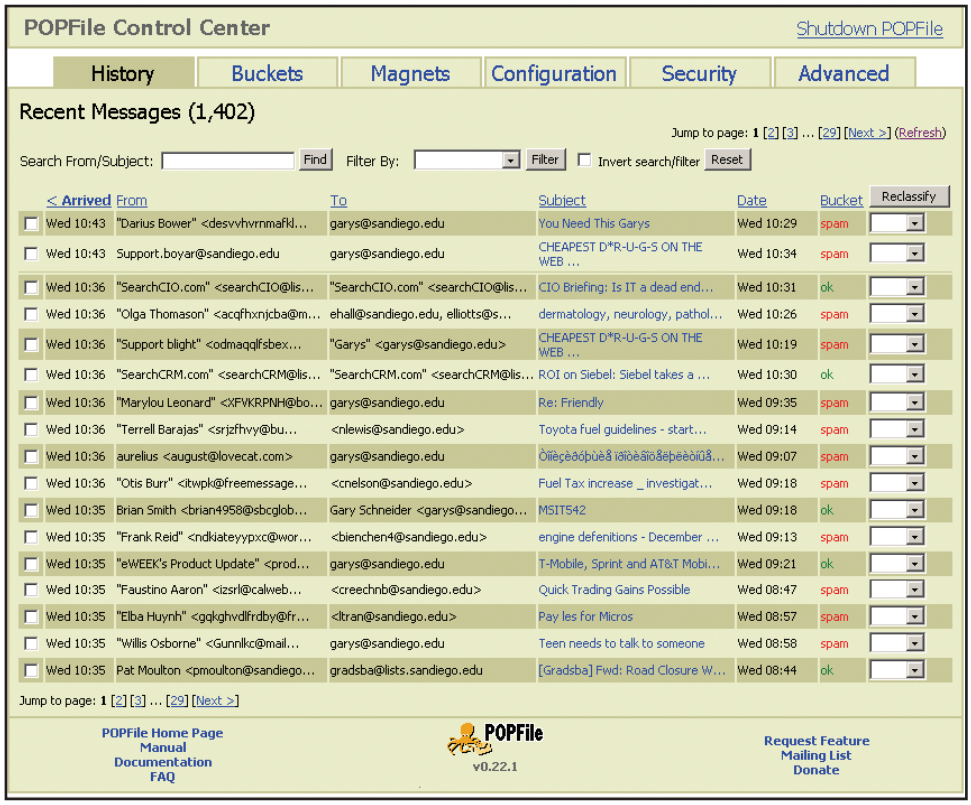


FIGURE 8-7 Training screen in the POPFile naïve Bayesian filter

Figure 8-8 shows the summary statistics page from a POPFile filter. This page reflects the filter's activity during a recent six-month period on one of the author's e-mail accounts.

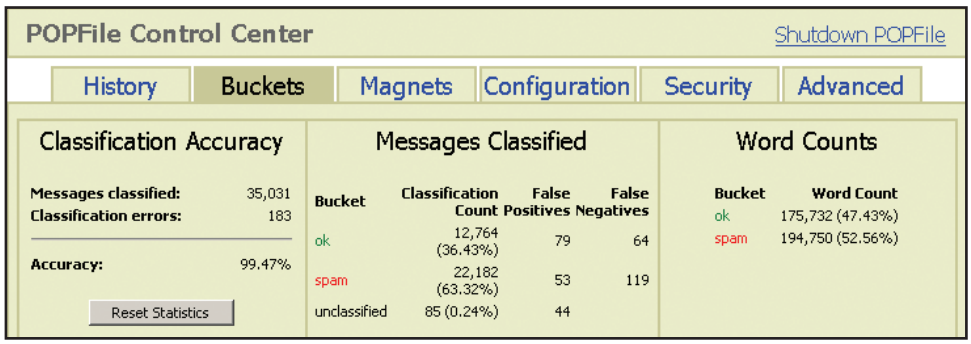


FIGURE 8-8 POPFile summary statistics page

Although the filter caught only 30 percent of spam messages when it was installed, within two weeks, it was catching more than 90 percent and eventually was more than 99 percent accurate. Note that the number of false positives in the spam category is also quite small. POPFile includes a feature, called magnets, that allows the user to implement white list and black list filtering. The user can create a magnet that classifies messages based on specific content in the message and does not send the classified message through the naïve Bayesian filter. In this example, magnets that classify messages as "OK" operate as white list filters and magnets that classify messages as "spam" operate as black list filters.

Naïve Bayesian filters are very effective client-level filters, but they do not work well as server-level filters. The content that is common in one person's spam might be common in another person's valid e-mail; therefore, one user's reclassifications tend to cancel out those of other users. This prevents the filter from building its accuracy to high levels. One good solution for organizations is to use black list filters at the server level combined with white list and naïve Bayesian filters at the client level. The major drawback of any client-level filtering approach is that it requires individual users to update their own filters regularly. Although it takes less time to update a filter than to delete hundreds of spam messages, it still does take time.

Legal Solutions

A number of U.S. jurisdictions have passed laws that provide penalties for the sending of spam. In January 2004, the U.S. CAN-SPAM law (the law's name is an acronym for "Controlling the Assault of Non-Solicited Pornography and Marketing") went into effect. Researchers who track the amount of spam noted a drop in the percentage of all e-mail that was spam in February and March. A **MessageLabs** study tracked the drop from 62 percent in January to 59 percent in February and 53 percent in March. However, by April, the rate was back up to a new high, 68 percent. It appears that spammers slowed down their activities immediately after the effective date of CAN-SPAM to see if a broad federal prosecution effort would occur. When the threat did not materialize, the spammers went right back to work. Today, most industry analysts estimate that more than 80 percent of all e-mail messages are spam.

The CAN-SPAM law is the first U.S. federal government effort to legislate controls on spam. It regulates all e-mail messages sent for the primary purpose of advertising or promoting a commercial product or service, including messages that promote the content displayed on a Web site. The law's main provisions include:

- *Misleading address header information*: E-mail headers and routing information, including the originating domain name and e-mail address, must be accurate and must identify the person who sent the e-mail.
- *Deceptive subject headers*: The e-mail's subject line cannot mislead the recipient about the contents or subject matter of the message.
- *Clear and conspicuous notice of message nature*: The e-mail must contain a clear and conspicuous notice that the message is an advertisement or solicitation and that the recipient can opt out of receiving further commercial e-mail from the sender.
- *Physical postal address*: The e-mail must include the sender's valid physical postal address.

- *Mandatory provision of an opt-out mechanism*: The e-mail must include a return e-mail address or another Internet-based response mechanism that allows a recipient to ask not to be sent future e-mail messages. These requests must be honored. The message may include a menu of choices that allows a recipient to opt out of certain types of messages, but one option on the menu must be an option to stop sending all commercial messages of any type.
- *Effectiveness of opt-out mechanism*: Opt-out requests must be honored within 10 business days. Any opt-out mechanism offered must be able to process opt-out requests for at least 30 days after the e-mail is sent. Once an opt-out request has been received, the sender is prohibited from helping any other entity send e-mail to the opt-out address or from having another entity send e-mail on the sender's behalf to that address.
- *Transfer of e-mail addresses*: Once a recipient has submitted an opt-out request, the sender is prohibited from selling or transferring that e-mail address to any other entity.

The law also prohibits misleading address header information in e-mail messages that facilitate an agreed-upon transaction or that update a customer in an existing business relationship. Each violation of a provision of the law is subject to a fine of up to $11,000. Additional fines are assessed for those who violate one of the above provisions and do one or more of the following:

- Harvest e-mail addresses from Web sites or Web services that have published a notice prohibiting the transfer of e-mail addresses for the purpose of sending e-mail.
- Send e-mail messages to addresses that have been generated by combining names, letters, or numbers into multiple combinations and permutations.
- Use scripts or other automated tools to register for multiple e-mail or user accounts that are then used to send commercial e-mail.
- Relay e-mails through a computer or network without the permission of the computer's or network's owner.

As you can see, a successful prosecution could cost the convicted spammer a great deal of money. The law further provides for criminal penalties, including imprisonment, for commercial senders of e-mail who do or conspire to do any of the following:

- Use another person's or entity's computer to send commercial e-mail from or through it without the computer owner's permission.
- Use a computer to relay or retransmit multiple commercial e-mail messages with the intent to deceive or mislead recipients or an Internet access service about the origin of the messages.
- Send multiple e-mail messages that contain false header information.
- Present false identification when registering for multiple e-mail accounts or domain names.
- Falsely represent themselves as owners of multiple IP addresses that are used to send commercial e-mail messages.

You can learn more about the law on the **U.S. Federal Trade Commission CAN-SPAM Law** information pages. The FTC issues new rules from time to time under the law.

To obtain current updates on those rules, visit the **U.S. Federal Trade Commission Spam** information pages.

In the months following CAN-SPAM's effective date, several large lawsuits were filed under the act. Microsoft, AOL, and Earthlink collectively filed six lawsuits and Microsoft filed an additional eight lawsuits on its own shortly thereafter. Beyond these actions, only a few spammers have been prosecuted under the law. The most significant verdict to date was the December 2003 verdict issued by a U.S. District Court in Iowa. That court used a combination of Iowa law and federal racketeering statutes to order three spammers to pay more than $1 billion to an ISP they had inundated with spam. Despite these few headline-grabbing cases, the large wave of prosecutions that many observers had hoped to see has not yet occurred. The FTC refused to create a do-not-spam list that would have been modeled after its do-not-call list, which has been reasonably successful in limiting marketers' phone calls.

Few industry experts expect CAN-SPAM or similar laws to be effective in stemming the tide of spam. After all, spammers have been violating existing deceptive advertising laws for years. Many spammers use mail servers located in countries that do not have (and that are unlikely to adopt) antispam laws. As you learned in Chapter 7, the issues of jurisdiction can be unclear for businesses that operate online. Even if a plaintiff is successful in court, enforcement of court-ordered fines or collection of damages can be difficult. Spammers can also evade cease-and-desist orders because they can move their operations from one server to another in minutes. Many spammers forward mail through servers that they have hijacked (you will learn more about threats to servers in Chapter 10).

Although laws are not likely to stop the most determined spammers, some industry observers hope that laws such as CAN-SPAM will at least enforce constraints on the legitimate marketers that send commercial e-mails. In January 2005, *PC World* conducted a study to determine whether e-mail marketers were complying with the law. The researchers signed up for 100 marketing information mailing lists, then tried to opt out of the mailings. Of the 100 lists, seven sent messages that did not include a valid postal address and two more did not include an opt-out mechanism. When the researchers opted out, 85 of the companies complied fully with the CAN-SPAM law provisions, but eight sent e-mails beyond the 10-day limit and another four had marketing partners that continued to send e-mails. One list sent messages that included an opt-out link that did not work.

Some critics argue that any legal solution to the spam problem is likely to fail until the prosecution of spammers becomes cost-effective for governments. To become cost-effective, prosecutors must be able to identify spammers easily (to reduce the cost of bringing an action against them) and must have a greater likelihood of winning the cases they file (or must see a greater social benefit to winning). The best way to make spammers easier to find is to make technical changes in the e-mail transport mechanism in the Internet's infrastructure.

Technical Solutions

The Internet was not designed to do many of the things it does today. It was not designed to be secure, to process transactions, or to handle billions of e-mail messages. As you learned in Chapter 2, Internet e-mail was an incidental afterthought in a system designed to transfer large files from one researcher to another. As it was originally designed, and as

it operates today, the Internet did not include any mechanisms for ensuring that the identity of an e-mail sender would always be known to the e-mail's recipient.

At least one technical strategy for fighting spam exploits a weakness in the original design of the Internet. The Internet protocol that governs communication among servers on the Internet (including e-mail servers), was designed to be a polite set of rules. When one computer on the Internet sends a message to another computer, it will wait to receive an acknowledgment that the message has been received before sending more messages. In the ordinary course of Internet communications, the acknowledgment messages come back in far less than a second. If a computer is set to send the acknowledgment back more slowly, the originating computer will slow down because it must continue to scan for the acknowledgment (which consumes some of its processing power) and it will not send any more messages to that address until it does receive the acknowledgement.

To use this characteristic of the Internet messaging rules to counter spam requires that the defending company develop a way to identify computers that are sending spam. Some vendors, such as IBM, sell software and access to a large database that tracks such computers continually. Other vendors sell software that identifies multiple e-mail messages coming from a single source in rapid succession (as would happen if a spammer were sending spam to everyone at a particular company). Once the spamming computer is identified, the software delays sending the message acknowledgements. It can also launch a return attack, sending e-mail messages back to the computer that originated the suspected spam. This practice is called **teergrubing**, which is from the German word for "tar pit." The objective is to ensnare the spam sending computer in a trap that drags down its ability to send spam. Although many organizations use teergrubing as part of their spam defense strategy, some are concerned that launching a counterattack might violate laws that were enacted, ironically enough, to punish spammers.

Most industry observers agree that the ultimate solution to the spam problem will come when new e-mail protocols are adopted that provide absolute verification of the source of each e-mail message. This will require all mail servers on the Internet to be upgraded. The new protocols have not yet been written, so this solution is several years away.

Proposals for identification standards have been made by Time Warner's AOL division, Microsoft, Yahoo!, and other companies and organizations. The Internet Engineering Task Force (IETF) working group that has responsibility for e-mail standards has rejected some of these proposals, but has stated its commitment to working out a set of standards that will accomplish sender authentication. You can learn more about current developments on this issue by following the links in the Online Companion in the Additional Information section under the heading **Spam Information Sites**.

WEB SITE AND INTERNET UTILITY PROGRAMS

In addition to Web server software, people who develop Web sites work with a number of utility programs, or tools. TCP/IP supports a wide variety of these utility programs. Some of these programs run on the Web server itself, while others run on the client computers that Web developers use when they are creating Web sites. E-mail was one of the earliest

Internet utility programs and it has become one of the most important. In earlier chapters, you learned how companies are using e-mail as a key element in their electronic commerce strategies. In this section, you will learn some of the more significant technical details of how e-mail works. You will learn about several of these programs and see examples of how they work.

Finger and Ping Utilities

Finger is a program that runs on UNIX operating systems and allows a user to obtain some information about other network users. A Finger command yields a list of users who are logged on to a network, or reports the last time a user logged on to the network. Many organizations have disabled the Finger command on their systems for privacy and security reasons. For example, if you send a Finger command to a server at www.microsoft.com, you receive no response. Some e-mail programs have the Finger program built into them, so you can send the command while reading your e-mail.

A program called **Ping**, short for **Packet Internet Groper**, tests the connectivity between two computers connected to the Internet. Ping provides performance data about the connection between Internet computers, such as the number of computers (hops) between them. It sends two packets to the specified address and waits for a reply. Network technicians often use Ping to troubleshoot Internet connections. Many freeware and shareware Ping programs are available on the Internet. You can send out a Ping from a Windows PC by opening an MS-DOS window and typing "ping" followed by the IP address.

Tracert and Other Route-Tracing Programs

Tracert (TRACE RouTe) sends data packets to every computer on the path (Internet) between one computer and another computer and clocks the packets' round-trip times. This provides an indication of the time it takes a message to travel from one computer to another and back, ensures that the remote computer is online, and pinpoints any data traffic congestion. Route-tracing programs also calculate and display the number of hops between computers and the time it takes to traverse the entire one-way path between machines.

Route-tracing programs such as Tracert work by sending a series of packets to a particular destination. Each router along the Internet path between the originating computer and the destination computer reports its IP address and the time it took to reach it. After the program completes its packet transmissions, it displays the number of hops and how much time it took to reach each node and travel the entire path.

Graphical user interface route-tracing programs provide a plot of the packets' route on a map. Network engineers can use route-tracing programs to determine the location of the greatest delays on the Internet. Companies that provide Internet connections to customers often run route-tracing programs to monitor and improve services. Visualware offers its **VisualRoute** route-tracing program for download, trial, and purchase. The site also offers a demonstration of VisualRoute that runs on its Web site so potential customers can test the program without downloading any software. Figure 8-9 shows a route traced from a VisualRoute server in Englewood, Colorado (USA), to a server at SingTel in Milton, Australia, using the VisualRoute program.

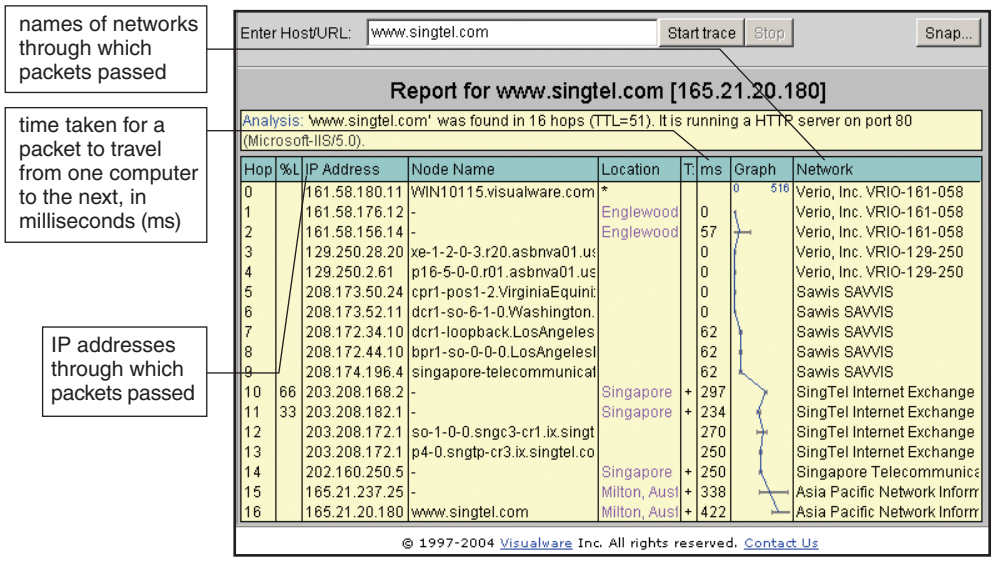


FIGURE 8-9 Tracing a path between two computers on the Internet

You can see that the packets traveled through 16 computers (that is, the path included 16 hops) and that the path went from Colorado to Virginia to Washington, D.C. to Los Angeles, to Singapore, to Australia. The test message took 422 milliseconds (ms), which is just under one-half of a second, to travel more than halfway around the world.

Telnet and FTP Utilities

Telnet is a program that allows users to log on to a computer that is connected to the Internet. This remote login capability can be useful for running older software that does not have a Web interface. Several Telnet client programs are available as free downloads on the Internet, and Microsoft Windows systems include a Telnet client called Telnet.exe. Telnet lets a client computer give commands to programs running on a remote host. Telnet programs use a set of rules called the **Telnet protocol**. Some Web browsers function as a Telnet client. A user can enter "telnet://" followed by the domain name of the remote host. As more companies place information on Web pages, which are accessible through any Web browser, the use of Telnet will continue to decrease.

The **File Transfer Protocol (FTP)** is the part of the TCP/IP rules that defines the formats used to transfer files between TCP/IP-connected computers. FTP can transfer files one at a time, or it can transfer many files at once. FTP also provides other useful services, such as displaying remote and local computers' directories, changing the current client's or server's active directory, and creating and removing local and remote directories. FTP uses TCP and its built-in error controls to copy files accurately from one computer to another.

Accessing a remote computer with FTP requires that the user log on to the remote computer. A number of FTP client programs exist; however, many people just use their Web browser software. Typing the protocol name, ftp://, before the domain name of the remote

computer establishes an FTP connection. Users who have accounts on remote computers can log on to their accounts using the FTP client. FTP establishes contact with the remote computer and logs onto the account on that computer.

An FTP connection to a computer on which the user has an account is called **full-privilege FTP**. Another way to access a remote computer is called anonymous FTP. **Anonymous FTP** allows the user to log on as a guest. By entering the username "anonymous" and an e-mail address as a password, users can read and copy files that are on the remote computer.

Indexing and Searching Utility Programs

Search engines and indexing programs are important elements of many Web servers. Search engines or search tools search either a specific site or the entire Web for requested documents. An indexing program can provide full-text indexing that generates an index for all documents stored on the server. When a browser requests a Web site search, the search engine compares the index terms to the requester's search term to see which documents contain matches for the requested term or terms. More advanced search engine software (such as that used by the popular search engine site Google) uses complex relevance ranking rules that consider things such as how many other Web sites link to the target site. Many Web server software products also contain indexing software. Indexing software can often index documents stored in many different file formats.

Data Analysis Software

Web servers can capture visitor information, including data about who is visiting a Web site (the visitor's URL), how long the visitor's Web browser viewed the site, the date and time of each visit, and which pages the visitor viewed. This data is placed into a Web **log file**. As you can imagine, the file grows very quickly—especially for popular sites with thousands of visitors each day. Careful analysis of the log file can be fruitful and reveal many interesting facts about site visitors and their preferences. To make sense of a log file, you must run third-party Web log file analysis programs. These programs summarize log file information by querying the log file and either returning gross summary information, or accumulating details that reveal how many visitors came to the site per day, hour, or minute, or which hours of the day were peak loading times. Popular Web log file analysis programs include products by **Analog**, **Keylime Software**, **Urchin Web Analytics**, **Web Side Story**, and **WebTrends**.

Link-Checking Utilities

Dedicated site management tools include a standard set of features, starting with link checking. A **link checker** examines each page on the site and reports on any URLs that are broken, seem broken, or are in some way incorrect. It can also identify orphan files. An **orphan file** is a file on the Web site that is not linked to any page. Other important site management features include script checking and HTML validation. Some management tools can locate error-prone pages and code, list broken links, and e-mail maintenance results to site managers.

On the company Web site, it is important to regularly check links that point to pages both within and outside the corporate Web site. Some Web server software does contain

link-checking features. A **dead link**, when clicked, displays an error message rather than a Web page. Maintaining a site that is free of dead links is vital because visitors who encounter too many dead links on a site might jump to another site. Web-browsing customers are just a click away from going to a competitor's site if they become annoyed with an errant Web link.

Some Web site development and maintenance tools, such as Macromedia's Dreamweaver, include link-checking features. Most link-checking programs, however, run as separate programs. One of these link-checking programs, **Elsop LinkScan**, is available in a demo version as a free download. The results of the link checker either appear in a Web browser or are e-mailed to a recipient. Besides checking links, Web site validation programs sometimes check spelling and other structural components of Web pages.

LinxCop is one of several reverse link checkers available. A **reverse link checker** checks on sites with which a company has entered a link exchange program (which you learned about in Chapter 4) and ensures that link exchange partners are fulfilling their obligation to include a link back to the company's Web site.

Remote Server Administration

With **remote server administration**, a Web site administrator can control a Web site from any Internet-connected computer. Although all Web sites provide administrative controls—most through a workstation computer on the same network as the server computer or through a Web browser—it is convenient for an administrator to be able to fix the server from wherever he or she happens to be. For example, an administrator can install **Web Site Garage** on any Internet-connected Windows computer and monitor and change anything on the Web site from that computer. **NetMechanic** offers a variety of link-checking, HTML troubleshooting, site-monitoring, and other programs that can be useful in managing the operation of a Web site.

WEB SERVER HARDWARE

Now that you have learned about Web server and Internet utility software, your next step is to learn about Web server hardware. Companies use a wide variety of computer brands, types, and sizes to host electronic commerce operations. Some small companies can run Web sites on desktop PCs. Most electronic commerce Web sites are operated on computers designed for site hosting, however.

Server Computers

Web server computers generally have more memory, larger (and faster) hard disk drives, and faster processors than the typical desktop or notebook PCs with which you are probably familiar. Many Web server computers use multiple processors; very few desktop PCs have more than one processor. Because Web server computers use more capable hardware elements and more of these elements, they are usually much more expensive than workstation PCs. Today, a high-end desktop PC with 2 GB of RAM, a 3.6 GHz processor, a fast 400 GB SATA disk drive, a good monitor, and a complement of DVD/CD-RW drives

costs between $2000 and $4000. A company might be able to buy a low-end Web server computer for about the same amount of money, but most companies spend between $6000 and $400,000 on a Web server. Companies that sell Web server hardware, such as **Dell**, **Gateway**, **Hewlett Packard**, and **Sun**, all have configuration tools on their Web sites that allow visitors to design their own Web servers. Figure 8-10 shows three typical midrange server computer configurations available for sale on Sun's Web site.

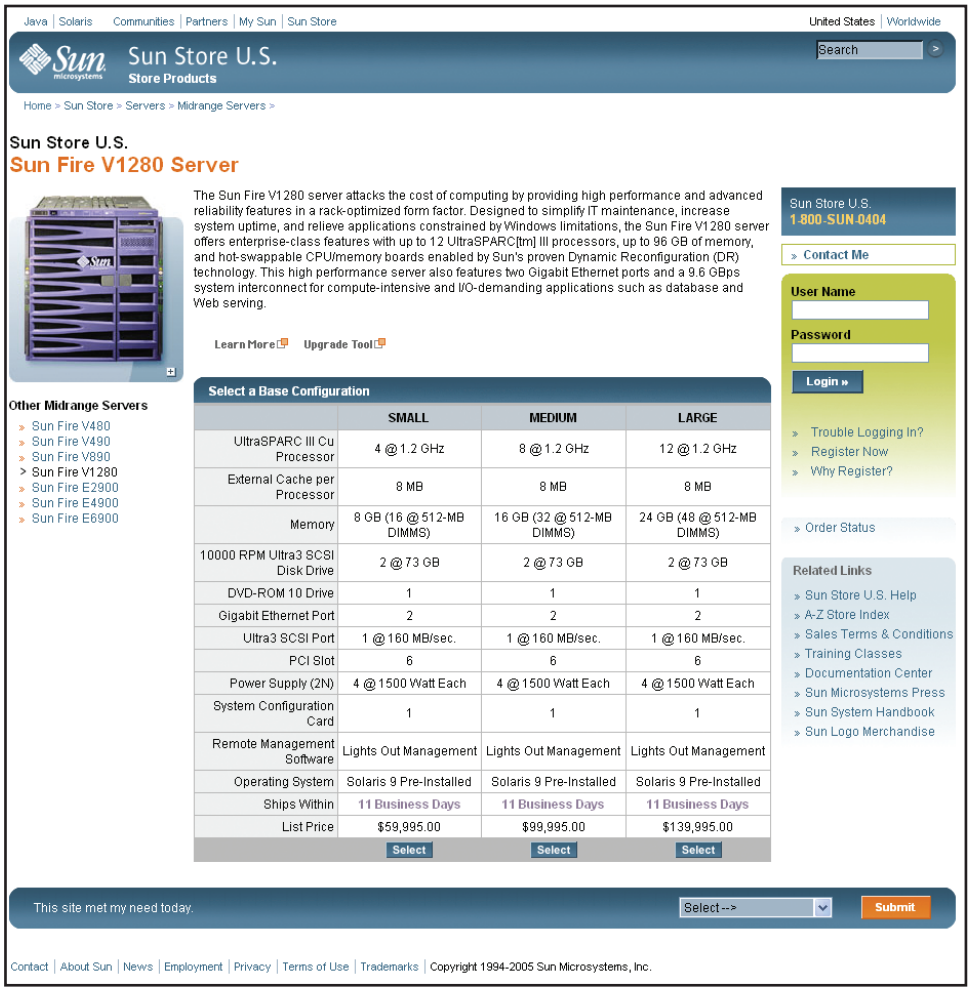


FIGURE 8-10 Three typical midrange Sun server computer configurations

Although some Web server computers are housed in freestanding cases, most are installed in equipment racks. These racks are usually about 6 feet tall and 19 inches wide. They can each hold several midrange server computers. A recent innovation in server computer design is to put small server computers on a single computer board and then install many of those

boards into a rack-mounted frame. These servers-on-a-card are called **blade servers**, and some manufacturers now make them so small that more than 300 of them can be installed in a single 6-foot rack.

Recall that the fundamental job of a Web server is to process and respond to Web client requests that are sent using HTTP. For a client request for a Web page, the server program finds and retrieves the page, creates an HTTP header, and appends the HTML document to it. For dynamic pages, the server uses an architecture with three or more tiers that uses other programs, receives the results from the back-end process, formats the response, and sends the pages and other objects to the requesting client program. IP-sharing, or a virtual server, is a feature that allows different groups to share a single Web server's IP address. A **virtual server** or **virtual host** is a feature that maintains more than one server on one machine. This means that different groups can have separate domain names, but all domain names refer to the same physical Web server.

Web Server Performance Evaluation

Benchmarking Web server hardware and software combinations can help in making informed decisions for a system. **Benchmarking**, in this context, is testing that is used to compare the performance of hardware and software.

Elements affecting overall server performance include hardware, operating system software, server software, connection speed, user capacity, and type of Web pages being delivered. When evaluating Web server performance, a company should know exactly what factors are being measured and ensure that these are important factors relative to the expected use of the Web server. Another factor that can affect a Web server's performance is the speed of its connection. A server on a T3 connection can deliver Web pages to clients much faster than on a T1 connection.

The number of users the server can handle is also important. This can be difficult to measure because results are affected by the server's line speed, the clients' line speeds, and the sizes of the Web pages delivered. Two factors to evaluate when measuring a server's Web page delivery capability are throughput and response time. **Throughput** is the number of HTTP requests that a particular hardware and software combination can process in a unit of time. **Response time** is the amount of time a server requires to process one request. These values should be well within the anticipated loads a server can experience, even during peak load times.

One way to choose Web server hardware configurations is to run tests on various combinations, remembering to consider the system's scalability. Of course, you need to have the hardware and software set up to do this, so it is difficult to evaluate potential configurations that you have not yet purchased. Independent testing labs such as **Mindcraft** test software, hardware systems, and network products for users. Its site contains reports and statistics comparing combinations of application server platforms, operating systems, and Web server software products.

Anyone contemplating purchasing a server that will handle heavy traffic should compare standard benchmarks for a variety of hardware and software configurations. Customized benchmarks can give Web managers guidelines for modifying file sizes, cache sizes, and other parameters. Web managers should run benchmarks regularly. Benchmarks are not as meaningful for small Web sites with much smaller numbers of daily visitors. In the latter case, a focus on Web design and site navigation can maximize clients' satisfaction.

Companies that operate more than one Web server must decide how to configure servers to provide site visitors with the best service possible. The different ways that servers can be connected to each other and to related hardware, such as routers and switches, are called **server architectures**.

Web Server Hardware Architectures

Earlier in this chapter, you learned that electronic commerce Web sites can use two-tier, three-tier, or n-tier architectures to divide the work of serving Web pages, administering databases, and processing transactions. Some electronic commerce sites are so large that more than one computer is required within each tier. For example, large electronic commerce Web sites must deliver millions of individual Web pages and process thousands of customer and vendor transactions each day.

Administrators of these large Web sites must plan carefully to configure their Web server computers, which can number in the hundreds or even thousands, to handle the daily Web traffic efficiently. These large collections of servers are called **server farms** because the servers are often lined up in large rooms, row after row, like crops in a field. One approach, sometimes called a **centralized architecture**, is to use a few very large and fast computers. A second approach is to use a large number of less powerful computers and divide the workload among them. This is sometimes called a **distributed architecture** or, more commonly, a **decentralized architecture**.

Each approach has benefits and drawbacks. The centralized approach requires expensive computers and is more sensitive to the effects of technical problems. If one of the few servers becomes inoperable, a large portion of the site's capability is lost. Thus, Web sites with centralized architectures must have adequate backup plans. Any server problem, no matter how small, can threaten the operation of the site. The decentralized architecture spreads that risk over a large number of servers. If one server becomes inoperable, the site can continue to operate without much degradation in capability. The smaller servers used in the decentralized architecture are less expensive than the large servers used in the centralized approach. That is, the total cost of 100 small servers is usually less than the cost of one large server with the same capacity as the 100 small servers. However, the decentralized architecture does require additional hubs or switches to connect the servers to each other and to the Internet. Most large decentralized sites use load-balancing systems, which do cost additional money, to assign the workload efficiently.

LEARNING FROM FAILURES

WEB SERVERS AT EBAY

Online auction site **eBay** is very popular, as you have learned in earlier chapters. Indeed, it is so popular that its Web servers deliver hundreds of millions of pages per day. These pages are a combination of static HTML pages and dynamically generated Web pages. The dynamic pages are created from queries run against eBay's Oracle database, in which it keeps all of the information about all auctions that are under way or have closed within the most recent 30 days. With millions of auctions under way at any moment, this database is extremely large. The combination of a large database and high transaction volume makes eBay's Web server operation an important part of the company's success and a potential contributor to its failure. The servers at eBay failed more than 15 times during the first five years (1995–2000) of the company's life. The worst series of failures occurred during May and June of 2000, when the site went down four times. One of these failures kept the site offline for more than a day—a failure that cost eBay an estimated $5 million. The company's stock fell 20 percent in the days following that failure.

At that point, eBay decided it needed to make major changes in its approach to Web server configuration. Many of eBay's original technology staff had backgrounds at Oracle, a company that has a tradition of selling large databases that run on equally large servers. Further, the nature of eBay's business—any visitor might want to view information about any auction at any time—led eBay management initially to implement a centralized architecture with one large database residing on a few large database server computers. It made sense also to use similar hardware to serve the Web pages generated from that database.

In mid-2000, following the worst site failure in its history, eBay decided to move to a decentralized architecture. This was a tremendous challenge because it meant that the single large auction information database had to be replicated across groups, or clusters, of Web and database servers. However, eBay realized that using just a few large servers had made it too vulnerable to the failure of those machines. Once eBay completed the move to decentralization, it found that adding more capacity was easier. Instead of installing and configuring a large server that might have represented 15 percent or more of the site's total capacity, clusters of six or seven smaller machines could be added that represented less than one percent of the site's capacity. Routine periodic maintenance on the servers also became easier to schedule.

The lesson from eBay's Web server troubles is that the architecture should be carefully chosen to meet the needs of the site. Web server architecture choices can have a significant effect on the stability, reliability, and, ultimately, the profitability of an electronic commerce Web site.

Load-Balancing Systems

A **load-balancing switch** is a piece of network hardware that monitors the workloads of servers attached to it and assigns incoming Web traffic to the server that has the most available capacity at that instant in time. In a simple load-balancing system, the traffic that enters the site from the Internet through the site's router encounters the load-balancing switch, which then directs the traffic to the Web server best able to handle the traffic. Figure 8-11 shows a basic load-balancing system.

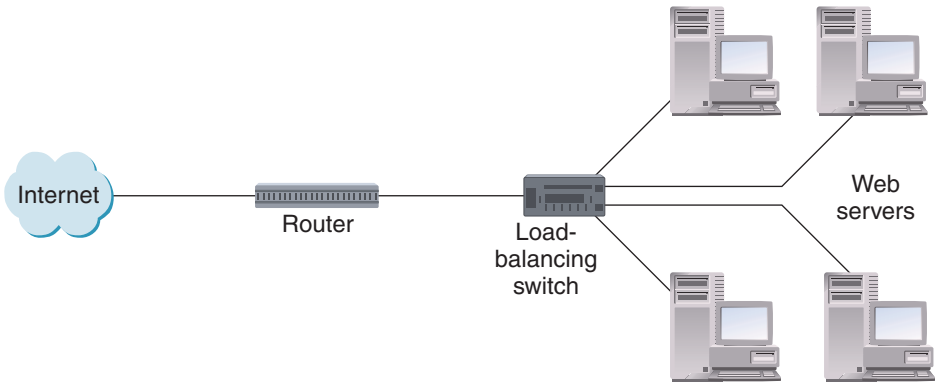


FIGURE 8-11 A load-balancing system in a decentralized architecture

In more complex load-balancing systems, the incoming Web traffic, which may enter from two or more routers on a larger Web site, is directed to groups of Web servers dedicated to specific tasks. In the sample complex load-balancing system that appears in Figure 8-12, the Web servers have been gathered into groups of servers that handle delivery of static HTML pages, servers that coordinate queries of an information database, servers that generate dynamic Web pages, and servers that handle transactions. Load-balancing switches and the software that helps them do their work cost roughly between $5,000 and $50,000, and include products such as **E-Load**, **Loadrunner**, **ServerIron**, and **Silkperformer**.

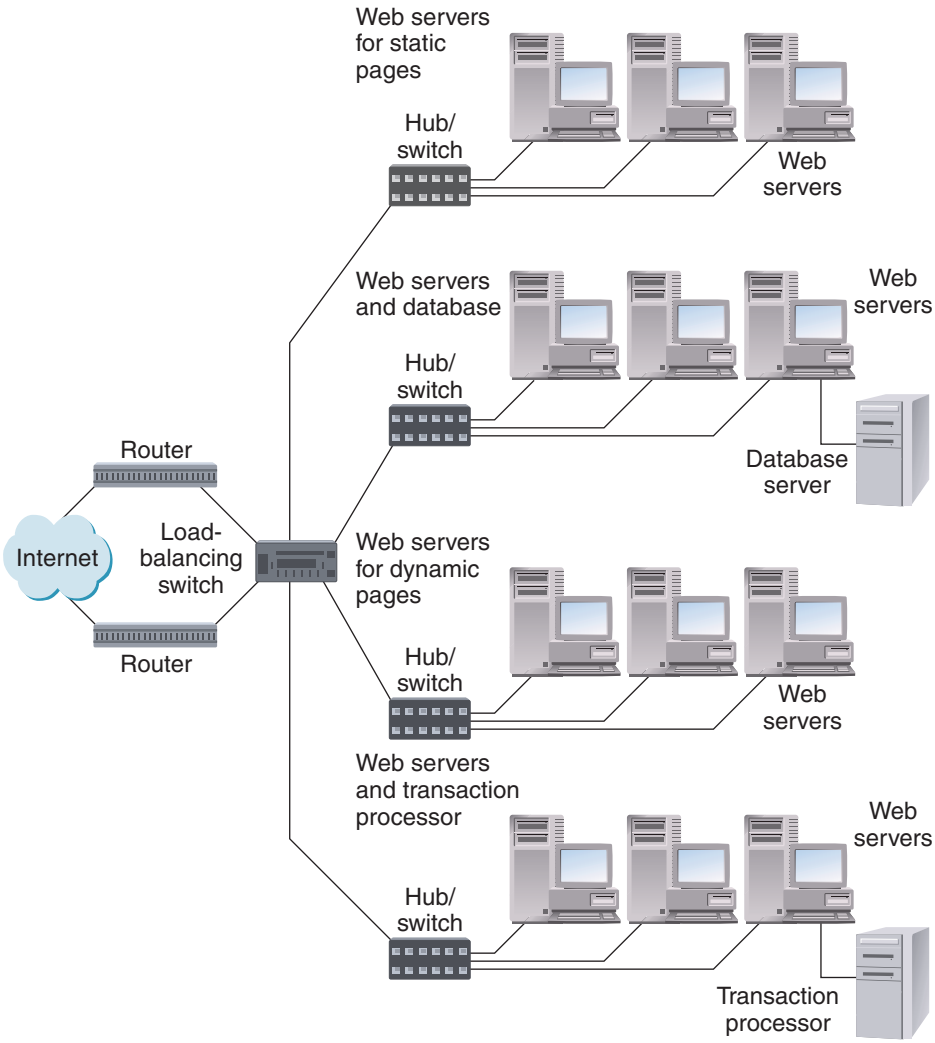


FIGURE 8-12 Complex load balancing

Summary

The Web uses a client/server architecture in which the client computer requests a Web page and a server computer that is hosting the requested page locates and sends a page back to the client. For simple HTTP requests, a two-tier architecture works well. The first tier is the client computer and the second tier is the server. More complicated Web interactions, such as electronic commerce, require the integration of databases and payment-processing software in a three-tier or higher (n-tier) architecture.

Operating systems commonly used on Web server computers include Microsoft server operating systems; UNIX-based operating systems; such as SunOS, FreeBSD; and Linux. The most popular Web server programs are Apache HTTP Server, Microsoft Internet Information Server, and Sun Java System Web Server. Web server computers also run a variety of utility programs such as Finger, Ping, Tracert, e-mail server software, Telnet, and FTP. Most Web servers also have software that helps with link checking and remote server administration tasks.

The problem of unsolicited commercial e-mail (spam) has grown dramatically in recent years. Content filters, particularly naïve Bayesian filters, are becoming available to deal with the problem. An increasing number of organizations are using a combination of server-level filters and client-level filters to reduce spam to tolerable levels. New laws designed to punish spammers have not stemmed the tide of spam. Some organizations are using counterattack strategies to impede spammers' ability to send large numbers of messages, but most industry experts believe spam will persist until new e-mail protocols are adopted that provide absolute authentication of e-mail senders' identities.

Web server hardware is also an important consideration in the design of an online business site. The server computer must have enough memory to serve Web pages to all site visitors and enough disk space to store the Web pages and the databases that store the elements of dynamically generated Web pages. Large Web sites that have many Web server computers use load-balancing hardware and software to manage their high-activity volumes.

The operating system, connection speed, user capacity, and the type of pages that the site serves affect overall Web server performance. Benchmarking software and consulting firms that use it can help companies evaluate specific combinations of Web server hardware, software, and operating systems.

Key Terms

Active Server Pages (ASP)
Anonymous FTP
Attachment
Bayesian revision
Benchmarking
Black list spam filter
Blade server
Centralized architecture
Challenge-response
Client-level filtering
Computer virus
Database server
Dead link
Decentralized architecture
Distributed architecture
Dynamic content
Dynamic page
Entity body
False positive
File Transfer Protocol (FTP)

Finger
Full-privilege FTP
JavaServer Pages (JSP)
Java servlet
Link checker
Load-balancing switch
Log file
Naïve Bayesian filter
National Center for Supercomputing Applications (NCSA)
N-tier architecture
Open-source software
Orphan file
PHP: Hypertext Preprocessor (PHP)
Ping (Packet Internet Groper)
Remote server administration
Request header
Request line
Request message
Response header field
Response header line
Response message
Response time
Reverse link checker
Server
Server architecture
Server farm
Server-level filtering
Server-side scripting (server-side includes or server-side technologies)
Server software
Spam (unsolicited commercial e-mail or bulk mail)
Static page
Teergrubing
Telnet
Telnet protocol
Three-tier architecture
Throughput
Tracert
Two-tier client/server architecture
Virtual host
Virtual server
Virus
White list spam filter

Review Questions

RQ1. Compare the two- and three-tier Web client/server architectures and indicate the role of each tier in each architecture. Which architecture is the most likely candidate for an electronic commerce site? Explain why in about 100 words.

RQ2. Describe and briefly discuss two important measures of a Web site's performance.

RQ3. Beginning with the links provided in the Online Companion, locate more information about two of the three Web servers discussed in the chapter: Apache, Microsoft IIS, and Sun JSWS. Write approximately 250 words about each of the two servers you choose. Include descriptions of six features for each Web server and indicate the computer platforms and operating systems on which each runs.

RQ4. In one paragraph, outline the main differences between a desktop PC and a computer that would be suitable as a Web server for a small Web site.

RQ5. Using the Web or your school library, find articles that discuss the types of server hardware used by at least one electronic commerce site. Outline that site's architecture and approach to server hardware and software in an essay of about 300 words.

Exercises

E1. You are the information systems director for Abbon Laboratories, a biotechnology research firm with about 100 employees. Alice Stampler, Abbon's president, is aware that Abbon's incoming e-mail includes a great deal of spam and has always complained about the time it takes her to delete it. More importantly, she is concerned about the time wasted by the company's employees. Just thinking about those expensive Ph.D. researchers spending time deleting half of their e-mail drives her to distraction. Alice just read a story in a business magazine about naïve Bayesian spam filters. Possessing a Ph.D. herself, she is fascinated by the prospect that sophisticated mathematics might solve the company's spam problem. Alice asks you to find out all you can about naïve Bayesian filters and present a short report (about 200 words) in which you evaluate the technique and whether it can work for Abbon. Alice envisions one filter installed on the e-mail server that would screen all e-mail as it enters the company's network. You can use your library, your favorite search engines, or the links in the Online Companion under the heading **Naïve Bayesian Filters** to do your research.

E2. You created a Web site for International Paper Products and Pulp, complete with links to other pages on your site and to pages on the Internet. Bob Pardee, your supervisor, wants you to check periodically that the links on the corporate site are still valid. Instead of purchasing and installing a link-checking program, you decide to investigate online link checkers (Web sites that allow you to enter a Web site's root or home address and then check all the links that emanate from that site). Use **Link Check** or **Elsop LinkScan Quick Check** to check the links on any site of your choice. Print a few pages of the report and be prepared to turn them in to your instructor. Be patient. The program can take some time to complete its work—especially on a Web page that has a large number of links.

E3. In researching Web server computers, you find that many companies that sell these computers offer a configuration option for controlling computers' disk drives called "RAID." Using the Web and your library, investigate the purpose of RAID controllers. Learn what these controllers do and how they do it. Summarize your findings in a 600-word briefing report suitable for presentation to a nontechnical manager.

Cases

C1. Microsoft and the People's Republic of China

Software piracy has been a major challenge for software makers such as Microsoft that want to sell software in the global marketplace. Laws that protect intellectual property vary from country to country, and the laws in many countries provide little or no protection. Governments in developing countries are reluctant to increase the protections afforded by their intellectual property laws because they see no point in passing laws that protect the profits of foreign corporations by imposing higher costs on their struggling local businesses and citizens. In the late 1990s, after years of holding firm on its global pricing, Microsoft began to offer significant discounts on its software to governments, small businesses, and individuals in developing countries. It also provided discounts on Windows operating systems software that was installed in new personal computers manufactured in developing countries. Microsoft donated software licenses to

schools in developing countries. Just as these efforts were beginning to show some results, however, Microsoft faced a new threat to its global market position—open source software.

Open source operating system software, such as Linux, gives governments and businesses in developing countries a way to avoid paying any server software licensing fees to Microsoft. In 2000, the Brazilian state of Penambuco became the first governmental entity to pass a law that requires the use of open source software on all computers used for state business. Shortly thereafter, the Brazilian state of Rio Grande do Sul passed a similar law that requires the use of open source software in all of the state's offices and in all privately operated utilities. In 2003, IBM realized the potential for open source consulting business in the country and opened several centers for the development of Linux-based application software in Brazil. Microsoft, concerned about a Latin American open source domino effect, embarked on a public relations campaign in the region that included increased advertising spending and donations to public schools. In 2002, Peru was considering passing a law that would require public schools to use open source software. Microsoft founder Bill Gates flew to Peru and, with great public fanfare, donated $550,000 to the schools that would have been affected by the legislation. In 2004, Microsoft announced that it would donate $1 billion in cash and software over five years through the United Nations Development Program to not-for-profit organizations in 45 countries.

Most industry observers believe that Microsoft's largest non-U.S. market today is the People's Republic of China (PRC). Although the PRC generates about $300 million in licensing revenue for Microsoft, more than 90 percent of all Microsoft products used in China today are pirated. Bootleg copies of the company's latest products can be purchased on the street for a few dollars. Thus, Microsoft believes that converting users to paid licenses could increase its PRC licensing revenues tremendously. As the PRC moves from being a less-developed country toward becoming a major economic power in the world, Microsoft sees an opportunity to increase its licensing revenue in the country. In the past, Microsoft has used a global anti-piracy strategy that relied on identifying users of pirated software and threatening those users with legal action, but the company is changing its approach in developing markets such as Latin America and Asia. In the PRC, Microsoft's near-term goal is to develop a market for full-price software licenses that includes large business and government customers. Its new approach focuses more on recruiting major PRC business organizations as customers and less on sending threatening letters to users of pirated Microsoft software.

In developing its business in the PRC, Microsoft faces a number of challenges. Juliet Wu, former general manager of Microsoft China, published a book in 2000 that was highly critical of the company. The book was widely read in the PRC and received many good reviews. PRC officials have often criticized Microsoft for many things ranging from high prices to the company's use of Taiwanese programmers (the PRC does not officially recognize Taiwan as an independent nation separate from the PRC). Government officials in the PRC are also concerned about security. Microsoft has always maintained that the code to its software products is a trade secret and has refused to allow its publication or distribution. Companies that develop software that runs on Microsoft Windows, for example, must sign a non-disclosure agreement with Microsoft to obtain information they need about how Windows operates so they can make their software compatible with it. Many PRC officials believe that Microsoft, as a U.S.-based company, might include secret back doors into its software that would allow the U.S. government to enter PRC government computers undetected in a time of international conflict or war. At a very basic level, the ideology of the PRC's socialist government is a polar opposite to the highly competitive capitalist

principles that have driven Microsoft to success. But the greatest challenge that Microsoft must overcome in the PRC is the attraction of open source software.

Open source programs' code is public; thus it cannot have secret back doors that can remain undetected in the code. The PRC is training 600,000 engineers each year and is expected to soon become the second-largest semiconductor manufacturer in the world. These semiconductors are increasingly being used to manufacture Chinese computers, many of which ship with a Linux operating system. In 1999, PRC officials announced a Chinese version of Linux called Red Flag. The PRC's national lottery and post office system all use Linux operating systems. The Province of Guangdong's accounting system runs on Linux-based computers. IBM is the primary consultant on a multi-year project to put the PRC's social security system onto Linux-based systems. As programming jobs in the United States dry up, some PRC Linux software developers have even been able to hire back Chinese programmers who had gone to U.S. companies. In 2003, the Procurement Center of the State Council issued an edict requiring that any computer purchased by the government after 2004 must be delivered with PRC-produced software only.

In the face of these challenges, Microsoft has worked hard to deliver its message that open source software can result in higher total costs because even though it is free, it requires more effort to install, maintain, and update than Microsoft products. In large organizations, this effort results in extra hours worked and thus, extra costs. Microsoft also argues that open source software's publicly available program code makes it a greater security risk. According to Microsoft, attackers can easily learn how any open source program works and develop strategies for attacking the software when it is running on publicly accessible computers, such as Web servers.

Required:

1. Assume that you are on the staff of a PRC legislator. Outline the arguments that you would use to support a law that required all government agencies to use only open source software on their Web servers.
2. Assume that you are working for the marketing department of Microsoft China. Develop a detailed list of briefing points that would help your salespeople convince top executives of large PRC companies to use Windows operating system software on their Web servers.
3. Assume that you are working for the business systems analysis department in IBM's PRC division, which offers both Microsoft Windows and Linux consulting services to PRC businesses and government offices. Develop a checklist that IBM analysts could use in consulting projects that could help advise clients as they make a choice between Windows or Linux operating system software for their Web servers.
4. Companies such as RedHat, Novell (with its SuSE distribution), and SCO (with its Caldera distribution) offer Linux operating system software for sale. Although Linux is available at no cost from various sources, these companies charge a fee for installation and configuration help. They also offer service contracts to help users maintain and upgrade the software on a continuing basis. Briefly outline the strategies that these companies might use to expand their market share in the PRC.

Note: Your instructor might assign you to a group to complete this case, and might ask you to prepare a formal presentation of your results to your class.

C2. Random Walk Shoes

Amy Lawrence, the owner of Random Walk Shoes, has asked you to help her as she launches her company's first Web site. In college, Amy was a business major with an artistic bent. She helped to pay her way through college by decorating sneakers with her hand-painted designs. Her business grew through word of mouth and through her participation in crafts fairs. By the time she earned her degree, Amy was running a successful business from her dorm room.

Amy expanded her sales efforts to include crafts fairs in nearby towns. She hired two college students to work for her, and she convinced several area gift shops to stock samples of her merchandise. The gift shops were not an ideal retail outlet for her products, however. Most people who want to buy decorated sneakers want to choose specific designs or have special designs created just for them. Customers also want to choose the specific shoes on which the design is placed. One of Amy's student workers suggested that she consider selling her products on the Web.

Realizing that the Web would give Random Walk Shoes a chance to reach a much wider audience and would allow customers to choose design-shoe combinations, Amy began gathering information and developing estimates about her planned Web activity. She bought a digital camera and took several hundred pictures of shoes, designs, and shoe-design combinations. She then hired a local Web designer to create sample pages for the Web site, including catalog pages that contained the digital images.

When the Web designer had completed a prototype of the site, Amy worked with the designer to calculate page sizes (including the images). The average page size was 85 KB. Amy and her employees then navigated the prototype site several hundred times to develop an estimate of how many pages an average visitor would download. They concluded that an average site visitor would visit 72 pages during each visit. Amy worked with the Web designer to develop estimates of activity they expect to occur on the Web site during its first two years of operation. These estimates include:

- The database of Web page information (including the images) will require about 80 MB of disk space.
- The database management software itself will require about 300 MB of disk space.
- The shopping cart software will require about 50 MB of disk space.
- About 6000 customers will visit the site during the first month, and site traffic will grow about 20 percent each month during the first two years.
- The site should accommodate a peak traffic load of 1000 visitors at one time.

Amy wants to include features on the site that are similar to those found on competing sites (a list of links to businesses that sell customized shoes on the Web is included in the Online Companion for your reference). Amy wants the site to provide a good experience for visitors. If the site is successful, it will generate sufficient revenue to allow an upgrade after two years. However, she does not want to spend more money than is necessary to get the site up and keep it running for the next two years.

Required:

1. Determine the features and capacities (RAM, disk storage, processor speed) that Amy should include in the Web server computer she will need for her site. Summarize your purchase recommendation in a one-page memorandum to Amy. You may include information from vendors' sites (such as **Dell**, **Hewlett Packard**, or **Sun**) as an appendix to your memorandum.
2. Consider the advantages and disadvantages of each major operating system that Amy might use on the new Web server computer. In a one-page memorandum to Amy, make a specific recommendation and support it with facts and a logical argument. If you do not believe that one operating system is clearly superior for this application, explain why.

3. Consider the advantages and disadvantages of each major Web server software package for accomplishing the goals that Amy has for this site. In a one-page memorandum to Amy, make a specific recommendation regarding which Web server software package she should use. Provide an explanation that supports your recommendation.

Note: Your instructor might assign you to a group to complete this case, and might ask you to prepare a formal presentation of your results to your class.

For Further Study and Research

Abualsamid, A. 2001. "Dishing Up Dynamic Content," *Network Computing*, 12(8), April 16, 90–92.

Alwang, G. 1998. "Internet Web Servers," *PC Magazine*, 17(9), May 5, 184–208.

Andrews, P. 2003. "Courting China," *U.S. News & World Report*, November 24, 44–45.

Ante, S. 2001. "Big Blue's Big Bet on Free Software," *Business Week*, December 10, 78–79.

Baran, N. 2001. "Load Testing Web Sites," *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, 26(3), March, 112–116.

Business Week Online. 2004. "China and Linux: Microsoft, Beware!" November 15. (http://www.businessweekonline.com)

Chabrow, E. 2005. "In The Fight Against Spam, A Few Knockouts: Microsoft Wins $7 Million Spam Settlement; Complaints From AOL Members Drop 85%," *InformationWeek*, August 15, 34.

Cheng, H. and I. Bose. 2004. "Performance Models of a Proxy Cache Server: The Impact of Multimedia Traffic," *European Journal of Operational Research*, 154(1), April, 218–229.

Drucker, D. 2000. "Going Once, Going Twice, It's Sun," *InternetWeek*, May 22, 18.

Dustin, E., J. Rashka, and D. McDiarmid. 2002. *Quality Web Systems*. Boston: Addison-Wesley.

Dyck, T. and J. Rapoza. 2001. "IIS: Stay or Switch?" *eWeek*, October 29, 61–64.

The Economist. 2001. "Stealing Each Other's Clothes: Sun's Battle with IBM Raises Questions About Its Long Term Strategy," October 13, 61–63.

Epstein, J. 2004. "Standing Up to Redmond," *Latin Trade*, 12(6), June, 19.

Forelle, C. "IBM Embraces Bold Method to Trap Spam," *The Wall Street Journal*, March 22, B1.

Galli, P., 2004. "New IBM Unit to Target Emerging Markets," *eWeek*, 21(30), July 26, 9–10.

Gaudin, S. 2004. "Record Broken: 82% of U.S. E-mail Is Spam," *Internet News*, May 5. (http://www.internetnews.com/stats/article.php/3349921)

Glassman, M. 2003. "Fortifying the In Box as Spammers Lay Siege," *The New York Times*, July 31. (http://www.nytimes.com/2003/07/31/technology/circuits/31basi.html)

Graham, P. 2002. "A Plan for Spam," Paul Graham, July. (http://www.paulgraham.com/spam.html)

Graham, P. 2003. “Better Bayesian Filtering,” Paul Graham, January. (http://www.paulgraham.com/better.html)
Gralla, P. 2002. “Making a List,” *PC Magazine*, 21(10), May 21, 62–63.
Gross, G. 2004. “Judge Awards ISP $1 Billion in Spam Damages,” *Computerworld*, December 20. (http://www.computerworld.com/governmenttopics/government/legalissues/story/0,10801,98421,00.html)
Harbaugh, L. 2000. “Balancing Act,” *InternetWeek*, January 24, 26–30.
Information Week. 2004. “AOL Reports Big Drop in Spam,” December 27. (http://www.informationweek.com/story/showArticle.jhtml?articleID=56200528)
Keizer, G. 2005. “CAN-SPAM Can’t Slam Spam,” *Information Week*, January 4. (http://www.informationweek.com/story/showArticle.jhtml?articleID=56900503)
Kopytoff, V. 2004. “Spam Mushrooms Despite a New Federal Law,” *The San Francisco Chronicle*, September 2, C1.
Krim, J. 2003. “Spam’s Cost To Business Escalates: Bulk E-Mail Threatens Communication Arteries,” *The Washington Post*, March 13, A1.
Krim, J. 2004. “E-Mail Authentication Will Not End Spam, Panelists Say,” *The Washington Post*, November 11, E01.
Kuo, J. 2001. “Work-Ready Linux,” *InternetWeek*, September 10, 23–25.
Lee, M. 2004. “Don’t Give Up on E-mail,” *Network Computing*, 15(12), June 24, 20.
Lee, Y. 2000. “Low-Cost Dedicated Servers,” *Web Techniques*, 5(7), July, 88–89.
Machrone, W. 2000. “Picking the Right Server is Key,” *PC Magazine*, 19(10), May 23, 52.
MacVittie, L. 2001. “IPlanet Goes Where No Server’s Gone Before,” *Network Computing*, 12(13), June 25, 96–99.
Mears, J. and D. Dubie. 2003. “Users Banking on Blades,” *Network World*, 20(40), October 6, 1–2.
Montalbano, E. 2001. “Sun Sets on IPlanet Alliance,” *Computer Reseller News*, August 27, 12.
Morgan, C. 2000. “Web Content Management,” *Computerworld*, 34(17), April 24, 72.
Nelson, D. 2003. “Defending Your Site Against Spam,” *O’Reilly Network*, June 26. (http://www.oreillynet.com/ pub/a/linux/2003/06/26/blocklist.html)
PC Magazine. 2001. “Servers,” 20(14), August 12, 118.
PC World, 2005. “Spam Law Test,” 23(1), January, 20–22.
Petreley, N. 2001. “The Cost of Free IIS,” *Computerworld*, 35(43), October 22, 49.
Popovich, K. 2002. “Blade Servers Boast Dual CPUs, Internal SCSI Drives,” *eWeek*, 19(34), August 26, 7–8.
Roberts, P. 2004. “IETF Panel Deals Setback to Microsoft’s Spam Proposal,” *Computerworld*, 38(38), September 20, 14.
Roberts-Witt, R. 2001. “The Internet Business,” *PC Magazine*, 20(5), March 6, 8–17.
Roberts-Witt, R. 2001. “Web Server Brawn,” *PC Magazine*, 20(10), May 22, 144–151.
Sarrel, M. 2002. “The Top 100 (Undiscovered) Web Sites: What We Found,” *PC Magazine*, February 26. (http://www.pcmag.com/article/0,2997,s=25087&a=21937,00.asp)
Schafer, S. 2004. “Microsoft’s Cultural Revolution,” *Newsweek*, June 28, E10-12.
Schuchart, S. 2001. “IBM’s Small Change,” *Network Computing*, 12(9), April 30, 51–58.
Schwartz, J. 2001. “Update: How the NYSE Crashed,” *InternetWeek*, June 8. (http://www.internetwk.com/story/INW20010608S0006)
Shankland, S. 2002. “Sun to Announce Leap into Linux,” *CNET News.com*, August 5. (http://news.com.com/2100-1001-948277.html)

Shankland, S., M. Kane, and R. Lemos. 2001. "How Linux Saved Amazon Millions," *CNET News.com*, October 30. (http://news.cnet.com/news/0-1003-200-7720536.html)
Shen, X. 2005. "Intellectual Property and Open Source: A Case Study of Microsoft and Linux in China," *International Journal of IT Standards & Standardization Research*, 3(1), January-June, 21–43.
Shonfeld, E. 2002. "To Preserve Social Capital Get Big Boxes and Some Really Mean Software," *Business* 2.0, 3(3), March, 57.
Stauffer, T. 2000. "New Features and Fast Load Times Keep LandsEnd.com Humming," *Publish*, November. (http://www.publish.com/features/0011/feature3.html)
Ulfelder, S. 2004. "Spam-busters,"*Network World*, 21(12), March 22, 69–71.
Vijayan, J. 2001. "Sun Attempts to Woo Users Away from IIS," *Computerworld*, 35(42), October 15, 28.
Wagner, M. and T. Kemp. 2001. "What's Wrong with eBay?" *InternetWeek*, January 15, 1–2.
Xiaobai, S. 2005. "Developing Country Perspectives on Software: Intellectual Property and Open Source, a Case Study of Microsoft and Linux in China," *International Journal of IT Standards & Standardization Research*, 3(1), January-June, 21–43.
Xinhua, 2004. "Microsoft Teams Up with China's Leading Server and Solutions Supplier, November 9. (http://www.xinhua.org)
Yager, T. 2002. "Server Blades," *InfoWorld*, 24(29), July 22, 36–37.