CHAPTER 1

Security Threats

In all but the smallest of computing environments, IT security has become a full-time job. System administrators must protect data, devices, and networks from a multitude of threats and vulnerabilities. And a day seldom passes without report of newly discovered threats or vulnerabilities. Security threats come in many forms — from malicious code to defaced Web sites. The damage inflicted through security breaches ranges from minimal and localized to widespread and devastating. As quickly as agile minds invent new ways to connect systems and data, others find dubious means of accessing and utilizing them.

The security administrator's job is to identify and patch system and network vulnerabilities before the bad guys use them for malicious purposes. He must also monitor systems and data for intrusions and improper access attempts. And he contends with these security needs on several levels: data, system, network, intranet, extranet, and Internet. The more users and devices added to the computing environment, the more work for the security administrator. After Internet connectivity is established, the work, the vulnerabilities, and the security threats multiply dramatically — when a system or network is connected to the Internet, it can potentially be accessed by anyone, from anywhere, at any time.

Fortunately, Windows Server 2003 administrators have a plethora of technology (NTSF, data encryption, IPSec, PPTP, and so on) for protecting all levels of the computing environment. In addition, auditing capabilities and security logs ease the burden of monitoring and reporting on intrusion attempts. Later chapters in this book provide specific information on configuring and using security tools and technologies. In this chapter, I explain the nature and origin of the security threats you are most likely to encounter both inside and outside the firewall.

In This Chapter

Defacements and Web vandalism

Espionage and data theft

Denial of Service

Internal versus external threats

Internal versus External Threats

Curiously, 80 percent of corporate security funds are spent on deterring external threats — access breaches by persons who are not authorized users on a network. But security experts estimate that 70 to 80 percent of all security incidents involve corporate insiders. Yet, perhaps because external security breaches are often highly publicized, companies focus their efforts on fighting external access attempts.

In this chapter, I introduce some of the means and methods used in internal versus external attacks. As a system administrator, you must address security threats from both inside and outside your network. Through reading this book, you will see that the processes and tools used for defending against both types of attack are often similar. A good security plan includes several layers of protection and secures resources and data from most unauthorized access — whether from corporate insiders or external attackers.

The key to getting a grasp on the security of your systems (although it sounds cliché) is to think like an attacker. If you can do this, you can better prioritize what needs to be secured and how much effort is required to do the job. Some systems may be simple to restore and may not contain any information of value to an attacker. In that case, it's better to focus your efforts somewhere else — verifying your off-site data storage, for example, and securing your Web servers against the latest worm. System security almost always involves compromises, and a good security administrator always knows how to prioritize.

Internal threat considerations

Internal threats pose the most serious problem to the security of a system or network. Often, an organization's servers and workstations reside on the same network behind a firewall, which gives the users of the workstations free access to attack the servers. In essence, a "trusted" user who already has some degree of access to a network initiates an internal attack.

Internal threats range from unskilled users to systems administrators — even the CTO. Unfortunately, every company employee must be considered a potential threat, but they cannot be treated with the same paranoia as external users. You must develop some level of perceived trust and provide convenient access for internal users, but also develop internal security. The convenience versus security debate has been raging as long as computers have existed.

Some internal security incidents are unintentional and result from carelessness, while others are the intentional work of a disgruntled employee. Still others are intentional attempts to profit from archived records — selling health and financial information, for example.

An internal user may attack a system for any number of reasons, including the following:

- ✦ Data theft
- ✦ Espionage
- ✦ Sabotage
- ✦ General malice

Curiosity about private information accounts for a very large number of internal security violations. Accounting records, such as payroll data, and e-mail are tempting targets. Most companies keep this information secured, and a curious employee becomes a network attacker when he or she decides to search for an answer to the question, "I wonder how much Joe makes?" In many cases, the curious insider discovers that he doesn't have access to the data and gives up. But others who don't have authorized access to the desired data will search for other means of access — cracking the password of a user who does have access, tricking an authorized user into revealing their password (commonly known as social engineering), or bribing an authorized user to provide the desired information.

A large number of internal attacks are initiated by disgruntled employees or former employees who want to get back at the company. This type of attacker takes many approaches:

- ✦ Deleting valuable corporate data
- ✦ Publishing or distributing private corporate data
- ✦ Changing administrative policy settings or passwords, or similarly disrupting the corporate network
- ✦ Sending offensive or inappropriate e-mail messages from the corporate messaging system

The level of damage inflicted by a trusted user largely depends on the user's computer and network skills. Those who have broader knowledge of the systems and network can literally halt the workflow of an entire company. While the scope of damage caused by a relatively unskilled user is not usually so large, the unskilled user can still cause serious harm.

The unskilled user

Consider the following situation:

One weekend, when the system administrators or tech support users are not in the office, an HR assistant decides to do some extra work. He attempts to access the payroll files, only to find that he doesn't have access. Either he is legitimately

denied access, or he may not have been added to a security group by accident. The HR assistant becomes incensed, especially because he commuted all the way to the office on a Sunday morning. The assistant goes directly to his manager on Monday and makes a complaint; the manager himself complains up the chain of management until eventually an edict comes down to remove the security on all files on the network: the result of uninformed decision-making.

This relaxing of security is a worst-case scenario for the systems administrators because when the payroll data is leaked by some user who wants to know Joe's take-home pay, it is the administrator's responsibility to explain how that happened. Of course, bad decision-making by management is never a valid excuse when those managers are the ones asking questions.

By planning a proper internal security architecture, you can have a secure internal network and still maintain a level of convenience for users. It may take some measure of skill, some trial and error in the lab (trial and error on the production network is a bad idea for numerous reasons), and even some luck to get everything balanced out. However, after everything is in place, many attacks will either be impossible to carry out or finding the attacker will be easy.

Unskilled internal users may attempt to find a variety of information. For example, they may try to access software that they would not normally have access to (to make copies for home, or for other uses), view other users' e-mail for malicious purposes, or steal company secrets. This type of user is typically easy to track and generally unsuccessful on the majority of attempts. Furthermore, they will be neither familiar with the network, nor have the expertise to use network discovery tools. To track the movements of the unskilled internal user and gather evidence, you can simply add your network monitor (you have a network monitor as part of your response kit, right?) and thereby monitor the user for purposes of discipline or prosecution.

The skilled user

The second type of user posing a threat to your network security is the skilled user, of which the best example is a system administrator. Typically, the skilled user doesn't attack the network for trivial purposes, but for malicious purposes such as espionage and selling company secrets, or worse, revenge. Tracking this type of user presents two major difficulties. A skilled user will likely be competent enough to cover his tracks. He may be removing or modifying system and security logs, and attacking from a variety of hosts; he probably knows the layout of the network, including the existence of any network monitoring and intrusion detection systems. Also, after you notice a problem and respond, the skilled user may be scared off before you gather enough information to identify and prosecute him. It is very important that the attacker is not aware of the monitoring systems.

It is possible, in a number of ways, to make a "blackened" network monitor. A blackened network monitor should be able to receive all data sent on the network but

not send any data onto the network. More difficult than creating the system that is logically blackened is physically hiding that system from the attacker, especially when the attacker has physical access to the data center. The attacker would notice something unusual, such as a new system or a strange connection to the network.

An internal network's layout should be configured so that users and servers are separated. The best design is one in which firewalls separate the two groups, but they should at least be on separate network subnets. More importantly, you should use an intrusion detection system (IDS) agent to watch traffic between the networks. With this layer of security, you can track intrusions from internal users.

Anatomy of an internal attack

The effect of an internal attack (by a skillful attacker) on a company can range from small inconvenience to irreparable damage. Consider, for example, that the attacker is someone who is involved with the systems on the network, such as a system administrator. As mentioned earlier in this chapter, the attacker can mount his attack for a number of reasons. Data theft is a fairly simple act; it is also undetectable in many cases — until your innovations start appearing on your competitor's products, or your competitor's sales force starts targeting your customers en masse.

Another motive for attacking a system — one where the effects may be less obvious than that of data theft, but just as damaging — is revenge. A disgruntled employee is probably the best example of an attacker with vengeful motives. He may be able to access the systems because he is given notice of his termination ahead of time, or he may be let go immediately but still have access for a brief period. In any case, he probably doesn't feel he has much to lose.

A well-planned attack will involve the theft of such items as backup tapes and system software, and possibly even hardware. By taking such items, which he knows are vital to operations, the attacker can disrupt service and possibly even cause total data loss. If nothing else drives home the importance of off-site backups, this situation should. After the attacker has removed all possibility for a quick restoration of service, he then begins the main attack against the network or data. Because he has already disabled the backup/restore devices, this portion of the attack may have devastating consequences. The attacker may have full access to many systems, so the attack could run the gamut from small and malicious to total network disablement. One of the simplest and most dangerous things that the attacker can do is format the hard disks. This makes it impossible to load the operating system or access the data stored on the hard disks.

This simple attack is complete. Data has been lost and the backups, operating system media, and application media are missing. The company has now suffered a potentially deadly attack. It will only be able to recover from it by starting from scratch.

A Network Configuration for Highly Important Data

Some government and military installations employ a network configuration that places a firewall between the user workstations and the servers. This isn't common practice in the private sector due to the difficulty of correctly configuring and maintaining the firewall to allow enough access—but not too much access. (One government administrator described such a firewall as resembling Swiss cheese when improperly configured.) This network configuration is used when the value and protection of the data on the network servers is of the utmost importance. Government, military, and financial institutions often demand this level of security.

External threat considerations

Most organizations spend a great deal of effort and money defending against external attackers, despite the fact that internal attacks are more common. Perhaps this is due to the intense publicity that some external attacks have generated. Perhaps it's because trusted employees (potential internal attackers) can't be further restricted from corporate data. Whatever the reason, more emphasis is placed on protecting systems and networks from external attackers, and internal security is often neglected as a result.

An external attack originates from outside a network's firewall. Depending on the location of your servers in the network architecture and the configuration of the firewalls at your network entry points, your servers may be vulnerable to attack from users who are not located on trusted networks under your control. The firewall separates your internal, private network from the external, public world—the Internet. In theory, users on the public networks know less about the configuration of your network and servers, so they would seem to be less of a threat; however, poor security can allow them to collect system and network information and create their own map of your systems.

Black hats, white hats, and script kiddies

External attackers' motives are usually different from those of the internal attacker. In rare cases, they are also looking for secrets and practicing espionage, but most of the time they are attempting simple Web page defacements or attacking for the thrill of it.

The most severe situation you will face involves an attacker looking for specific data on your network. This person is typically a lone wolf who is highly skilled and doesn't leave much of a trace, if any. This attacker doesn't share his attacks with anyone, reducing the likelihood of being caught (there's no "honor among thieves," as the saying goes). This type of attacker has mastered each system he intends to attack—he leaves no evidence of his presence in log files and does not destroy any

data on your systems, only making copies of whatever he is looking for. He spends much time in preparation before mounting the attack, which you may never even detect. This type of attacker may also be capable of finding new vulnerabilities and coding the associated exploits, or coding new exploits for old vulnerabilities. The best defense against this type of attacker is to keep up-to-date on all vulnerabilities in your systems and their respective fixes, have a secure firewall layer, and implement a good blackened intrusion detection system (discussed later in this chapter).

The attacker who attempts Web defacements (the *script kiddie*) is a much weaker enemy. His skill set is much smaller than that of the lone wolf, as he is only capable of using what is provided to him and doesn't write his own malicious code. He launches an attack from a previously compromised system on the Internet to compromise your systems for either a Web page defacement or a staging area for his next attack. To defend against the script kiddies, keep your systems patched and your firewalls locked down as tightly as possible.

Nowadays, with millions of people connected to the Internet, the threats have changed and multiplied. The expert attackers can be categorized as *white hat* and *black hat* hackers. The white hat hackers discover security problems in systems and then work with the software vendors to fix these problems. Many of these experts believe in full disclosure — that is, releasing the information about these problems both to the software vendors and to the general public through paths such as the Bugtraq mailing list. Normally, the vendors are contacted first and given time to address and repair the problems before information about the vulnerabilities is publicly released. The black hat hacker locates problems with systems, but does not release them to the vendor or to the general public, choosing instead to exploit them for his own gain.

White hat hackers do not pose a threat to network and system security. On the contrary, they work with the software vendors to make the software that is used on a daily basis more secure, and in situations where the vendor is unresponsive, they notify the general public so that the software can be avoided. The black hat hacker is a rare threat. Most black hat hackers (remember, they know the systems thoroughly) will use the security holes they find for their own gain, through industrial espionage or outright theft. Black hat hackers are rare; you are less likely to encounter one "in the wild" than you would another type of attacker, such as the script kiddie.

The phrase "in the wild" refers to something you come across on the public Internet rather than in a lab environment. It is commonly used to refer to particular viruses that are commonly seen on the Internet.

You are almost guaranteed to come across a script kiddie when administrating a network connected to the Internet. Script kiddies possess no real skills and use the exploits created by expert hackers to maliciously attack systems. Everyone in the Information Technology industry looks down upon script kiddies, who simply

install a copy of the Linux operating system, download a few exploits, and begin to attack systems. Most of the time, script kiddies use these attacks to facilitate Web page defacements as a way to flaunt their own skills. In most environments, it is common to see some form of attack from a script kiddie daily, even if it is only a port scan.

Human imperfection should also be considered in a discussion about security. Humans don't usually function well under stress, can be easily distracted, and may forget things. Furthermore, they become complacent when they perform the same procedures over and over again. When considering the individuals who will potentially come into contact with your network, you will probably decide that it's best not to trust anyone: internal users as well as anonymous external users. Don't take your own actions for granted either. Leaving a security hole in a system that you configure or maintain is not much different from an attacker installing a backdoor to give himself access. The process is similar, and the results are the same. Admittedly, comparing a system administrator to an attacker is an extreme way of thinking, but it guarantees the highest level of protection for your system.

A *backdoor* is a program installed without your knowledge that allows an attacker to re-enter a system that he has compromised. However, backdoor programs originated not with the malicious hackers, but from system developers and testers who left alternative ways into a system to facilitate their development and testing.

Anatomy of an external attack

The attack of an intruder who does not have knowledge of your network will progress in one of two ways. If the intrusion is simple, such as the compromise of a system that is not behind a firewall, he may simply attack that machine and not bother exploring any further. A more determined attacker, however, is willing to do more work. Keep in mind that the attack outlined in this section is possible with a firewall in place, although some firewalls can be configured to stop parts of the attack.

The determined attacker first attempts to get more information about your network while avoiding detection. He can obtain this information from a number of places. Protocols such as Simple Network Management Protocol (SNMP) can be queried using default passwords to get information about router and firewall configurations. Some network devices also have other vulnerabilities and will divulge information when queried the right way. These types of exploits provide the attacker with the network layout, including the following valuable information:

- ✦ IP addresses
- ✦ MAC addresses
- ✦ Routing information

The attacker can then begin to determine what systems exist on these networks. The most common method of finding systems is by using a port scanner such as nmap. nmap is discussed in more detail later in the book, but essentially, it scans an IP address or range of IP addresses and checks each port for a response. A number of methods can be used to check open ports, the simplest of which is to send a TCP packet and check for a response. Many of these methods are detectable by firewalls and IDSs, but some methods of scanning, known as stealth scans, are not detectable. Figure 1-1 shows a stealth port scan using nmap showing the open ports on a target system.

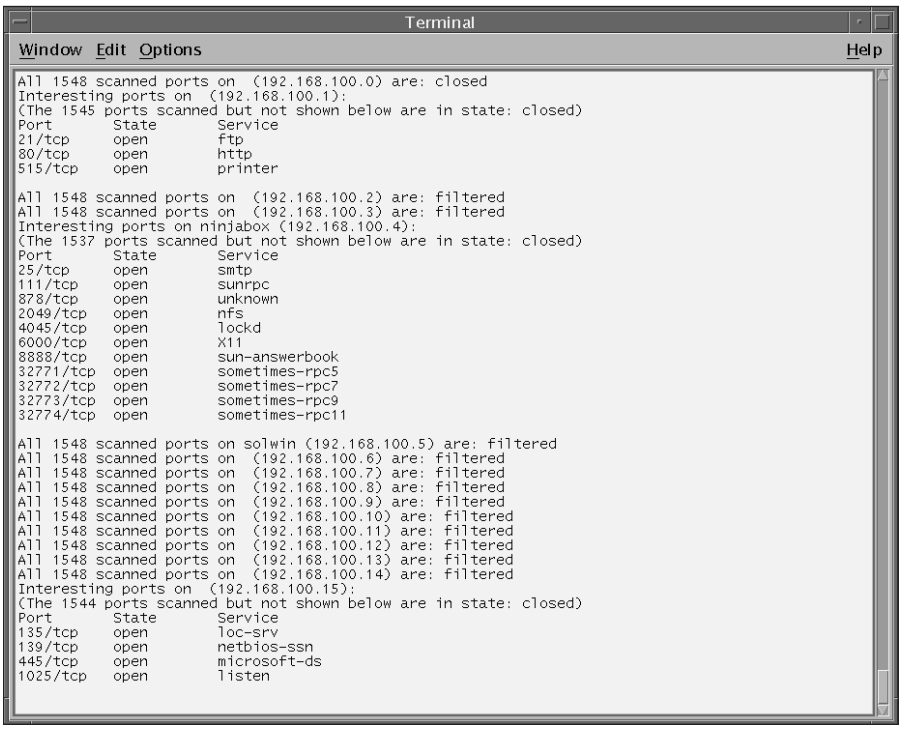


Figure 1-1: nmap is used to find the open ports on a target system.

At this point, the attacker possesses a large amount of information about the network. He queried your network devices to determine the network layout and mapped the systems with open ports. The attacker now knows what to attack, but not how to mount the attack. To determine which vulnerabilities exist and which exploits he can use, he must first determine what type of operating system is on each target machine. The best way he can do this and remain undetected is through deduction. Certain services run by default on particular operating systems. For

example, a system with NetBIOS ports (137–139) or Microsoft Exchange–related ports open is most likely a Microsoft Windows system, and so on. You can configure a system in many ways so that the default ports are closed, and software configurations exist that make one type of system look like another, such as a UNIX system running the Samba package. The attacker can also use other less stealthy methods of determining the type of system. A number of applications, such as nmap (if not using stealth scanning) and QueSO, are capable of looking at signatures present in the Transmission Control Protocol (TCP) stack of the target, as well as open ports, and determining the type of operating system (this is known as stack fingerprinting).

QueSO is a freeware network utility used to detect the operating system running on a host machine. QueSO works by sending a faulty TCP packet to the targeted computer. A Unix system and a Windows system respond differently to malformed TCP packets, so the response to the bad packet reveals the operating system of the targeted computer. Interestingly, the name QueSO comes from the Spanish phrase "Que Sistema Operativo?" which loosely translates to "What is the operating system?"

Nmap — or Network Mapper — is another freeware utility that is used for both port scanning and stack fingerprinting. Nmap is considered more powerful than QueSO for several reasons: it can be used for port scanning while QueSO cannot, and it can be used to scan a range of IP addresses while QueSO scans only a single address.

The nmap utility includes a variety of options (or switches). For example, the –O switch (typed nmap –O) is used to perform stack fingerprinting, while the –p switch designates which port is to be scanned.

QueSO and nmap are valuable administrative tools used by legitimate administrators. Unfortunately, they're also used by less principled people who use them as reconnaissance tools for gathering information about a system or network prior to an attack.

Figure 1-2 shows nmap being used to determine the operating system type with the –O switch.

Finally, the attacker has everything he needs to carry out the attack: the network layout, and a map of all the systems, including their operating systems and running software with open ports. He can now determine which tactics to use and begin to attack systems on the network. All he has to do now is compromise one internal system. One easy and commonly used exploit for the attacker is to run a program that inserts an access point by which he can freely access the system. A thorough attack will progress to using the compromised machine (or machines) to gather information about the internal network, finding all of the information about every system. The attacker essentially has free reign over the entire network now that he has a steppingstone behind the firewall.

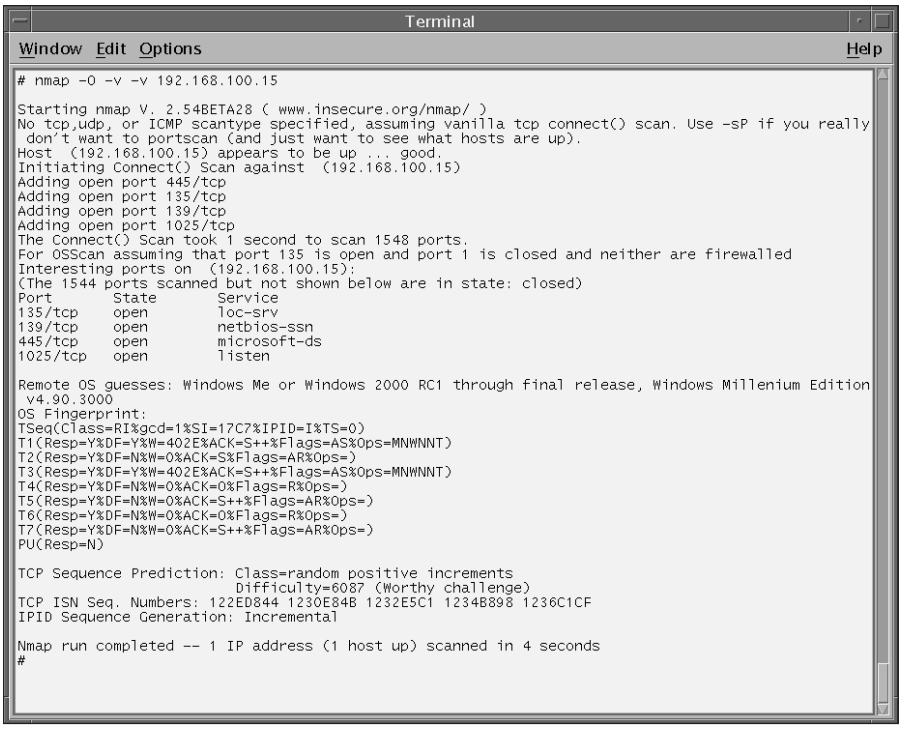


Figure 1-2: nmap can be used to determine the type of operating system on a target system through stack fingerprinting.

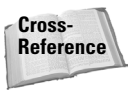
Anyone with a little time, a moderate amount of knowledge, a lot of determination, and little or no fear of getting caught can carry out this type of attack. With the proper pieces in place, you can prevent such attacks and limit damage if the network is compromised.

Internal versus External Security Measures

System security is widely misunderstood and often overlooked. When a new security problem is discussed in the media, many computer users — including system administrators — fail to understand the problem and its repercussions. This book will help you understand potential security problems and how to prevent and deal with their symptoms.

To simplify, you can view an effective security plan as a series of roadblocks, each roughly equal in strength and importance. If the attacker is not stopped by the first roadblock, he will likely be stopped by the second or third.

To properly secure your systems, you must follow the same type of design. A multilayer security architecture is the most important design consideration. With proper network, system, and application security and monitoring, you are far less likely to suffer damage from an attack.



Chapter 2 discusses multilayer security architecture in detail.

System security measures can take two forms: *static* and *dynamic*. Static security measures include the installation and securing of the system, and application and development of procedures and documentation. These procedures do not change very often and when they do, the change is often minor. Dynamic security measures involve keeping up-to-date with vulnerabilities and the necessary changes required to offset them. Dynamic security is an ongoing task.

System administrators are now faced with many threats. In the past, many systems were on private networks and the only threats were internal to those networks that were tightly controlled, or through dial-up connections. As the Internet grew out of the connection of these networks to one another, the user base could not be controlled, increasing the need for greater system security. Even in the Internet's early stages, the attackers were highly trained experts who knew the operating systems (such as Unix, VMS, and MVS) inside and out.

External security measures

External threats are addressed somewhat differently from internal threats. While internal security measures focus on appropriately configuring access to data, external security measures focus on blocking access to the internal network and its systems.

External threats normally originate from attackers with no knowledge of the layout of your network (unless, of course, you're attacked from outside the firewall by a current or former employee). Before an external attacker can damage your systems or data, he must first find entry points and locate his targets on your systems or network. By configuring good security between your internal and external networks, you can limit the amount of information that an attacker can learn as well as stop many attacks at the network edge. One of the key elements to stopping external attackers is a well-secured firewall between your network and the outside. In this book, you'll learn to implement and maintain the correct security measures to protect your computer systems from an external attack.

Internal security measures

Beyond the Internet-type security provisions, more specific changes are needed to secure against an internal attacker. Many security provisions are built into Windows

Server 2003 and Windows XP, the most basic of which are Access Control Lists (ACLs). By using ACLs, you can block users from accessing files on the network that they are not supposed to access.

Group policies also allow for the specific configuration of security. You can restrict which systems or types of systems (workstations, domain controllers, and so on) that a user can log in to. You can also take more control of the desktop with the more advanced policy features. You can avoid having your internal users run unauthorized programs, which may introduce security vulnerabilities or be used to attack other internal systems.

The bottom line is that security is often more of a people problem than a system problem. Users often access files they shouldn't because they are snooping around. If you're viewing your security issues as purely technological problems, you're considering only a single element of a total security plan. Compromises can often be avoided by simply making users think that they'll be caught, and that the repercussions will be severe. This can be accomplished by writing and publishing a Computer and Network Security Policy. Some companies display a brief security statement when a user logs on to the network. Others require each employee to sign a document stating that they have read and understand the corporate security policy. Many companies do both.

Technological security comes into play for those who either don't know that what they're doing causes a security flaw, or don't care or believe that they'll be caught or that the consequences will be extreme.

Forms of Attack

There are probably as many ways to attack a computer system or network as there are reasons for attacking. With good reason, many people do not view the Internet as a safe place. They worry about using their credit card or having their personal information stolen. Take a minute to think about your attitude towards the Internet. Do you feel it is safe and has adequate security, or does its insecurity concern you?

The Internet is not inherently safe or unsafe. It all depends on the implementation of the services being used and the behavior of the user. For example, sending your credit card over a Secure Socket Layer (SSL) connection is a safe activity. E-mailing your credit card information in plain text is not.

The Internet — by design — is not secure. The protocols that are now used on the Internet were designed (with a few exceptions) in the days of *DARPAnet*, many years before the advent of the modern Internet. This early network of government, and later educational, sites is the birthplace of TCP/IP and many of the application

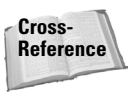
layer protocols now in use. Originally, these protocols were designed for a network with a restricted user base; only the staff of government and educational institutions would have access to these systems. Because the user base was assumed to be trustworthy, security was not designed into these protocols.

DARPAnet was a computer network created in 1969 that connected academic institutions, government agencies, and businesses engaged in computer research, Originally called ARPAnet and created by the Advanced Research Projects Agency (ARPA), the network was later renamed to DARPAnet when ARPA became the Defense Advanced Research Projects Agency. DARPAnet is commonly regarded as the precursor to the Internet.

The problem is becoming obvious. Since its inception, the Internet has been a series of compromises. Insecure protocols were used and protocols were modified to perform in ways that were not originally intended. For example, the layout of DARPAnet and many of the protocols used by DARPAnet were modified to allow public access, resulting in the creation of the Internet. As the Internet began to grow, additional modifications were made. During the early 1990s, the growth of the Internet exploded. The Internet protocols are now straining to handle the load of the Internet, and are still being modified to keep up with its ongoing evolution.

These protocols are so ingrained into the Internet that they cannot be easily replaced with better solutions. We are forced to work with the existing protocols, securing them so that local systems are not vulnerable to attack.

Today, administrators encounter a variety of security threats that were probably never dreamed of in the days of DARPAnet, including Trojan horses, worms, logic bombs, and viruses. The complete list is long and overwhelming.



Chapter 2 discusses these forms of attack in detail.

Luckily, Microsoft Windows Server 2003 — the latest version in the evolution of Microsoft Windows — has become more secure over its many incarnations. Microsoft has strengthened the security of the operating system, and added additional security features in the network services as well. While numerous security flaws may still exist, as the traditional Internet protocols cannot be removed, many new security features have been implemented, such as IP Security (IPSec) and Kerberos.

Keep in mind, Windows Server 2003 is not intrinsically secure. It includes tools and settings that allow it to be secure, but the application of those tools and settings is left up to the administrator. If the administrator doesn't make the server secure, it won't be secure

A secure host or secure network can be connected to the Internet and still be considered safe. While the original, insecure Internet protocols remain a huge problem,

the problem can be overcome if you have the necessary knowledge and an excellent security plan.

The rest of this chapter provides a general overview of the more common threats to the security of your network. The next chapter is more thorough in its discussion of the actual methods used by attackers.

Defacements and Web vandalism

The most basic form of attack on a Web server is defacing a Web page. Web page defacement is also the most audacious type of attack, as the attacker usually uses the defaced page to brag about his skills (ironically, very little skill is required to deface a Web page). Sometimes, Web page defacements are politically motivated; activists (often referred to as "hacktivists") use them as a way to spread their message. A "war" of Web page defacements between crackers in the United States and China has recently broken out for this reason.

Web page defacements are common among attackers working in groups. One teenage member of an infamous group defaced the White House Web site and was later convicted for the attack. The same group also declared war on Iraq and China, but later retracted their declaration.

You can find an archive of defaced Web sites at `www.attrition.org/mirror/attrition/`. However, with so many Web site defacements occurring on a daily basis, the task of maintaining the archive became overwhelming and the site is no longer actively maintained.

Because only a single page needs to be replaced (the `index.html` file), an attacker does not even need full access to a system to perform a Web page defacement, depending on the configuration of the Web server software and the system. The simplicity of this most basic security attack stresses the importance of using a proper multilevel security architecture and keeping up on the latest security patches for your operating system and applications.

Web page defacements are often preventable. Many attacks are implemented through vulnerable Common Gateway Interface (CGI) scripts such as forums and other standard "canned" code that is used on Web sites. As this code is freely distributed, the vulnerabilities are often exposed and exploits created. Certain CGI scanners even search the Web for vulnerable sites. By simply running one of these scanners, a script kiddie can find an entire list of pages that he can deface using an exploit that has already been written. The process requires almost no effort on his part, other than that of running the scanner, running the exploit, and uploading the defaced page.

Many of these vulnerabilities are widely known. It is important that you check all the software you use on your Web site for updates to prevent yourself from becoming a victim of these attacks.

In addition to CGI vulnerabilities, Web servers are often vulnerable to many of the other types of exploits, such as buffer overflows. Buffer overflow was at the root of the Code Red worm infestation that attacked many of the Microsoft IIS servers on the Internet in the year 2001. The attack was performed through a known buffer overflow vulnerability in Microsoft Index Server, for which a fix had been released six months before the actual Code Red outbreak. The fact that there was plenty of time to apply the fix or make a configuration change to prevent an attack, but that the attack still had such devastating consequences, indicates that security is not understood or taken seriously in many organizations.

Web page defacements are also one of the most obvious types of attack. While data theft can sometimes go completely undetected and Denial of Service (DoS) attacks can look like other types of traffic, Web defacements are immediately apparent.

Espionage and data theft

System and network attacks also result from industrial espionage and data theft. Almost every company possesses confidential and business-critical data. Proprietary information can be damaging to a company if it ends up in the wrong hands — their competitors', for example. Data theft can involve stealing files or e-mail messages.

The theft of proprietary files is somewhat difficult when good security practices are in place; stealing e-mail is a much easier task, because it is usually sent unencrypted across public networks when it is not intended for local recipients.

See Chapter 8 for information on encrypting data.

Stealing files from protected systems is quite difficult. Normally, the systems where important files exist are protected behind a firewall and are not externally accessible. For internal security, important files are password protected. This lulls a user into a false sense of security. Any time you have an externally accessible system on the same network as an internal system, it can be used as a jumping off point for attacks on the internal system.

Files can be stolen in a number of ways. The most obvious way is an attack on your network. If one of your network systems is accessible from the Internet, even a Web server, it is vulnerable to attack. If the attacker manages to control that particular system, he can use it as a staging point for deeper penetration into the network. If the system is not kept strictly separate from the rest of the network, the attacker can learn the layout of the internal network and access internal systems from the initial compromised system. In the worst-case scenario, the attacker could access all your network systems. This scenario clearly outlines one of the downfalls of the traditional Microsoft Windows domain architecture of trusted authentication, where a user logged into one system on the domain has a token

allowing him access to any system in the domain with his assigned privilege level. While convenient for internal users, this practice is highly insecure in the case of an attack where a domain member is compromised. By using exploits that elevate the privileges of the user, the attacker can gain administrator access to the entire domain.

Internal users have an even easier time with data theft. They can use a similar process to the one described in the preceding paragraph, with the additional advantage of being able to access the entire domain (if that architecture is in place). By finding a method to elevate their privilege level, they basically have free reign over the entire network. Without proper security implemented across the entire network, users can get any data they want.

Stealing e-mail is much simpler than stealing files. E-mail is transmitted in plain text (in most cases) and can be sniffed any time the attacker has access to any network segment over which the mail is transmitted, the sending or receiving mail servers, or the client system. As mail travels over many parts of the Internet that are not under your control, it is highly vulnerable. This reinforces the importance of using encryption for important information.

Sniffing traffic is the practice of looking at all traffic flowing over a network connection. Normally, only the traffic destined for your system will reach it but sniffing allows you to see all traffic.

The danger of data theft and industrial espionage shows the importance of a proper multilevel security architecture from the firewall (ingress and egress points) to the client (encryption).

Denial of Service attacks

An entirely different class of attacks is known as *Denial of Service (DoS) attacks*. Instead of trying to gain access to systems (for various reasons), an attacker attempts to stop others from having normal access to the system(s) under attack.

An attacker can mount a DoS attack in a number of ways. Normally, he uses known DoS exploits in a piece of software or hardware. Many DoS attacks are performed by sending a certain type of network packet to the vulnerable system. For example, Windows 95 and Windows NT (before patches) were vulnerable to an attack called WinNuke, which is performed by sending Out of Band data to port 139 (a NetBIOS port). During a WinNuke attack, the Out of Band data occurs when the URGENT flag in the TCP header is set and the URGENT POINTER is set to the end of the frame where no more normal data occurs. This attack causes the victim machine to suffer a Blue Screen of Death (BSoD) on Windows NT (requiring a reboot), or an exception error on Windows 95, which would cause the system to stop network communications.

Many DoS attacks are found in the wild, and can attack everything from Windows 95 workstations to the large routers that run the Internet. It is important to use one of the available resources, such as the Bugtraq mailing list, to keep up-to-date on new DoS attacks to which your systems may be vulnerable. Responsive vendors usually release patches soon after the vulnerabilities are exposed, but some vendors may be less responsive, allowing vulnerabilities to exist for weeks or months.

Bugtraq is a discussion list focusing on computer security vulnerabilities and how to fix them. To subscribe to Bugtraq, simply send an e-mail to `bugtraq-subscribe@securityfocus.com`. Your message does not require any special text or subject line.

The *Distributed Denial of Service attack (DDoS)* has been used to bring down major Web sites such as Yahoo! and eBay. The steps required to perform a DDoS are fairly simple, as the necessary software has already been written. The attacker first compromises a number of systems on the Internet, installing DDoS agents on each machine. For the best performance of a DDoS, the DDoS agents should be on different network segments, preferably on different backbones. The attacker may also leave a root kit on the systems so he can access them later. (A root kit is a package of modified system files that are designed to hide an intruder's activities on your system. Using the files from the root kit, the intruder can gain access, deposit files, and run programs undetected.) A central management program then controls these agents remotely. Often, the central management point resides on an Internet Relay Chat (IRC) channel as a bot (a software agent) that the attacker can use to control the attack anonymously. The attacker launches the attack against a target from the central point and the agents begin sending spurious network traffic (usually UDP packets) to the target (often on port 80). This huge amount of traffic will bring down the target machine, and often the network link on which the target resides. Figure 1-3 shows the architecture of a DDoS attack.

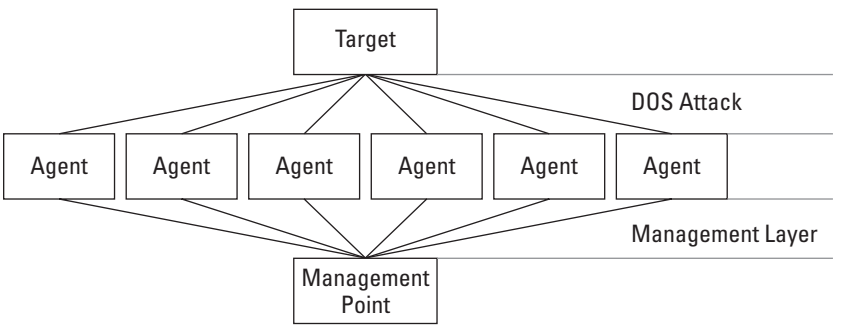


Figure 1-3: The architecture of a DDoS attack shows the agents and the central management point.

You cannot stop a DoS attack without a widely distributed effort between ISPs and backbone carriers. As the victim of a DoS, you can't do anything except try to block the traffic from the outermost network router that you control (usually an edge router, which is a router logically located at the outer edge of your network). Even after blocking the traffic, the router will suffer, as it is utilizing most of its resources just to discard packets.

DoS attacks can theoretically be stopped through a distributed joint effort between providers. With a simple DoS attack, the Internet Service Provider (ISP) providing service to the host originating the attack can cut off the network or dial-up connection from which the attack is originating. This type of procedure is simple enough to coordinate with the abuse department of the ISP in question, as long as the company is responsive. DoS attacks often originate from countries with lax computer security laws, and getting help is not an option.

To make a DoS attack even more complicated, what happens if the DoS originates from a system that has been compromised and does not belong to the attacker, such as a system at an educational or governmental institution? Since the ISP or carrier does not control the organization's internal network, it would have to disable access from the entire campus, which is obviously not an option. Another level of human contact may be brought in, the abuse department or network administrators at the organization where the attack is originating. This group may or may not be responsive and may not take responsibility for fixing the problem.

Furthermore, what happens if the attack is not a simple DoS from one system, but a DDoS from a number of systems distributed across the Internet? Your woes are now multiplied by the number of DDoS agents the attacker has employed. Each of these systems needs to be disabled. If they are on different carriers and in different organizations, you must deal with a large number of people to end the attack.

One final situation that has not yet occurred but easily could is that a really intelligent attacker could design a DDoS network that uses a larger number of agents spread widely across the Internet. The controller could dynamically change the source of the DDoS to any subset of these agents at any time. If the DDoS had a large enough pool of agents to use and was constantly changing, it would be nearly impossible to stop. Simply shutting down the network connection from one agent would not influence the overall effectiveness of the DDoS attack, as another agent would automatically take its place. It's probably just a matter of time before such an attack occurs.

Some companies are attempting to find solutions to their susceptibility to DoS attacks. Many of these solutions involve alliances with a number of companies working together to develop a product that will detect DDoS agents and stop them from attacking. The implementation of this product would most likely need to be

rolled out on many devices, as the attacks will be stopped locally. For example, the product would be deployed on firewalls or routers, and if the device detected the pattern of activity identifying a DoS attack from a local system, that system would be quarantined.

Summary

The information presented in this chapter is meant to be somewhat intimidating. Computer security is an extremely broad topic, and I want you to develop a healthy regard for its complexities. Administrators must know how to protect their data from both internal and external attackers, while still providing trusted users with access to the resources necessary to perform their jobs. While remaining vigilant to attack from any direction, administrators must also learn to recognize and defend their systems from a variety of attacks. A detailed security plan, covering both internal and external security requirements, is essential for any successful administrator.

In the following chapters of this book, you'll discover a variety of tools that will help you monitor, secure, and maintain your Windows Server 2003 computer systems and network. When properly implemented and maintained, the security features available in Windows Server 2003 provide a secure connected computing environment.

✦ ✦ ✦