Upon completing this chapter, you will be able to do the following:

- Identify the need for network security
- Identify some of the causes of network security problems
- Identify some characteristics and motivating factors of network intruders
- Identify the most significant network security threats
- Choose countermeasures to thwart network security attacks

CHAPTER 1

Evaluating Network Security Threats

This chapter examines the potential threats to an enterprise network. It considers why we need network security by examining the network security issues and challenges facing a network manager and the three primary reasons for network security vulnerabilities. The chapter attempts to provide a snapshot of network intruders so that you can understand your adversary. It addresses some of the most common types of threats, the tools used to implement them, and the tools used to thwart them. This chapter presents a summary of some of the tools and methods used to execute the four major categories of network security threats: reconnaissance, unauthorized access, denial of service, and data manipulation. The summary of attacks also gives some tips on how you can thwart the attacks. This chapter concludes with a list of resources you can access to learn more about how to protect yourself from network intruders.

Why We Need Network Security

The Internet economy is rapidly changing the way we work, live, play, and learn. It is especially affecting businesses and governments on a global scale. Business leaders recognize the strategic role that the Internet plays in their company's ability to survive and compete in the 21st century. Consumers and end users want secure methods to communicate and carry out electronic commerce. Unfortunately, because the Internet was based on open standards and ease of communication, it was initially missing some key security components such as controlling remote access, privacy of communications, and prevention of attacks that can deny services to others. The need to secure Internet-related communications has driven the growth of network security technologies.

Businesses face a daunting security problem: how to implement and constantly update defenses and practices to reduce business vulnerability to evolving hacker threats.

Security can be difficult to implement uniformly across the enterprise because some solutions work only in the campus, and others work only in the wide-area network (WAN). Some security solutions work well for smaller enterprises but are not practical as the enterprise grows in terms of effort, time, or cost to implement. This security problem is compounded by the added vulnerability created by the Internet connection, which gives a network intruder potential entry into your business infrastructure.

The security challenge that businesses face today is one of sorting through a wide range of solutions and choosing the right combination. A large number of security technologies and products exist. It is not the lack of technology that makes securing the network difficult. The problem is choosing among the many different selections available and adopting those that satisfy your unique network and business requirements while minimizing the support required with differing vendor technologies.

After the network engineer or system administrator has selected the right mix of security products for the network environment, the different products must be integrated throughout the entire enterprise to achieve a single, consistent security policy, which is a large challenge in today's environment. Cisco currently has many security products that enable a powerful security policy, and it is developing many more security products for future release. Cisco's security products are being developed under the Cisco SAFE architecture, a dynamic security framework for e-business networks.

Why We Have Security Issues

Campus, dialup, and Internet network access are being widely implemented in today's business environment. Yet each of these network environments poses network security risks and issues. The network and computing equipment used to implement access might inherently contain security exposures, might be configured incorrectly, or might be implemented and managed improperly. When you additionally consider the types and motivating factors of network intruders themselves, the need for network security becomes apparent. Each of these issues is considered in the following sections.

Three Primary Reasons for Security Issues

There are at least three primary reasons for network security threats:

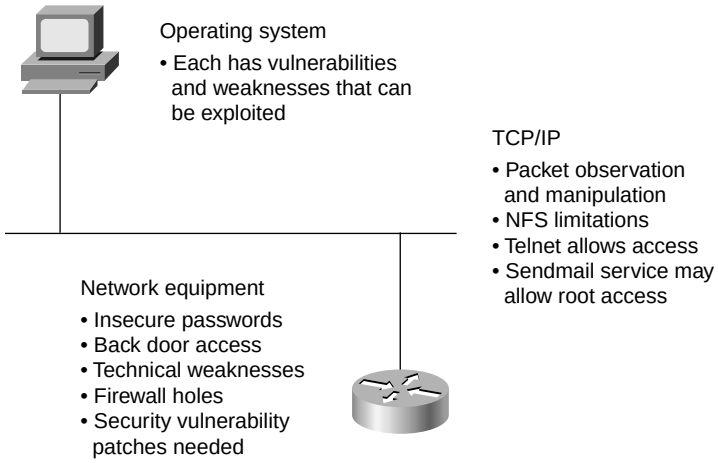
- **Technology weaknesses**—Each network and computing technology has inherent security problems.
- **Configuration weaknesses**—Even the most secure technology can be misconfigured or misused, exposing security problems.
- **Policy weaknesses**—A poorly defined or improperly implemented and managed security policy can make the best security and network technology ripe for security abuse.

There are people who are eager, willing, qualified, and sometimes compensated to take advantage of each security weakness and to continually discover and exploit new weaknesses. Each weakness is explored in a bit more depth in the next few sections.

Technology Weaknesses

Computer and network technologies have intrinsic security weaknesses or vulnerabilities. Technology weaknesses considered here include TCP/IP, the operating system, and network equipment weaknesses, as illustrated in Figure 1-1.

Figure 1-1 *Networking and Computing Equipment Contains Technology Weaknesses*



The Computer Emergency Response Team (CERT) archives at www.cert.org document many technology weaknesses for protocols, operating systems, and network equipment. CERT advisories address Internet-technology security problems. They offer an explanation of the problem, information that helps you determine if your site has the problem, fixes or workarounds, and vendor information.

TCP/IP Weaknesses

TCP/IP was designed as an open standard to facilitate communications. The services, tools, and utilities derived from it were each designed to also assist in open communications. Here are some examples of the intrinsic vulnerabilities of TCP/IP and its services:

- IP, TCP, and UDP packet headers and their contents can be observed, modified, and re-sent without detection.
- The Network File System (NFS) can enable insecure trusted access to hosts. NFS does not provide for user authentication and uses randomly assigned UDP ports for its sessions, making limited protocol and user access virtually impossible.

- Telnet is a powerful service that can give users access to many Internet utilities and services that might not be otherwise available. Hackers can use Telnet by specifying a port number parameter in addition to a host name or IP address to initiate an interactive dialog with a service that is known to be insecure.
- The UNIX sendmail daemon can allow access to the UNIX root, enabling unintended access to the UNIX system. sendmail is a program used to run e-mail on UNIX systems. It is a complex program that has a long history of security problems, including the following:
  - sendmail can be used to gain access to the UNIX root level by exploiting sendmail commands in fabricated e-mail transmissions.
  - Intruders can determine which operating system **sendmail** is running on by looking at the version number returned by fabricated **sendmail** messages. This information can then be used to launch attacks on vulnerabilities specific to the operating system version.
  - sendmail can be used to learn which hosts belong to a domain name.
  - sendmail can be exploited to redirect mail to unauthorized destinations.

Operating System Weaknesses

Each operating system has security problems and weaknesses that must be addressed. Linux, UNIX, Microsoft Windows 2000, Windows NT, Windows 98, Windows 95, and IBM OS/2 each have problems that have been detected and documented.

The CERT archives document many operating system weaknesses. Each operating system vendor or developer has information on specific known vulnerabilities and methods to overcome them. It is likely that many other operating system vulnerabilities exist that have yet to be detected, documented, and resolved.

Network Equipment Weaknesses

Network equipment from each vendor has security weaknesses that must be recognized and protected against. Some examples include insecure password protection, lack of authentication, routing protocols, and firewall holes. Most vendors quickly fix network equipment weaknesses when they are discovered. You can usually easily repair such weaknesses by applying a software revision or patch or by upgrading the equipment's operating system.

A hole allows unauthorized users to access or increase their level of access to a system. It can be a feature or bug in hardware or software. Most holes in network equipment and networked computers are well known and documented, such as in CERT advisories. For example, Cisco notifies users and the Internet community about potential security problems

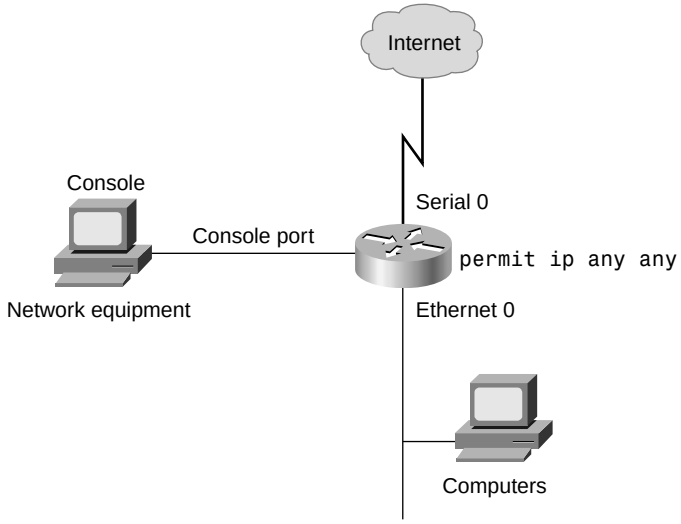
in Cisco products through Internet Security Advisories, summarized at www.cisco.com/warp/customer/707/advisory.html. This URL requires a CCO login for access. Advisories published by Cisco are usually summarized or referenced at the CERT Web site.

Note that early or limited deployment releases of Cisco IOS Software typically contain unknown holes, compared with general deployment releases, which are more thoroughly tested.

Configuration Weaknesses

Configuration weaknesses, illustrated in Figure 1-2, are a close relative of technology weaknesses. Configuration weaknesses are problems caused by not setting up or configuring networked equipment to prevent known or potential security problems. The good news about configuration weaknesses is that, once they are known, they can easily be corrected at minimal cost.

Figure 1-2 *Security Problems Caused by Configuration Weaknesses or Misuse of Equipment*



Here are some examples of configuration weaknesses:

- **Insecure default settings within products**—Many products have default settings that enable security holes. Users should consult manufacturers or user groups to identify and correct insecure default settings.
- **Misconfigured network equipment**—Misconfiguration of the equipment itself can cause significant security problems. For example, misconfigured access lists, routing protocols, or SNMP community strings can open up large security holes.

- **Insecure user accounts**—User account information may be transmitted insecurely across the network, exposing usernames and passwords to snoopers.
- **System accounts with easily guessed passwords**—This common problem is the result of poorly selected and easily guessed user passwords. For example, NetWare, UNIX, and Windows NT systems might contain legacy accounts with the username guest with the password guest.
- **Misconfigured Internet services**—A common problem is to turn on Java and JavaScript in Web browsers, enabling attacks via hostile Java applets. Network equipment or computer operating systems might enable insecure TCP/IP services that could allow remote access.

The good news about configuration weaknesses is that you can easily learn what the configuration weaknesses are and correctly configure computing and network devices to compensate by consulting CERT advisories, current documentation from network equipment vendors, and informational Requests for Comments (RFCs) that describe the best current practices for network configuration, such as RFC 2827, "Network Ingress Filtering."

Network Security Policy Weaknesses

A written, sound security policy that can readily be implemented by the organization creates a bulwark of network security. Yet some security problems can be caused by security policy weaknesses, including the following:

- **Lack of a written security policy**—An unwritten policy cannot be consistently applied or enforced.
- **Internal politics**—Political battles, turf wars, and internecine conflict will hinder the ability to have and enforce a consistent security policy.
- **Lack of business continuity**—Frequent replacement of personnel leads to an erratic approach to security.
- **Logical access controls to network equipment are not applied**—Poorly enforced and administered user password procedures allow unauthorized access to the network.
- **Security administration is lax, including monitoring and auditing**—Inadequate monitoring, auditing, and correction of problems allow attacks and unauthorized use to continue, which wastes company resources and exposes the company to legal action.
- **Lack of awareness of having been attacked**—The organization might not even be aware that it has been attacked because it does not monitor the network closely or have an intrusion detection system.

- **Software and hardware installation and changes do not follow the policy**—Unauthorized changes to the network topology or installation of unapproved applications create security holes.
- **Security incident and disaster recovery procedures are not in place**—The lack of a security incident or disaster recovery plans allows chaos, panic, and confusion to occur when someone attacks the enterprise.

Chapter 2, "Evaluating a Network Security Policy," covers how to evaluate a security policy in more depth.

Know Your Enemy: Inside the Mind of the Intruder

You can better protect your network if you know who the intruder is. The people who steal from you can be relentless. They are probably intelligent and are likely to find ways around static security implementations. For effective long-term security, you need to invest in a robust security architecture and a continuous, multistep security process. Refer to the "References" section at the end of this chapter for historical information on actual hacker exploits.

Who are network intruders? They are an extremely diverse lot who defy categorization. Yet this section attempts to help you know your enemy. Network intruder motivations are complex and numerous. The network intruder may fall under either the internal or external threat category.

In this book, we refer to an individual who attempts to access network or computer resources without authorization as a *network intruder*, or *intruder*. The intruder can be further classified as either a cracker or a hacker:

- **Cracker**—A person who uses advanced knowledge of the Internet or networks to probe or compromise network security without authorization. The cracker usually has malicious intent.
- **Hacker**—A person who investigates the integrity and security of an operating system or network. Usually a programmer, the person uses advanced knowledge of hardware and software to hack systems in innovative ways. The hacker then often freely shares his knowledge with others, usually over the Internet, which can prove an embarrassment to the victim. The hacker usually does not have malicious intent and is trying to offer a service to the Internet community. Hackers are also known as "ethical hackers" or "white hat" hackers.

Internal Threats

Internal threats are perpetrated by those inside an organization through intentional or unintentional activities such as the following:

- **Current employees with less-than-honorable intentions**—Employees who might want to test security vulnerabilities or who might even have malicious intent, hoping to exploit their employer's trust for profit or theft.
- **Current employees pursuing unintentional activities**—Employees who accidentally download a virus or other harmful program or who accidentally access a sensitive internal network or host.
- **Employees who mismanage the environment**—Employees who do not use safe passwords or who misconfigure network equipment out of ignorance.

External Threats

External threats are carried out by those outside an organization through intentional or unintentional activities such as the following:

- **Thrill seekers**—Many intruders do their work for excitement or to impress peers.
- **Competitors**—Your competition might enlist the help of a competitive analysis group to gain access to sensitive competitive information.
- **Enemies**—Many governments are concerned about information warfare from friendly or hostile countries motivated by nationalism, zealotry, or ideology. For example, during the Kosovo conflict, the NATO Web site experienced increased hacker activity.
- **Thieves**—Intruders might seek specific, valuable information for profit or some other purpose.
- **Spies**—Industrial espionage is on the increase.
- **Hostile former employees**—Employees with inside knowledge seeking revenge, thrills, or profit.
- **Others**—People might perform network intrusions for sport or for the challenge of it, to learn, or out of boredom, curiosity, or the need for acceptance from peers.

Original Intruder Skill Set and Characteristics

Network intrusion started with people gaining unauthorized access to telecommunications resources, commonly known as *phracking*. Early network intruders typically had the following skill set and characteristics:

- Knew how to code in several programming languages:
  - **C, C++**—Essential programming languages
  - **Perl**—A computer scripting language
  - **CGI (Common Gateway Interface)**—A Web server application programming interface

  - — **Microsoft Visual Basic (VB)**—A user-friendly programming environment
  - — **Java**—A portable derivative of C and C++
- Had in-depth knowledge of TCP/IP protocols, services, and tools.
- Was very experienced at using the Internet.
- Intimately knew at least two operating systems. For example, could use UNIX and DOS, or UNIX and VMS.
- Had a job using computers or networks. Enjoyed working with computer equipment as a way of life.
- Collected computer hardware and software. Had a variety of computers to work with.

Current Intruder Skill Set and Characteristics

Network intrusion techniques and tools are now widely known and available. The current network intruder skill set and characteristics can be typified in the following key points:

- Can download prewritten software tools from hacker Internet or bulletin board sites. Many network intrusion and testing tools exist, their source code is readily available, and more are being added daily. The only skill required is using a compiler to generate an executable from the hacker source code.
- Uses prewritten scripts and utilities in creative ways to intrude into networks and computer systems. Can use a tool to automatically probe a network for weaknesses.
- Is in an age group that has plenty of time to experiment and develop techniques. A student or hobbyist with a great interest in technology. Is also known as a "script kiddie" because his attacks are primarily carried out with scripts or programs written by someone else.

Note the shorter list for current intruders. Regardless of the type, category, or motivation of network intruders, we must find methods to thwart them.

Security Threat Types

The vast range of network security threats defies efforts to categorize them, understand what they are, and devise methods to protect against them. To help you get your arms around network security threats, we have created the following categories of network security threat types, which are considered in turn in this section of the chapter:

- Reconnaissance
- Unauthorized access
- Denial of service
- Data manipulation

The categories of security threats are generally known as *vulnerabilities*—attributes of a computer or network that permit someone to initiate exploits against the network. An *exploit* is a method to take advantage of a vulnerability by a manual procedure, a script, or an executable program. The purpose of the exploit is to collect system information (reconnaissance), deny system services to valid users, gain unauthorized access to systems or data, or manipulate data.

Vulnerabilities and exploits each have a telltale *signature*—a distinctive, recognizable state or state transition. Intrusion detection systems (IDS), such as the CiscoSecure IDS, can recognize exploit signatures as they are carried out. As soon as vulnerability and exploit signatures are known and recorded, countermeasures can be identified to either fix the vulnerability itself or somehow block the exploit from working against the vulnerability. Vulnerability science is the study of vulnerabilities and exploits.

In the following sections, we will consider the four vulnerability and exploit categories.

Reconnaissance

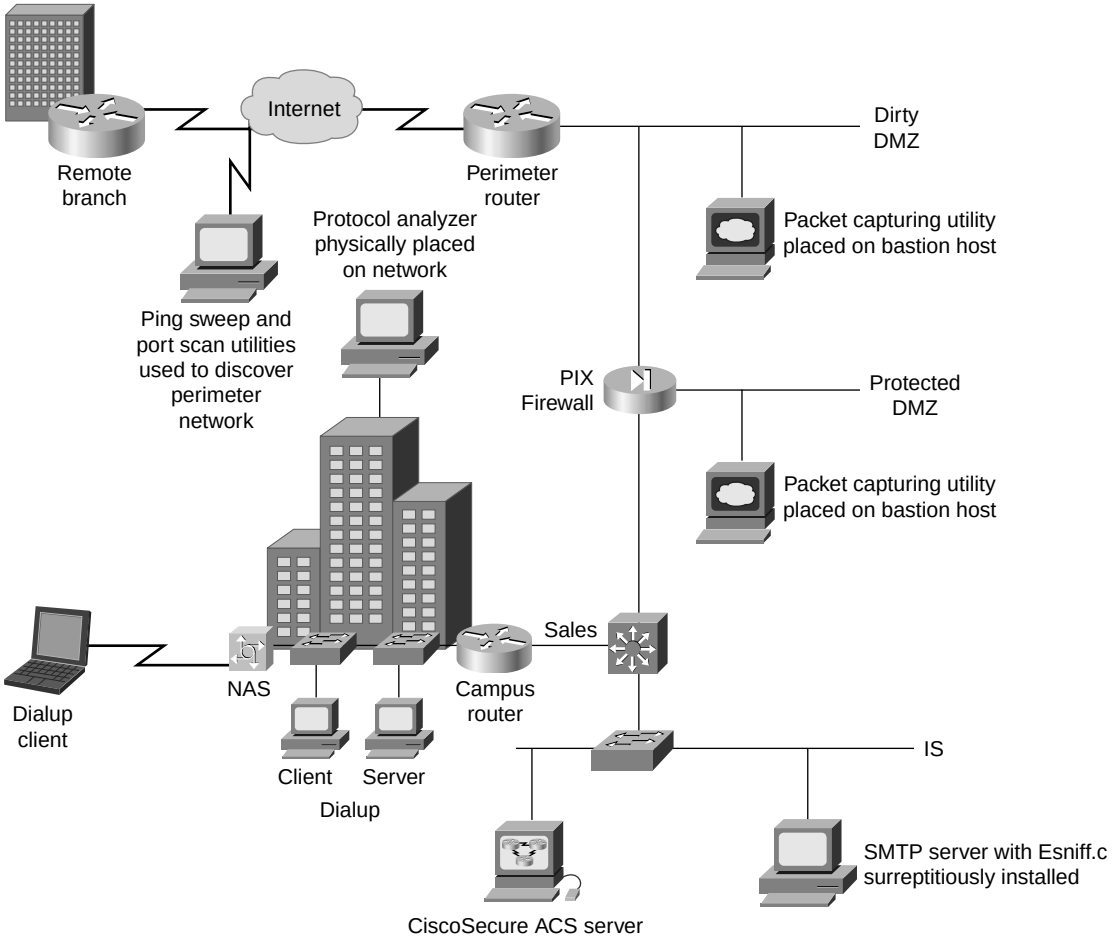
Reconnaissance is the unauthorized discovery, mapping, and monitoring of systems, services, or vulnerabilities in a network. Reconnaissance also includes monitoring network traffic. Reconnaissance can be carried out either actively or passively. The information gathered by reconnaissance can then be used to pose other attacks to the network or to steal vital data. Figure 1-3 illustrates where reconnaissance attacks can take place in an enterprise network.

Reconnaissance attacks can take the form of target discovery, eavesdropping, and information theft. The following sections consider reconnaissance attack types, examine exploits used to carry out the attacks, and describe countermeasures you can take to prevent reconnaissance attacks.

Target Discovery

Discovering the targets for reconnaissance includes finding out domain names and associated IP addresses, learning the IP address range of a target organization, or finding out specific IP addresses of target hosts. A specific host can then be targeted to learn of available services or host information. For example, the hacker might try to learn the IP address of a perimeter router's interface connection to an ISP so that he can attack the router. Target discovery can be carried out with common query-type network commands, **ping** sweeps, and port scans.

Figure 1-3 *Examples of Reconnaissance Attack Locations*



Network Commands

Reconnaissance can be accomplished using network commands readily available on UNIX, Windows, and Linux systems: **ping**, **whois**, **finger**, **rusers**, **nslookup**, **rpcinfo**, **telnet**, **dig**, **nmap**, and other commands or utilities that provide information about a host or network. The commands can be exercised individually or by using public domain utilities that combine query-type commands to accomplish a specific purpose. Some utilities can be used to gather information about network devices by exercising IP header options using synthesized packets and then gathering information sent in reply to the bogus packets.

ping Sweeps

Although individual **ping** commands can be entered to gather information about a network or hosts, ping sweep utilities have been devised to automate the discovery of hosts within a network or subnet. The ping sweep utility pings a range of IP addresses and is used to perform network mapping. The ping sweep utility is used to identify potential targets to zero in on for more in-depth reconnaissance. The **ping** command generates an ICMP Echo Request against a specific host. The host must reply with a variety of ICMP reply messages. Sometimes the ping sweep utility combines the series of ICMP Echo Requests with other ICMP requests such as ICMP Timestamp, ICMP Address Mask, or ICMP Information Request to gather more information.

Port Scans

When a hacker discovers an interesting host, he or she can then carry out a port scan against the host. A port-scan utility checks a range of TCP or UDP ports on a host to determine network services that are available, such as Telnet, FTP, HTTP, or RCP. A port scan can be general, in which a range of ports are probed, such as ports 1 to 1023. A port scan can also be specific, zeroing in on certain ports to discover such things as operating system information, host names, or usernames. Port-scanning utilities can use packet fragmentation and set SYN and FIN bits in TCP headers in combination to attempt to conceal the port scan.

After specific ports are found to be open, an attack against a specific port can be mounted. For example, after SMTP is discovered to be available on a host, the hacker can send SMTP commands to gather more information or even to gain unauthorized access. Or the hacker could try to gain Telnet or FTP access to a host to learn more about the host from header information sent in reply to the access. In another example, after a hacker determines that the Domain Name System (DNS) is available on a host, he could try to access Host Information (HINFO) records from the DNS service. The HINFO record is an optional record type that allows system information to be recorded and retrieved. This information typically includes the operating system and hardware platform that the system is running on. There is very little need to include this record in the database, and it provides attackers with valuable targeting information. Port scanners can explore an open port of insecure user accounts or vulnerability to remote access.

Examples of tools used to carry out ping sweeps and port scans include System Administrators Tool for Analyzing Networks (SATAN); security scanners made by networking vendors; **portscan.c**, nmap, and neptune (Linux port scanners that report the services running on another host); and other public-domain scanners such as Network Toolbox.

Eavesdropping

Eavesdropping (also known as information gathering) is a method of passively observing network traffic with a device or utility. The purpose of eavesdropping is to observe traffic patterns and to capture the traffic for analysis and information theft. *Network snooping* and *packet sniffing* are common synonyms for eavesdropping. The information gathered by eavesdropping can be used to pose other attacks to the network or to steal information. A common way to eavesdrop on communications is to capture TCP/IP packets and decode the contents using a protocol analyzer or a similar utility. Captured packets that are part of a session logon can be replayed to help the intruder gain access.

Network intruders can use eavesdropping to identify usernames and passwords in order to gain unauthorized access to network hosts or to identify information carried in the packet, such as credit card numbers or sensitive personal information.

An example of data susceptible to eavesdropping is SNMP version 1 community strings, which are sent in cleartext. An intruder could eavesdrop on SNMP queries and learn valuable information about network equipment configuration.

Table 1-1 describes some types of devices used for eavesdropping.

Table 1-1 *Devices Used for Eavesdropping*

Category	Type	Description
Packet-capturing utilities	**tcpdump**, **esniff.c** for UNIX **linsniffer.c** for Linux Microsoft's Network Monitor on Windows NT systems	Utility software installed on host. Requires network interface card in promiscuous mode.
Protocol analyzers	Network Associates' Sniffer or NetXray Hewlett Packard's Internet Advisor LAN Protocol Analyzers	Software installed on host or dedicated test equipment.

Information Theft

Network eavesdropping can lead to information theft. The theft can occur as data is transmitted over the internal or external network. The network intruder can also steal data from networked computers by gaining unauthorized access.

A common network intrusion is for the intruder to take files or use resources that do not belong to him. Examples include breaking into or eavesdropping on financial institutions and obtaining credit card numbers. Another example is accessing and copying a computer's password file and then using another computer to crack that file.

Table 1-2 describes methods you can take to counteract reconnaissance attacks against your network.

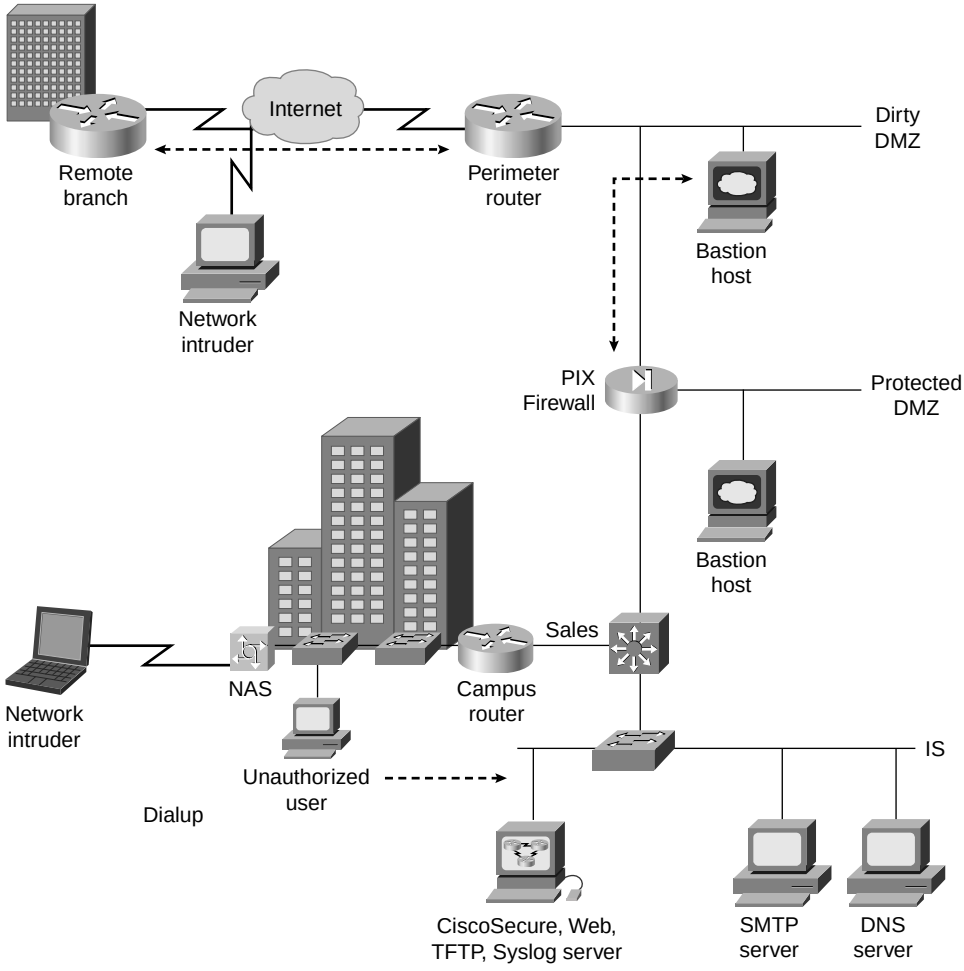
Table 1-2 *Methods to Counteract Reconnaissance Attacks*

Attack	Prevention
Target discovery	Turn off responses to query-type commands such as **finger** and **nslookup** on routers and network hosts. Use an IDS to detect such attempts.
ping sweep	Turn off responses to **ping**s and use IDS to detect.
Port scan	Use a port scanner to identify open ports. Turn off nonessential services on routers and network hosts and use IDS to detect port scans. Can reduce vulnerability to SYN flooding attacks.
Eavesdropping	Limit physical access to campus network equipment to prevent placement of protocol analyzers on network segments. Use Ethernet switches in the internal network to segment a local area network and prevent capturing of network-wide traffic from one workstation. Prevent unauthorized network access to hosts to prevent placement of packet-capturing utilities. Use file integrity checkers on hosts to detect any unauthorized placement. Use network interface cards that cannot be placed in promiscuous mode in sensitive hosts. Physically check the host for promiscuous mode interfaces. Run promiscuous mode-checking software such as **cpm** and **ifstatus** on each host. Use data encryption technology to limit the ability to observe network traffic content across insecure networks. Encryption must meet the organization's data security needs without imposing an excessive burden on the system resources or the users.

Unauthorized Access

A network intruder can gain unauthorized remote access to networked computers or networking devices through a variety of means. A common goal of the intruder is to gain root (UNIX) or administrator (Windows NT) access to a networked computer, where the intruder has great power to control the target computer or to access other networked computers. Figure 1-4 illustrates some of the points at which a network intruder might attempt to gain unauthorized access.

Figure 1-4 *Unauthorized Access Attack Points*



Gaining Initial Access

The network intruder usually needs to gain initial access to a networked host, and then he tries to increasingly penetrate the host and any connected networks. The intruder establishes a connection to a host without owning an account on it and can then try to find holes that allow more privileged access such as root access on UNIX systems.

The network intruder typically tries to gain as much information as possible about the target host through reconnaissance techniques and then uses the information gained to obtain initial access. The intruder might discover the IP address of a host he wants to penetrate and then might use a packet sniffer to capture usernames and passwords on the host. The intruder might attempt to find vulnerable Internet services that can be exploited to gain remote access using a port scanner.

The intruder might use social engineering to gain initial or even privileged access. Social engineering is a way to overcome information security devices by convincing someone to reveal needed information, such as usernames and passwords, or other remote access information.

The network intruder might try to gain access through dialup access using a "war dialer," a program that simplifies dialing a range of telephone numbers with the hopes of finding data ports connected to modems.

The network intruder might also be working from inside an organization, being a local user. The intruder can then exploit the trust relationship he enjoys by being an employee or other trusted person. Because the inside user has account privileges, he is inside the firewall and has fairly free rein inside the network.

Password-Based Attacks

A variety of password-based attacks can allow a network intruder to gain remote access. As soon as the network intruder has obtained a username, he hopes that the user has created an easily guessed password, and he tries to guess the user's password manually using brute force. The easiest way to access a network host is through the front door by entering the login command. The intruder can then try to enter a password that will let him gain initial access.

The intruder might be able to capture a username and password sent in the clear using a packet sniffer. Or he might obtain a host's password file such as the UNIX /etc/passwd or the Windows NT SAM hive, and he might try to learn the passwords using a password-cracking utility. The password cracker is used to guess encrypted user passwords. The utility does not really "crack" passwords. It simply guesses passwords by using computing power to match password hashes. Cracking utilities are readily available for both UNIX and NT. The intruder then uses the username and password to gain trusted access to the networked computer.

You can protect against password-based attacks by creating and enforcing a security policy that requires hard-to-crack or hard-to-guess passwords that include nonalphanumeric and capitalized characters, by not sending unencrypted passwords over an insecure network, and by carefully guarding remote access to a host's password file.

Gaining Trusted or Privileged Access

A *trusted computer* is a computer that you have administrative control over or one that you consciously make a decision to "trust" to allow access to your network.

As soon as the intruder has networked computer access, he exploits this access to gain access to more powerful privileged access or to exploit the trust relationship between networked computers to access other network hosts. The goal of gaining privileged access is to achieve root or administrator-level privileges on a host without owning a privileged account on it. The attacker can spoof a trusted user by using that person's username and password, or he can spoof a trusted host to gain access to other hosts.

The most commonly used applications in UNIX systems that use trusted host features are the **rlogin**, **rsh**, and **rcp** commands.

Intruders might also attempt to exploit operating system vulnerabilities that allow an intruder to gain unauthorized root access.

Gaining Secondary Access

After an intruder gains initial access, he might attempt to establish an inconspicuous access avenue back to the host, clear any evidence of his intrusion, and later return and use the penetrated host as a springboard to access more targets. The intruder might try some of the following means to hide the unauthorized access:

- Clean logs and remove traces of remote access
- Move accounting files to the /tmp directory, the contents of which are eventually deleted
- Install a packet sniffer to observe traffic
- Install a backdoor(s) by establishing usernames and passwords or by installing Trojan horse programs such as rootkit or BackOrifice

Attack Services Allowing Remote Access

Many IP applications and services can make your hosts and network devices very vulnerable to remote access attacks. Many applications were developed to facilitate, not prevent, communications. Some services have little or no authentication methods built in to ensure that the remote user is allowed access. You should disable unused services that can allow remote access on network hosts and equipment.

Table 1-3 summarizes some of the many IP services that are vulnerable to attack. It lists the type of service and briefly describes the service's vulnerability.

Table 1-3 *IP Applications or Services That Are Vulnerable to Remote Access Attacks*

Type	Vulnerability Description
BSD **r** commands	Authentication of remote access using the **r** commands is by source address and is easily spoofed, providing full access to remote hosts running the remote services.
FTP	Anonymous FTP allows intruders to read and possibly write files to a host. Do not use it unless absolutely necessary. Control write access.
finger	**finger** service can be used to discover information about users, a prelude to obtaining usernames.
NFS	Allows access to files on remote systems. Has weak authentication (source IP address) of requests that is easily spoofed.
Telnet	Allows a user to remotely access a command shell in cleartext. Controlled by a simple username/password authentication mechanism that can be easily spoofed.
TFTP	Intruders can easily request file transfers using TFTP because it has no authentication mechanism.
SMTP, POP, MIME, sendmail applications	Intruders can manipulate the sendmail environment to gain root privileges.
HTTP, Web servers	Vulnerabilities include bugs in server software, misconfiguration, and insecure operating systems. Java, JavaScript, and ActiveX applets can act as viruses or Trojan horses.

Program Vulnerabilities That Permit Remote Access

Many programs used for Internet communications and applications were written in the C programming language. The programs use buffers, areas of fixed length in working memory, for variable data. Buffer overflows occur when a network intruder with a knowledge of C programming deliberately tries to exceed the fixed length of program buffers to gain unauthorized access to the target host. Probably the most famous example of exploiting buffer overflow errors in programs is the Morris Worm that spread across the Internet in 1988. Buffer overflows are a common programming error. You can protect

yourself from buffer overflows by installing the latest software and software patches, which you can discover by monitoring operating system advisories and vendor Web sites.

Network intruders can attempt to gain remote access by exploiting operating system vulnerabilities. Every operating system has inherent vulnerabilities that a network intruder can exploit to gain unauthorized access. You can protect yourself from operating system vulnerabilities by monitoring Web sites, network advisories, and vendor Web sites to learn which versions of operating systems are the most secure and by installing the latest operating system patches.

Misuse of Systems After Gaining Unauthorized Access

After network intruders gain access to networked computers, they can then use the hosts for unauthorized purposes. They can place unauthorized files or resources on another system for ready access by other intruders. Examples of unauthorized files include the following:

- **GIFs**—Unauthorized use of a computer to create a library of GIF and other electronic picture files. Altering GIFs and Web site content.
- **Hacker tools**—Unauthorized use of a computer to store, test, and distribute software tools that are useful for network intrusion. These tools are then widely available by associated network intruders.
- **Unlicensed versions of software for free distribution**—The term *WareZ* applies to unauthorized distribution of software.

Methods Used to Counteract Remote-Access Attacks

You can take many steps to counteract remote access attacks. Starting at the network perimeter, you can use a Cisco perimeter router to permit Internet access to only destinations you choose. You can use Cisco IOS Software features to limit access to remote services; to control user access; to filter traffic based on source and destination address, protocol, or port; or even to use the Lock and Key feature. Lock and Key couples access control lists with a challenge/response mechanism that challenges users requesting access to a corporate or campus network. At the network perimeter, you can set up a second line of defense with the CiscoSecure PIX Firewall to protect access to your internal network. Cisco routers and the PIX Firewall can interoperate with a CiscoSecure Access Control Server to control access by username, password, and service. At network hosts, you can simply turn off unneeded services; install hardened programs that offer remote-access services; ensure that operating systems and servers are correctly configured with the latest version, updates, and patches installed; and take other prudent steps to limit remote-access attacks. And you can use the CiscoSecure Intrusion Detection System to scan for and detect remote-access attack signatures and to alert you to attacks.

Table 1-4 summarizes methods you can take to counteract remote-access attacks against your network.

Table 1-4 *Methods to Counteract Remote-Access Attacks*

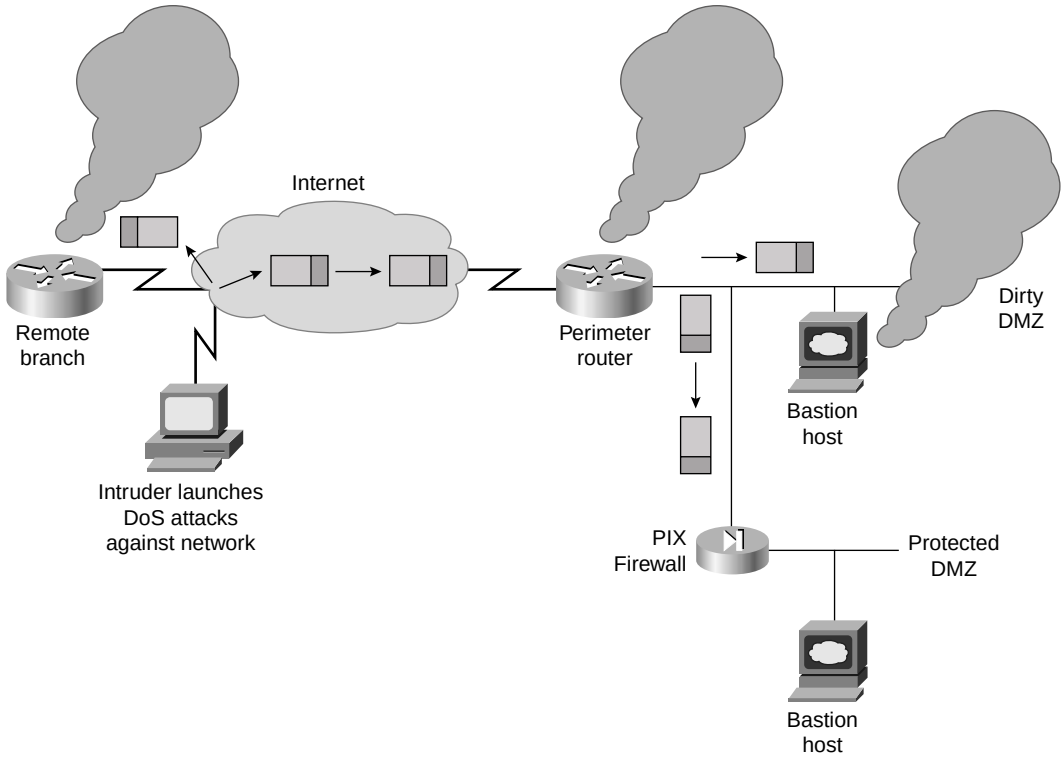
Attack	Prevention
Initial access	Limit points of access to your network, such as controlling setting up unauthorized dialup access by internal users.
	Use a AAA server to manage remote-access privileges, usernames, and passwords such as CiscoSecure Access Control Server.
	Use a more secure remote access protocol such as PPP, CHAP, or MS-CHAP.
	Limit or do not use a shell login that only requires a password.
	Use the Lock and Key security feature in Cisco IOS Software.
Password attacks	Enforce a hard-to-crack/hard-to-guess password policy.
	Run password crackers as an administrator to detect weak passwords.
	Use password aging features to force users to change their passwords often.
Trusted access	Ensure access security to root or administrator levels.
	Monitor and maintain trust relationships.
Secondary access	Scan for viruses and Trojan horses that might have been installed as back doors.
	Scan for open ports originating from network hosts that are not expected that might have been opened by Trojan horses.
	Install and run file integrity checkers to monitor for unauthorized file and directory tampering.
	Use encryption to secure the data on your host's hard drives.
Remote-access services	Disable all unused services and commands that you are not actively using.
	Ensure that trust relationships between hosts are secure.
	Install secure versions of programs that run Internet services such as the latest versions of Web, mail, and FTP servers.
	Change default configuration values, such as allowing Read/Write access to everyone, to values you have determined.

Denial of Service

Denial of Service (DoS) is an attempt to disable or corrupt networks, systems, or services and thereby deny network services to legitimate users. Network intruders sometimes derive pleasure from denying the use of a public service to others, similar to vandalism. Network intruders might use DoS attacks to test a system's vulnerability to attack, as a prelude to

further attacks, to cover their tracks after gaining unauthorized access, or simply as retaliation. The IP protocol is vulnerable to DoS attacks, so many attack types are available and relatively easy to carry out, much as it is relatively easy to commit vandalism. DoS attacks can be launched against a perimeter router, bastion host, or firewall, as illustrated in Figure 1-5.

Figure 1-5 *DoS Attack Points*



Resource Overload

Resource overload DoS exploits are an attempt to overload a target host or network equipment's resources with the result of causing the target host or network equipment to cease operating or be unavailable to legitimate users. The exploits attempt to overload target resources including bandwidth of an interface, internal memory space (buffers), CPU processing capability, or disk drive space.

Table 1-5 lists the attack type, lists typical tools used to carry out the exploit, describes the exploit, and summarizes some countermeasures you can take to mitigate the attack.

Table 1-5 *Resource Overload DoS Attacks*

Type	Exploit Name	Description	Countermeasures
ping flood	**pingflood.c**, **smurf.c**, **fraggle.c**, **papasmurf.c**	**pingflood.c** sends a large number of ICMP Echo Requests to a host. **smurf** sends a large amount of ICMP Echo Request (ping) traffic to a broadcast address, with each ICMP Echo packet containing the spoofed source address of a victim host. When the spoofed ICMP Echo Request packet arrives at the destination network, all hosts on the network send ICMP Echo Reply packets to the spoofed address. The initial ICMP Echo Request is multiplied by the number of hosts on the network. Generates a storm of replies to the victim host, tying up network bandwidth, using up CPU resources, or even crashing the victim host. **fraggle** is the UDP version of smurf.	Set perimeter routers to reject responses to ICMP Echo Request packets. Turn off directed broadcasts on all internal and external routers. Set perimeter routers to reject incoming ICMP Reply packets.
Half-open syn attack	**neptune.c**, **synk4.c**	Partially initiates numerous TCP sessions against a port so that no new connections can be initiated by legitimate users.	Use TCP Intercept features in Cisco IOS software. Use syn flooding protection in CiscoSecure PIX Firewall. Use IDS to detect.

Table 1-5 *Resource Overload DoS Attacks (Continued)*

Type	Exploit Name	Description	Countermeasures
Packet storms	**chargen**, **Pepsi5.c**, UDP Bomb	**chargen** runs on port 19. It generates a never-ending stream of ASCII characters for testing. The chargen attack consists of sending a flood of UDP packets from a spoofed source IP address to the subnet broadcast address with the destination port set to 19. The target host on the subnet running chargen responds to each broadcast, creating a flood of UDP packets in an infinite loop, which ultimately results in a Denial of Service of the host. Many variations of this attack exist. **Pepsi5.c** floods a target with UDP packets containing random source host addresses. UDP Bomb forms UDP packets that have an incorrect length field in the packet header, causing some hosts to suffer a kernel panic.	Disable the chargen and echo services on all machines. Use syn flooding protection in CiscoSecure PIX Firewall. Install operating system patches.

Out-of-Band Data DoS Attacks

Out-of-band data DoS attacks are context-based in that they manipulate the IP header (TCP or UDP) to try to exceed the normal operation of IP. The result is that the target host or network equipment ceases operating.

Table 1-6 describes some out-of-band data DoS attacks, lists typical tools used to carry out the exploit, describes the exploit, and summarizes some countermeasures you can take to mitigate the attack.

Table 1-6 *Out-of-Band Data DoS Attacks*

Type	Exploit Name	Description	Countermeasure(s)
Oversized packets	**ping** of death (**simping.c**)	Modifies the IP portion of header, indicating that there is more data in the packet than there actually is, or sends a data payload exceeding the maximum allowed packet size (larger than 65,535 bytes), causing the receiving system to crash.	Filter large or fragmented ICMP traffic from your network. Cisco Systems' IDS will detect these attacks.
Overlapped packets	**winnuke.c**	Sends out-of-band data to an established connection on a Windows 95 or Windows NT host (typically to NetBIOS, port 137), causing the host to reboot or cease operating.	Turn off NetBIOS if it isn't needed. Install operating system patches (service packs) on hosts according to CERT advisory lists.
Fragmentation	**teardrop.c**	Takes advantage of some implementations of the TCP/IP IP fragmentation reassembly code that do not properly handle overlapping IP fragments, causing a memory buffer overrun.	Discard fragmented IP packets at the perimeter router for packets coming from the outside. Install operating system patches on hosts according to CERT advisory lists.
IP source address spoofing	**land.c**	Causes a computer to create a TCP connection to itself, get caught in a loop, and have to be rebooted.	Filter IP-spoofed packets at the perimeter or host. Install operating system patches on hosts according to CERT advisory lists.
Packet headers malformed	UDP Bomb	Forms UDP packets that have an incorrect length field in the packet header, causing some hosts to suffer a kernel panic.	Install operating system patches on hosts according to CERT advisory lists.

Other DoS Attacks

Unfortunately, many other DoS attacks are used to exploit IP networks. DoS attacks can also exploit vulnerabilities in specific services or hardware not necessarily related to TCP/IP. Ultimately they prevent authorized people from using a service by using up system resources. Here is a sampling of some other DoS threats:

- **Distributed Denial of Service (DDoS)**—Uses multiple coordinated systems to attack a Web site or host. See *Strategies to Protect Against Distributed Denial of Service (DDoS) Attacks* at www.cisco.com/warp/public/707/newsflash.html for more information.
- **E-mail bombs**—Many free programs exist that send bulk e-mails to individuals, lists, or domains, monopolizing e-mail services.
- **CPU hogging**—Programs such as Trojan horses or viruses that tie up CPU cycles, memory, or other resources, denying computer resources to legitimate users.
- **Malicious applets**—Java, JavaScript, or ActiveX programs that can act as Trojan horses or viruses to cause destruction or tie up computer resources.
- **Misrouting traffic**—Disabling traffic by misconfiguring routers to reroute traffic away from the intended network or host.
- **Accidental DoS**—Legitimate users or system administrators can cause DoS attacks due to misconfiguration or misuse.
- **Buffer overflows**—Microsoft's Internet Information Server (IIS) version 4.0 is susceptible to buffer overflows that will crash the server. Known susceptibilities can be fixed by installing patches or service packs.
- **CGI exploits**—Web browsers will divulge critical information when a malicious user appends certain characters to the end of the URL that refer to a server-side include file. A remote user can recover the source code for the file, disclosing proprietary information, copyrighted source code, and even usernames and passwords used to log into databases.
- **Server DoS**—Microsoft's NT Server version 4.0 (service pack 3 or 4) will reboot or freeze, depending on the amount of memory the server has, when a character string of sufficient length appears at a certain port during a Telnet session, followed by an execution command.

Methods Used to Counteract Denial of Service Attacks

The CiscoSecure Integrated Software running in Cisco routers and the PIX Firewall contains powerful security technology to provide firewall capabilities that can prevent DoS attacks or lessen their effect, including the following:

- Context-Based Access Control (CBAC) can be used for DoS detection and prevention, such as defending against SYN attacks.

- Java blocking in the PIX Firewall can filter out Java applets.
- The TCP Intercept feature in Cisco IOS Software detects and controls SYN attacks.
- Ensure that the correct version of Cisco IOS Software is installed to prevent known vulnerabilities.

You can also use the following methods to lessen the impact of DoS attacks:

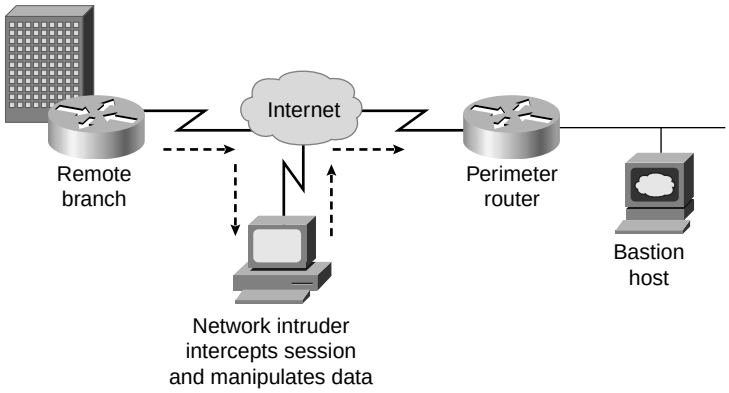
- Use audit trails to detail transactions, recording time stamp, source host, destination host, ports, duration, and total number of bytes transmitted.
- Use a real-time alerts log to generate alerts in case of Denial of Service attacks or other preconfigured conditions.

Data Manipulation

A network intruder can capture, manipulate, and replay data sent over a communication channel by using data manipulation. *Data manipulation* is also known as *impersonation*. It can take the form of IP address spoofing, session replay and hijacking, rerouting, and repudiation. Data manipulation can also include graffiti—vandalizing a Web site by accessing the Web server and altering Web pages. Data manipulation is made possible by vulnerabilities in the IP protocol and associated services and applications. Data manipulation attacks are also known as man-in-the-middle attacks because the attack usually involves a person in the middle exploiting IP session susceptibilities between two TCP/IP hosts.

Figure 1-6 illustrates where in the network data manipulation attacks can occur.

Figure 1-6 *A Data Manipulation Attack Point*



IP Spoofing

A network intruder can use IP spoofing to impersonate the identity of a host for applications or services that use source or destination addresses for authentication. An IP spoofing attack occurs when a network intruder outside your network pretends to be a trusted computer inside or outside your network. The spoof uses an IP address that is within the range of IP addresses for your network or uses an authorized external IP address that you trust and to which you want to provide access to specified resources on your network.

Spoofing usually includes manipulating TCP/IP packets to falsify IP addresses, thereby appearing to be another host. For example, the intruder could use IP address spoofing to assume the identity of a valid or trusted host and to gain the host's access privileges by falsifying the source address of a trusted host. Spoofing is also known as a masquerade attack.

An attacker can specify an arbitrary source address for a packet in an attempt to bypass address-based authentication mechanisms. This is especially effective if the arbitrary source address is that of a host behind a perimeter router or firewall.

Normally, an IP spoofing attack is limited to the injection of data or commands into an existing stream of data passed between a client and server application or a peer-to-peer network connection.

Attackers using IP spoofing might be able to bypass authentication mechanisms and, if they are improperly implemented, might subvert filters on packet-filtering routers.

Countermeasures against IP spoofing include filtering packets at the perimeter router that come from outside but claim to be from inside. The CiscoSecure IDS detects these attacks.

Session Replay and Hijacking

Session replay is an attack in which a network intruder intercepts and captures a sequence of packets or application commands, manipulates the captured data (such as to alter the dollar amount of a transaction), and then replays the data to cause an unauthorized action. Session replay exploits weaknesses in authentication of data traffic.

Session hijacking is an attack in which a network intruder takes over an IP session and inserts falsified IP data packets after session establishment. Session hijacking methods include IP spoofing, source and/or destination address manipulation over TCP/IP, and sequence number prediction and alteration. The network intruder uses a protocol analyzer or utility program to observe, predict, and then alter and retransmit TCP/IP packet sequence numbers.

An example of a documented session hijacking attack is the use of a tool that redirects Xterminal output to an intruder's terminal instead of the intended terminal.

One exploit that is a session replay attack involves the use of JavaScript to allow a network intruder to exploit a hole in Hotmail and other Web-based e-mail systems. The hole lets the malicious hacker create a piece of incriminating e-mail that can be falsely traced to another person's computer. The user is exposed to this attack by being lured to a seemingly innocent Web page into which the hacker has inserted the malicious JavaScript code.

Session replay and hijacking attacks can only be carried out by skilled programmers, so there have been few documented attacks. One session hijacking tool is the hunt-1.0 program that runs on Linux systems.

Countermeasures for session replay and hijacking include the following methods and technologies:

- Adjust the Web browser's security setting to prevent downloads of applets or make the browser notify you for permission to execute mobile code when it is encountered.
- Block corporate access to public e-mail sites to limit the risk of infection or disclosure of confidential data.
- Use access control features in the perimeter.
- Use authentication such as CiscoSecure TACACS+ or RADIUS, or SSL.
- Use encryption technologies to protect the integrity and privacy of data.
- Use digital signatures offered by certification authorities for nonrepudiation.

Rerouting

Network intruders can use rerouting by gaining unauthorized access to routers and altering the routing configuration or by spoofing the identity of routers or hosts along a network path. The consequence of rerouting is that it can allow a remote host to pose as a local host on your network. Services that rely on IP addresses as authentication might be compromised as a result.

The countermeasures for rerouting attacks are to limit access to routers to prevent reconfiguration of routes, to filter source-routed packets at the router, to use route authentication features in Cisco IOS Software, and to disable source routing on all hosts. The CiscoSecure IDS systems detect these attacks.

Repudiation

One or more users involved in a communication, such as a secure financial transaction, can deny participation, jeopardizing electronic transactions and contractual agreements. This prevents a third party from being able to prove that a communication between two other parties ever took place. This is a desirable quality if you do not want your communications to be traceable. Nonrepudiation is the opposite quality—a third party can prove that a

communication between two other parties took place. Nonrepudiation is desirable if you want to be able to trace your communications and prove that they occurred.

The Security Opportunity

How do companies effectively protect their networks from security attacks? Security solutions that offer all the protection you need for your network are readily available from Cisco and other vendors.

A good security solution helps you reduce the exposure of a network to security threats and thereby saves the company money. It should also reduce the total cost of implementation and operation of network security measures.

A good security solution also enables new networked applications and services that many would consider unwise and potentially dangerous, such as business-to-business electronic commerce or extranet applications to link you more closely with your suppliers and partners.

Security is a fundamental element of networking. A good security solution does the following:

- Reduces the costs of implementation and operation of network security measures
- Enables new networked applications and services
- Makes the Internet a global, low-cost access medium

Summary

This chapter established the need for network security by focusing on the following key points:

- The great increase in network security threats makes it complicated and difficult to implement integrated network security uniformly.
- There are three primary reasons for security issues: technology, configuration, and policy weaknesses.
- Network intruders have a variety of motivations and available tools for attacking networks.
- A large number of tools are available to the network intruder, including protocol analyzers, network scanners, and tools developed by network intruders to attack networks.
- Reconnaissance is a technique for learning more about a network and its equipment with the purpose of launching further attacks.

- Unauthorized access consists of a network intruder attempting to gain access to network resources without permission. This includes initial access, privileged access, and secondary access.
- Denial of Service is similar to vandalism in that the goal of the network intruder is to deny network services or access to legitimate users.
- Data manipulation is an attempt to intercept and alter data communications between TCP/IP hosts.
- Each form of network intrusion has specific countermeasures that you can implement to prevent or lessen the attack.
- The system administrator or network engineer can implement an effective security solution that can reduce security implementation costs, enable new networked applications and services, and give your organization a competitive advantage.

Review Questions

Answer the following review questions to test your knowledge of evaluating network security threats:

1. What are two characteristics of the network security problem facing businesses today?
2. List five driving factors in the growth of network security.
3. What are the three primary reasons for network security issues?
4. What are some security policy weaknesses?
5. Who typically carries out internal attacks?
6. How is a packet sniffer used to carry out reconnaissance attacks?
7. List the four stages of unauthorized access attacks.
8. Why are DoS attacks so prevalent?
9. What is the most common data manipulation attack?
10. How can an organization benefit from having network engineers and system administrators with network security expertise?

References

The topics considered in this chapter are complex and should be studied further to more fully understand them and put them to use. Use the following references to learn more about the topics in this chapter.

Network Security and Business

T. Bernstein, A. Bismani, E. Schultz, and C. Siegel, *Internet Security for Business*, Wiley Computer Publishing, 1996. Describes how to plan for and implement network security.

D. Chapman, S. Cooper, and E. Zwicky, *Building Internet Firewalls*, Second Edition, O'Reilly and Associates Publishing, 2000. Describes how to build a firewall and explains network security vulnerabilities.

Hacking and Hacker Tools

Anonymous, *Maximum Security: A Hacker's Guide to Protecting Your Internet Site and Network*, Second Edition, Sams.net Publishing, 1998. Describes hacking from the hacker's perspective.

AntiOnline, a comprehensive security site with hacking tools, at www.antionline.com.

S. McClure, J. Scambray, et al., *Hacking Exposed: Network Security Secrets and Solutions*, McGraw-Hill, 1999. Describes hacking methods and mitigation.

Root Shell, a site that summarizes vulnerabilities and provides hacker tools, at www.rootshell.com.

T. Shimomura and J. Markoff, *Takedown: The Pursuit and Capture of Kevin Mitnick, America's Most Wanted Computer Outlaw—By the Man Who Did It*, Hyperion Books, 1996. Describes a real-life hacking drama.

The Web Site for 2600—The Hacker Quarterly, summarized hacks, vulnerabilities, and security issues, at www.2600.com.

L0pht, "The top US hackers hang out at the L0pht. But why can't they spell?" *spews.net* magazine, December 1995, at www.l0pht.com.

Security Web Sites

Computer Emergency Response Team (CERT) Coordination Center, a focal point for incident response, vulnerability analysis, and training, at www.cert.org.

Microsoft Security, the official Microsoft security home page, at www.microsoft.com/security/default.asp.

NT Bugtraq, a mailing list for Windows NT vulnerabilities and countermeasures, at www.ntbugtraq.com.

SANS (System Administration, Networking, and Security) Institute, a cooperative research and education organization with a mailing list, at www.sans.org.

SecurityFocus.com, a Web site designed to facilitate discussion on security-related topics, to create security awareness, and to provide the Internet's largest and most comprehensive database of security knowledge and resources to the public, at www.securityfocus.com. It also maintains the popular Bugtraq mailing list.

U.S. Department of Energy's Computer Incident Advisory Capability (CIAC), provides computer security services free of charge to employees and contractors of the Department of Energy, at ciac.llnl.gov.

Security Surveys and Reports

1996 Information Systems Security Survey, a summary of a security survey by WarRoom Research LLC, at www.warroomresearch.com/researchcollabor/infosecuritysurvey.htm.

1998 Annual Global Information Security Survey, Ernst & Young and Computerworld, at www.ey.com/aabs/isaas.

Accounts of Network Intruders

B. Cheswick, *An Evening with Berferd in Which a Cracker Is Lured, Endured, and Studied.* Bill Cheswick of AT&T chronicles the attacks of a cracker. This can be found at the Purdue University COAST Web site at www.cs.purdue.edu/coast/archive/data/categ40.html.

D. Farmer and W. Venema, *Improving the Security of Your Site by Breaking Into It.* This paper looks through the eyes of a potential intruder, illustrating that even seemingly harmless network services can become valuable tools in the search for a system's weak points. You can find a copy at the Advanced Laboratory Workstation System Web site at www.alw.nih.gov/Security/Docs/admin-guide-to-cracking.101.html.