Chapter 5

Malware

Chapter Objectives

After reading this chapter and completing the exercises, you will be able to do the following:

- Understand viruses (worms) and how they propagate, including the Sobig and Sasser types
- Have a working knowledge of several specific virus outbreaks
- Understand how virus scanners operate
- Understand what a Trojan horse is and how it operates
- Have a working knowledge of several specific Trojan horse attacks
- Grasp the concept behind the buffer-overflow attack
- Have a better understanding of spyware and how it enters a system
- Defend against each of these attacks through sound practices, antivirus software, and antispyware software

Introduction

In Chapter 4, "Denial of Service Attacks," we examined the denial of service attack. It is a very common attack and one that can easily be perpetrated. In this chapter, you will continue your examination of security threats by learning about several other types of attacks. First, you will learn about virus outbreaks. Our discussion will focus on crucial information about how and why virus attacks work, including their deployment through Trojan horses. This chapter is not a "how to create your own virus" tutorial, but rather an introduction to the concepts underlying these attacks as well as an examination of some specific case studies.

This chapter will also explore buffer-overflow attacks, spyware, and several other forms of malware. Each of these brings a unique approach to an attack, and each needs to be considered when defending a system. Your ability to defend against such attacks will be enhanced by expanding your knowledge of how they work. In the exercises at the end of the chapter, you will have the opportunity to research preventative methods for viruses and to try out antivirus methods from McAfee and Norton.

Viruses

By definition, a computer virus is a program that self-replicates. Generally, a virus will also have some other unpleasant function, but the self-replication and rapid spread are the hallmarks of a virus. Often this growth, in and of itself, can be a problem for an infected network. The last chapter discussed the Slammer virus and the effects of its rapid, high-volume scanning. Any rapidly spreading virus can reduce the functionality and responsiveness of a network. Simply by exceeding the traffic load that a network was designed to carry, the network may be rendered temporarily nonfunctional. The infamous I Love You virus actually had no negative payload, but the shear volume of emails it generated bogged down many networks.

How a Virus Spreads

A virus will usually spread primarily in one of two ways. The first is to simply scan your computer for connections to a network, then copy itself to other machines on the network to which your computer has access. This is actually the most efficient way for a virus to spread. However, this method requires more programming skill than other methods. The more common method is to read your email address book and email itself to everyone in your address book. Programming this is a trivial task, which explains why it is so common.

The latter method is, by far, the most common method for virus propagation, and Microsoft Outlook may be the one email program most often hit with such virus attacks. The reason is not so much a security flaw in Outlook as it is the ease of working with Outlook. All Microsoft Office products are made so that a legitimate programmer who is writing software for a business can access many of the application's internal objects and thereby easily create applications that integrate the applications within the Microsoft Office suite. For example, a programmer could write an application that would access a Word document, import an Excel spreadsheet, and then use Outlook to automatically email the resulting document to interested parties. Microsoft has done a good job of making this process very easy, for it usually takes a minimum amount of programming to accomplish these tasks. Using Outlook, it takes less than five lines of code to reference Outlook and send out an email. This means a program can literally cause Outlook itself to send emails, unbeknownst to the user. There are numerous code examples on the Internet that show exactly how to do this, free for the taking. For this reason, it does not take a very skilled programmer to be able to access your Outlook address book and automatically send emails. Essentially, the ease of programming Outlook is why there are so many virus attacks that target Outlook.

While the overwhelming majority of virus attacks spread by attaching themselves to the victim's existing email software, some recent virus outbreaks have used other methods for propagation, such as their own internal email engine. Another virus propagation method is to simply copy itself across a network. Virus outbreaks that spread via multiple routes are becoming more common.

The method of delivering a payload can be rather simplistic and rely more on end-user negligence than on the skill of the virus writer. Enticing users to go to websites or open files they should not is a common method for delivering a virus and one that requires no programming skill at all. Regardless of the way a virus arrives at your doorstep, once it is on your system, it will attempt to spread and, in many cases, will also attempt to cause some harm to your system. Once a virus is on your system, it can do anything that any legitimate program can do. That means it could potentially delete files, change system settings, or cause other harm.

Recent Virus Examples

The threat from virus attacks cannot be overstated. While there are many web pages that give virus information, in my opinion, there are only a handful of web pages that consistently give the latest, most reliable, most detailed information on virus outbreaks. Any security professional will want to consult these sites on a regular basis. You can read more about any virus, past or current, at the following websites:

- www.f-secure.com/virus-info/virus-news/
- www.cert.org/nav/index_red.html
- http://securityresponse.symantec.com/
- http://vil.nai.com/vil/

The sections below will look at a few recent virus outbreaks (and some not so recent) and review how they operated and what they did.

W32/Netsky-P

This worm was first found in 2006 but is still going around in 2011. It is a fairly typical virus in that it spreads primarily through email, but also uses file sharing utilities to copy itself. It also copies itself to various directories and shared folders. In one interesting twist, it attempts to copy itself to C:\WINDOWS\FVProtect.exe. The name would make many people (including otherwise technically savvy people) think this program was actually part of some antivirus utility. It also copies itself to C:\WINDOWS\userconfig9x.dll. Again, it would appear to be a system file, thus making people less likely to delete it.

This is also a classic worm/virus in that the email it sends out had a fairly generic title and content that attempts to get the recipient to open the attachment. For example, the body of the message might say something like "Please see the attached file for details" or "Your file is attached."

Troj/Invo-Zip

This particular worm is a classic worm/Trojan horse. It was first reported in mid 2010. It is transmitted as a zip file attached to an email. The email claims that the zip file contains data related to an invoice, tax issue, or similar urgent paperwork. This is a classic example of attempting to entice the recipient to open the attachment. And in this case, the recipients most likely to be enticed would be businesspeople.

If the recipient does open the attachment, then he or she will have installed spyware on his machine that would first disable the firewall then start attempting to capture information including financial data, and even taking screenshots of the user's desktop.

MacDefender

This virus is very interesting for multiple reasons. First because it specifically targets Macintosh computers. Most experts have long agreed that Apple products remained relatively virus free simply because their products did not have enough market share to attract the attention of virus writers. It has long been suspected that if Apple garnered a greater market share, they would also begin to get more virus attacks. That has proven to be true.

This virus was first seen in the early months of 2011. It is embedded in some web pages and when a user visits those web pages, he or she is given a fake virus scan that tells the user that they have a virus and it needs to be fixed. The "fix" is actually downloading a virus. The point of the virus is to get end users to purchase the MacDefender "antivirus" product. This is the second reason this case is noteworthy. Fake antivirus attacks, also known as scareware, have been becoming increasingly common.

The Sobig Virus

This is obviously not a recent virus, as it was first found in 2003. However, it is an excellent virus to study. This virus is important to study because it received the most media attention and perhaps caused the most harm in 2003.The first interesting thing to study about this virus was how it spread. It spread utilizing a multimodal approach to spreading. This means that it used more than one mechanism to spread and infect new machines. It would copy itself to any shared drives on your network and it would email itself out to everyone in your address book. For these reasons, this virus was particularly virulent, and this is also why it is important to study.

FYI: Virulent Virus

The term *virulent* means essentially the same thing in reference to a computer virus as it does to a biological virus. It is a measure of how rapidly the infection spreads and how easily it infects new targets.

In the case of Sobig, if one person on a network was unfortunate enough to open an email containing the virus, not only would his machine be infected, but so would every shared drive on that network to

which this person had access. However, Sobig, like most email-distributed virus attacks, had tell-tale signs in the email subject or title that could be used to identify the email as one infected by a virus. The email would have some enticing title such as "here is the sample" or "the document" to encourage you to be curious enough to open the attached file. The virus would then copy itself into the Windows System directory.

This particular virus spread so far and infected so many networks that the multiple copying of the virus alone was enough to bring some networks to a standstill. This virus did not destroy files or damage the system, but it generated a great deal of traffic that bogged down the networks infected by it. The virus itself was of moderate sophistication. Once it was out, however, many variants began to spring up, further complicating the situation. One of the effects of some variants of Sobig was to download a file from the Internet that would then cause printing problems. Some network printers would just start printing junk. The Sobig.E variant would even write to the Windows Registry, causing itself to be in the computer startup (F-Secure, 2003). These complex characteristics indicate that the creator knew how to access the Windows Registry, access shared drives, alter the Windows startup, and access Outlook.

This brings up the issue of virus variants and how they occur. In the case of a biological virus, mutations in the genetic code cause new virus strains to appear, and the pressures of natural selection allow some of these strains to evolve into entirely new species of viruses. Obviously, the biological method is not what occurs with a computer virus. With a computer virus, what occurs is that some intrepid programmer with malicious intent will get a copy of a virus (perhaps her own machine becomes infected) and will then reverse-engineer it. Since many virus attacks are in the form of a script attached to an email, unlike traditionally compiled programs, the source code of these attacks is readily readable and alterable. The programmer in question then simply takes the original virus code and introduces some change, then re-releases the variant. Frequently, the people who are caught for virus creation are actually the developers of the variants who lacked the skill of the original virus writer and therefore were easily caught.

The Mimail Virus

This is another older virus that is still worth studying. The Mimail virus did not receive as much media attention as Sobig, but it had its intriguing characteristics. This virus not only collected email addresses from your address book, but also from other documents on your machine (Gudmundsson, 2004). Thus, if you had a Word document on your hard drive and an email address was in that document, Mimail would find it. This strategy meant that Mimail would spread farther than many other viruses. Mimail had its own built-in email engine, so it did not have to "piggy back" off your email client. It could spread regardless of what email software you used.

These two variations from most virus attacks made Mimail interesting to people who study computer viruses. There are a variety of techniques that allow you to programmatically open and process files on your computer; however, most virus attacks do not employ them. The scanning of the document for email addresses indicates a certain level of skill and creativity on the part of the virus writer. In this author's opinion, Mimail was not the work of an amateur, but rather a person with professional-level programming skill.

The Bagle Virus

This is the last of the "historical viruses" that we will examine. It is noteworthy in that it combined email attachments along with the fake virus warning. The Bagle virus began to spread rapidly in the fourth quarter of 2003. The email it sent claimed to be from your system administrator. It would tell you that your email account had been infected by a virus and that you should open the attached file to get instructions. Once you opened the attached file, your system was infected. This virus was particularly interesting for several reasons. To begin with, it spread both through email and copying itself to shared folders. Second, it could also scan files on your PC looking for email addresses. Finally, it would disable processes used by antivirus scanners. In biological terms, this virus took out your computer's "immune system." The disabling of virus scanners is a new twist that indicates at least moderate programming skills on the part of the virus creator.

A Nonvirus Virus

Another new type of virus has been gaining popularity in the past few years, and that is the "nonvirus virus" or, put simply, a hoax. Rather than actually writing a virus, a hacker sends an email to every address he has. The email claims to be from some well-known antivirus center and warns of a new virus that is circulating. The email instructs people to delete some file from their computer to get rid of the virus. The file, however, is not really a virus but part of a computer's system. The jdbgmgr.exe virus hoax used this scheme (Rhode Island Soft Systems, Inc., 2003). It encouraged the reader to delete a file that was actually needed by the system. Surprisingly, a number of people followed this advice and not only deleted the file, but promptly emailed their friends and colleagues to warn them to delete the file from their machines.

FYI: The Morris Internet Worm

The Morris worm was one of the first computer worms ever to be distributed over the Internet. And it was certainly the first to gain any significant media attention.

Robert Tappan Morris, Jr., then a student at Cornell University, wrote this worm and launched it from an MIT system on November 2, 1988. Morris did not actually intend to cause any damage with the worm. Instead, he wanted the worm to reveal bugs in the programs it exploited in order to spread. However, bugs in the code allowed an individual computer to be infected multiple times, and the worm became a menace. Each additional "infection" spawned a new process on the infected computer. At a certain point, the high number of processes running on an infected machine slowed down the computer to the point of being unusable. At least 6,000 UNIX machines were infected with this worm.

Morris was convicted of violating the 1986 Computer Fraud and Abuse Act and was sentenced to a $10,000 fine, 3 years probation, and 400 hours of community service. But perhaps the greatest impact of this worm was that it led to the creation of the Computer Emergency Response Team (CERT). CERT is an organization hosted at Carnegie Mellon University (www.cert.org/) that is a repository for security bulletins, information, and guidelines. CERT is a source that any security professional should be familiar with.

Rules for Avoiding Viruses

You should notice a common theme with all virus attacks (except the hoax), which is that they want you to open some type of attachment. The most common way for a virus to spread is as an email attachment. This realization leads to some simple rules that will drastically reduce the odds of becoming infected with a virus.

- Use a virus scanner. McAfee and Norton (explored in the exercises at the end of this chapter) are the two most widely accepted and used virus scanners. However, Kaspersky and AVG are also good, reputable choices. Each costs about $30 per year to keep your virus scanner updated. Do it. Now each antivirus has its proponents and detractors—I won't delve into the opinions on which is better. For most users, any of the four major antivirus programs would be effective. I personally rotate which one I use periodically, just so I can stay familiar with all of them.
- If you are not sure about an attachment, do not open it.
- You might even exchange a code word with friends and colleagues. Tell them that if they wish to send you an attachment, they should put the code word in the title of the message. Without seeing the code word, you will not open any attachment.
- Do not believe "security alerts" that are sent to you. Microsoft does not send out alerts in this manner. Check the Microsoft website regularly, as well as one of the antivirus websites previously mentioned.

These rules will not make your system 100% virus proof, but they will go a long way toward protecting your system.

Trojan Horses

Recall from earlier chapters that *Trojan horse* is a term for a program that looks benign but actually has a malicious purpose. We have already seen viruses that are delivered via a Trojan horse. You might receive or download a program that appears to be a harmless business utility or game. More likely, the Trojan horse is just a script attached to a benign-looking email. When you run the program or open the attachment, it does something else other than or in addition to what you thought it would. It might

- Download harmful software from a website.
- Install a key logger or other spyware on your machine.
- Delete files.
- Open a back door for a hacker to use.

It is common to find combination virus plus Trojan horse attacks. In those scenarios, the Trojan horse spreads like a virus. The MyDoom virus opened a port on your machine that a later virus, doomjuice, would exploit, thus making MyDoom a combination virus and Trojan horse.

A Trojan horse could also be crafted especially for an individual. If a hacker wished to spy on a certain individual, such as the company accountant, he could craft a program specifically to attract that person's attention. For example, if he knew the accountant was an avid golfer, he could write a program that computed handicap and listed best golf courses. He would post that program on a free web server. He would then email a number of people, including the accountant, telling them about the free software. The software, once installed, could check the name of the currently logged-on person. If the logged-on name matched the accountant's name, the software could then go out, unknown to the user, and download a key logger or other monitoring application. If the software did not damage files or replicate itself, then it would probably go undetected for quite a long time. There have been a number of Trojan horses through the years. One of the earliest and most widely known was Back Orifice.

FYI: Virus or Worm?

As noted in the previous chapter, there is disagreement among the experts as to the distinction between a virus and a worm. Some experts would call MyDoom (as well as Sasser, which will be discussed later) a worm because it spread without human intervention. However, I would define a virus as any file that can self replicate, and a worm is any program that can propagate without human interference. This is also the most common definition you will find among security experts.

Such a program could be within the skill set of virtually any moderately competent programmer. This is one reason that many organizations have rules against downloading *any* software onto company machines. I am unaware of any actual incident of a Trojan horse being custom tailored in this fashion. However, it is important to remember that those creating virus attacks tend to be innovative people.

It is also important to note that creating a Trojan horse does not require programming skill. There are free tools on the Internet, such as EliteWrapper, that allow one to combine two programs together, one hidden and one not. So one could easily take a virus and combine it with, for example, a poker game. The end user would only see the poker game, but when it was run it would launch the virus.

Another scenario to consider is one that would be quite devastating. Without divulging programming details, the basic premise will be outlined here to illustrate the grave dangers of Trojan horses. Imagine a small application that displays a series of unflattering pictures of Osama Bin Laden. This application would probably be popular with many people in the United States, particularly people in the military, intelligence community, or defense-related industries. Now assume that this application simply sits dormant on the machine for a period of time. It need not replicate like a virus because the computer user will probably send it to many of his associates. On a certain date and time, the software connects to any drive it can, including network drives, and begins deleting all files. If such a Trojan horse were released "in the wild," within 30 days it would probably be shipped to thousands, perhaps millions, of people. Imagine the devastation when thousands of computers begin deleting files and folders.

This scenario is mentioned precisely to frighten you a little. Computer users, including professionals who should know better, routinely download all sorts of things from the Internet, such as amusing flash videos and cute games. Every time an employee downloads something of this nature, there is the

chance of downloading a Trojan horse. One need not be a statistician to realize that if employees continue that practice long enough, they will eventually download a Trojan horse onto a company machine. If so, hopefully the virus will not be as vicious as the theoretical one just outlined here.

The Buffer-Overflow Attack

You have become knowledgeable about a number of ways to attack a target system: denial of service, virus, and Trojan horse. While these attacks are probably the most common, they are not the only methods. Another method of attacking a system is called a buffer-overflow (or buffer-overrun) attack. A buffer-overflow attack happens when one tries to put more data in a buffer than it was designed to hold (searchSecurity.com, 2004a). Any program that communicates with the Internet or a private network must take in some data**.** This data is stored, at least temporarily, in a space in memory called a buffer. If the programmer who wrote the application was careful, when you try to place too much information into a buffer, that information is then either simply truncated or outright rejected. Given the number of applications that might be running on a target system and the number of buffers in each application, the chances of having at least one buffer that was not written properly are significant enough to cause any prudent person some concern.

Someone who is moderately skilled in programming can write a program that purposefully writes more into the buffer than it can hold. For example, if the buffer can hold 1024 bytes of data and you try to fill it with 2048 bytes, the extra 1024 bytes is then simply loaded into memory. If that extra data is actually a malicious program, then it has just been loaded into memory and is thus now running on the target system. Or, perhaps the perpetrator simply wants to flood the target machine's memory, thus overwriting other items that are currently in memory and causing them to crash. Either way, the buffer overflow is a very serious attack.

Fortunately, buffer-overflow attacks are a bit harder to execute than a DoS or simple Microsoft Outlook script virus. To create a buffer-overflow attack, you must have a good working knowledge of some programming language (C or C++ is often chosen) and understand the target operating system/application well enough to know whether it has a buffer overflow weakness and how that weakness might be exploited.

It must be noted that modern operating systems and web servers are not susceptible to this attack. Windows 95 was quite susceptible, but it has been many years since a Windows operating system was susceptible. Certainly Windows 7 cannot be compromised with a buffer overflow. However, the same cannot be necessarily said for all the custom applications developed to run on various systems. It is always possible that an Internet-enabled application, including but not limited to web applications, might be susceptible to this attack.

Essentially, this vulnerability only exists if programmers fail to program correctly. If all programs truncate extra data, then a buffer overflow cannot be executed on that system. However, if the program does not check the boundaries of variables and arrays, and allows excess data to be loaded, then that system is vulnerable to a buffer overflow.

The Sasser Virus/Buffer Overflow

This is an older attack, but one that demonstrates the use of a buffer-overflow attack. Sasser is a combination attack in that the virus (or worm) spreads by exploiting a buffer overrun.

The Sasser virus spreads by exploiting a known flaw in a Windows system program. Sasser copies itself to the Windows directory as avserve.exe and creates a Registry key to load itself at startup. In that way, once your machine is infected, you will start the virus every time you start the machine. This virus scans random IP addresses, listening on successive TCP ports starting at 1068 for exploitable systems—that is, systems that have not been patched to fix this flaw. When one is found, the worm exploits the vulnerable system by overflowing a buffer in LSASS.EXE, which is a file that is part of the Windows operating system. That executable is a built-in system file and is part of Windows. Sasser also acts as an FTP server on TCP port 5554, and it creates a remote shell on TCP port 9996. Next, Sasser creates an FTP script named cmd.ftp on the remote host and executes that script. This FTP script instructs the target victim to download and execute the worm from the infected host. The infected host accepts this FTP traffic on TCP port 5554. The computer also creates a file named win.log on the C: drive. This file contains the IP address of the localhost. Copies of the virus are created in the Windows System directory as #_up.exe. Examples are shown here:

- c:\WINDOWS\system32\12553_up.exe
- c:\WINDOWS\system32\17923_up.exe
- c:\WINDOWS\system32\29679_up.exe

A side effect of this virus is that it causes your machine to reboot. A machine that is repeatedly rebooting without any other known cause may well be infected with the Sasser virus.

This is another case in which the infection can easily be prevented by several means. First, if you update your systems on a regular basis, your systems should not be vulnerable to this flaw. Second, if your network's routers or firewall block traffic on the ports mentioned (9996 and 5554), you will then prevent most of Sasser's damage. Your firewall should only allow in traffic on specified ports; all other ports should be shut down. In short, if you as the network administrator are aware of security issues and are taking prudent steps to protect the network, your network will be safe. The fact that so many networks were affected by this virus should indicate that not enough administrators are properly trained in computer security.

Spyware

In Chapter 1, "Introduction to Computer Security," spyware was mentioned as one of the threats to computer security. Using spyware, however, requires a great deal more technical knowledge on the part of the perpetrator than some other forms of malware. The perpetrator must be able to develop spyware for the particular situation or customize existing spyware for his needs. He must then be able to get the spyware on the target machine.

Spyware can be as simple as a cookie used by a website to record a few brief facts about your visit to that website, or spyware could be of a more insidious type, such as a key logger. Recall from Chapter 1 that key loggers are programs that record every keystroke you make on your keyboard; this spyware then logs your keystrokes to the spy's file. The most common use of a key logger is to capture usernames and passwords. However, this method can capture every username and password you enter and every document you type, as well as anything else you might type. This data can be stored in a small file hidden on your machine for later extraction or sent out in TCP packets to some predetermined address. In some cases, the software is even set to wait until after hours to upload this data to some server or to use your own email software to send the data to an anonymous email address. There are also some key loggers that take periodic screenshots from your machine, revealing anything that is open on your computer. Whatever the specific mode of operation, spyware is software that literally spies on your activities on a particular computer.

Legal Uses of Spyware

There are some perfectly legal uses for spyware. Some employers have embraced such spyware as a means of monitoring employee use of company technology. Many companies have elected to monitor phone, email, or web traffic within the organization. Keep in mind that the computer, network, and phone systems are the property of the company or organization, not of the employee. These technologies are supposedly only used for work purposes; therefore, company monitoring might not constitute any invasion of privacy. While courts have upheld this monitoring as a company's right, it is critical to consult an attorney before initiating this level of employee monitoring as well as to consider the potential negative impact on employee morale.

Parents can also elect to use this type of software on their home computer to monitor the activities of their children on the Internet. The goal is usually a laudable one—protecting their children from online predators. Yet, as with employees in a company, the practice may illicit a strong negative reaction from the parties being spied upon (namely, their children). Parents have to weigh the risk to their children versus what might be viewed as a breach of trust.

How Is Spyware Delivered to a Target System?

Clearly, spyware programs can track all activity on a computer, and that information can be retrieved by another party via a number of different methods. The real question is this: How does spyware get onto a computer system in the first place? The most common method is a Trojan horse. It is also possible that when you visit a certain website spyware may download in the background while you are simply perusing the website. Of course, if an employer (or parent) is installing the spyware, it can then be installed non-covertly in the same way that an organization would install any other application.

Obtaining Spyware Software

Given the many other utilities and tools that have been mentioned as available from the Internet, you probably will not be surprised to learn that you can obtain many spyware products for free, or at very low cost, on the Internet. You can check the Counterexploitation (www.cexx.org) website, shown in

Figure 5.1, for a lengthy list of known spyware products circulating on the Internet and for information about methods one can use to remove them. The Spyware Guide website (SpywareGuide, 2004) (www.spywareguide.com) lists spyware that you can get right off the Internet should you feel some compelling reason to spy on someone's computer activities. Figure 5.2 shows the categories of malware that are available from this site. Several key logger applications are listed on this site, as shown in Figure 5.3. These applications include well-known key loggers such as Absolute Keylogger, Tiny Keylogger, and TypO. Most can be downloaded for free or for a nominal charge from the Internet.

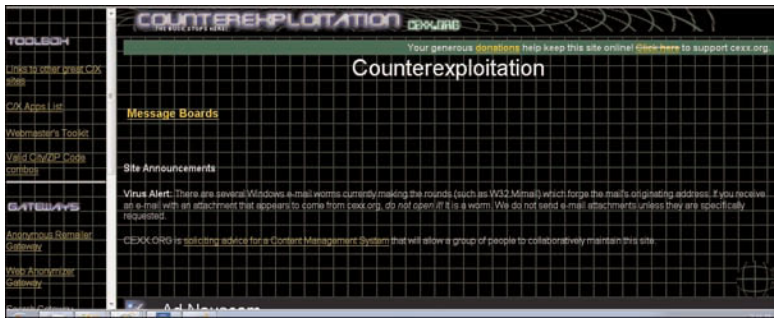

FIGURE 5.1 Counterexploitation website.

FIGURE 5.2 Malware categories at the Spyware Guide website.

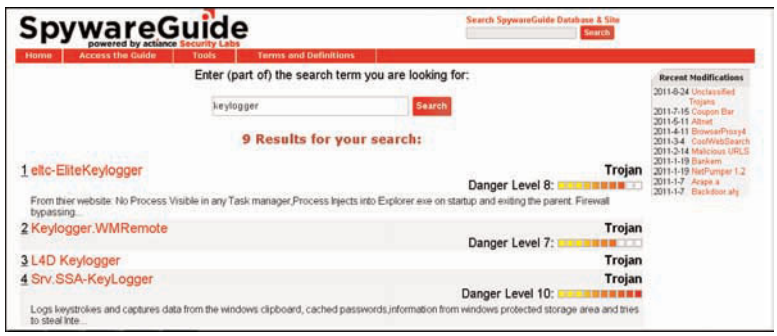

FIGURE 5.3 List of key loggers available through the Spyware Guide website.

Some well-known Trojan horses are also listed at this site (as shown in Figure 5.4), such as the 2nd Thought application that downloads to a person's PC and then blasts it with advertisements. This particular piece of spyware is one that downloads to your PC when you visit certain websites. It is benign in that it causes no direct harm to your system or files, nor does it gather sensitive information from your PC. However, it is incredibly annoying as it inundates your machine with unwanted ads. This sort of software is often referred to as adware. Frequently, these ads cannot be stopped by normal protective pop-up blockers because the pop-up windows are not generated by a website that you visit but rather by some rogue software running on your machine. Pop-up blockers only work to stop sites you visit from opening new windows. Websites use well-known scripting techniques to cause your browser to open a window, and pop-up blockers recognize these techniques and prevent the ad window from opening. However, if the adware launches a new browser instance, it bypasses the pop-up blocker's function.

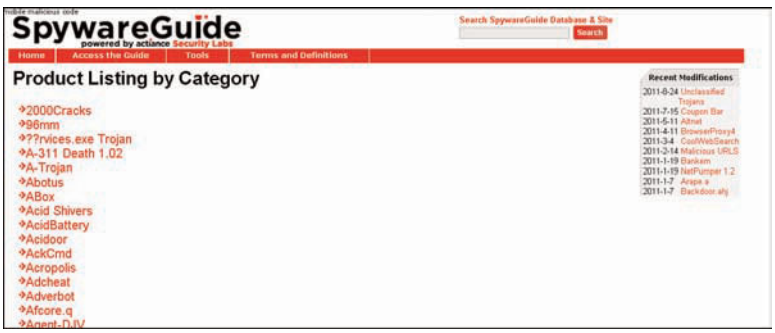

FIGURE 5.4 Trojan horses available at the Spyware Guide website.

Other Forms of Malware

In this and preceding chapters, the most prominent forms of malware have been discussed. There are, however, many other forms of attack. It is beyond the scope of this book to explore each of these, but you should be aware of the existence of these other forms of malware. Simply being aware can go a long way toward enabling you to defend your system efficiently. This section will touch upon just a few other forms of malware. You should reference the websites discussed in the end of chapter exercises and projects often so that you can stay up-to-date with all current forms of attack and defenses.

Rootkit

A rootkit is a collection of tools that a hacker uses to mask her intrusion and obtain administrator-level access to a computer or computer network. The intruder installs a rootkit on a computer after first obtaining user-level access, either by exploiting a known vulnerability or cracking a password. The rootkit then collects user IDs and passwords to other machines on the network, thus giving the hacker root or privileged access.

A rootkit may consist of utilities that also

- Monitor traffic and keystrokes
- Create a "back door" into the system for the hacker's use
- Alter log files
- Attack other machines on the network
- Alter existing system tools to circumvent detection

The presence of a rootkit on a network was first documented in the early 1990s. At that time, Sun and Linux operating systems were the primary targets for a hacker looking to install a rootkit. Today, rootkits are available for a number of operating systems and are increasingly difficult to detect on any network (searchSecurity.com, 2004b).

Malicious Web-Based Code

A malicious web-based code, also known as a web-based mobile code, simply refers to a code that is portable to all operating systems or platforms such as HTTP, Java, and so on. The "malicious" part implies that is it a virus, worm, Trojan horse, or some other form of malware. Simply put, the malicious code does not care what the operating system may be or what browser is in use. It infects them all blindly (Yakabovicz, 2003).

Where do these codes come from, and how are they spread? The first generation of the Internet was mostly indexed text files. However, as the Internet has grown into a graphical, multimedia user experience, programmers have created scripting languages and new application technologies to enable a more interactive experience. As with any new technology, programs written with scripting languages run the gamut from useful to poorly crafted to outright dangerous.

Technologies such as Java and ActiveX enable these buggy or untrustworthy programs to move to and execute on user workstations. (Other technologies that can enable malicious code are executables, JavaScript, Visual Basic Script, and plug-ins.) The Web acts to increase the mobility of code without differentiating between program quality, integrity, or reliability. Using available tools, it is quite simple to "drag and drop" code into documents that are subsequently placed on web servers and made available to employees throughout the organization or individuals across the Internet. If this code is maliciously programmed or just improperly tested, it can cause serious damage.

Not surprisingly, hackers have used these very useful tools to steal, alter, and erase data files as well as gain unauthorized access to corporate networks. A malicious code attack can penetrate corporate networks and systems from a variety of access points, including websites, HTML content in email messages, or corporate intranets. Figure 5.5 shows the rapid growth of mobile malicious code in recent years versus viruses.

FIGURE 5.5 Growth of mobile malicious code.

Today, with literally billions of Internet users, new malicious code attacks can spread almost instantly through corporations. The majority of damage caused by malicious code happens in the first hours after a first-strike attack occurs—before there is time for countermeasures. The costs of network downtime or theft of IP make malicious code a top priority (finjan software, 2004).

Logic Bombs

A logic bomb is a type of malware that executes its malicious purpose when a specific criteria is met. The most common factor is date/time. For example a logic bomb might delete files on a certain date/time. An example is the case of Roger Duronio. In June 2006, Roger Duronio, a system administrator for UBS, was charged with using a logic bomb to damage the company's computer network. His plan was to drive the company stock down due to damage from the logic bomb and so he was charged with securities fraud. Duronio was later convicted and sentenced to 8 years and 1 month in prison and ordered to pay $3.1 million restitution to UBS.

Spam

Spam is something most readers are probably familiar with. It is unwanted email. Spam is email that is sent out to multiple parties, that is unsolicited. Often it is used for marketing purposes, but it can be used for much more malicious goals. For example, spam is a common way to spread a virus or worm. Spam is also used to send emails enticing recipients to visit phishing websites in order to steal the recipient's identity. Essentially, spam is, at best, an annoyance and, at worst, a vehicle for spyware, viruses, worms, and phishing attacks.

Detecting and Eliminating Viruses and Spyware

Antivirus Software

In this chapter, and throughout this book, the need for running virus–scanning software has been discussed. It is prudent at this point to provide you with some details on how virus scanners work and information on the major virus-scanning software packages. This information should help you better understand how a virus scanner might help protect your system and help you make intelligent decisions regarding the purchase and deployment of some antivirus solution.

A virus scanner can work in one of two ways. The first is to look for a signature (or pattern) that matches a known virus. This is why it is important to keep your virus software updated so that you have the most recent list of signatures with which to work.

The other way in which a virus scanner might check a given PC is to look at the behavior of an executable. If that program behaves in a way consistent with virus activity, the virus scanner may flag it as a virus. Such activity could include the following:

- Attempting to copy itself
- Attempting to access the address book of the system's email program
- Attempting to change Registry settings in Windows

Figure 5.6 shows the Norton AntiVirus software in action. You can see that the virus definitions are up-to-date, that the virus scanning is enabled, auto-protection is enabled, and the Internet worm protection is enabled as well. The other popular virus scanners have many of the same features.

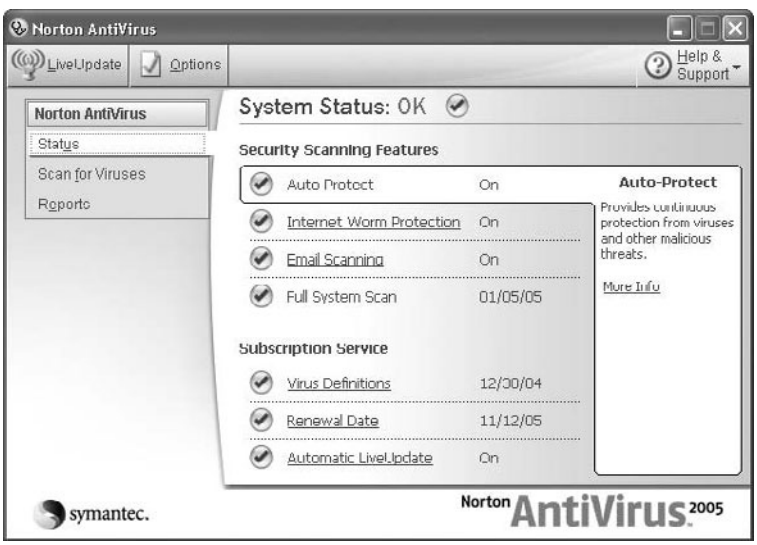

FIGURE 5.6 Norton AntiVirus interface.

Most antivirus software today offer additional features. Some of these features include warning the user of known phishing websites, detecting spyware as well as viruses, and even detecting likely phishing attempts. Any modern antivirus should be a comprehensive package, protecting against a variety of attacks, rather than just stopping viruses.

Antispyware Software

Fortunately, just as there are many different spyware applications available, there are likewise many different software applications on the market that are designed specifically to detect and remove spyware. These applications are also usually available at extremely low cost. You can often get a free trial version to use for a limited time so that you can make a more intelligent purchasing decision. Of course, the most prudent course of action you can take to avoid getting spyware on your machine is to never download anything from the Internet that does not come from a very well-known and trusted website. However, in an organizational environment, you cannot simply rely on your employees to do the right thing. It is prudent as the company's computer security expert to take steps yourself to prevent the employees from compromising your system security.

Some of the better known and more widely used antispyware applications include Spy Sweeper from www.webroot.com, Zero Spyware Removal from www.zerospyware.com, and Spector Pro from www.spectorsoft.com. All of these applications can be obtained for anywhere from $20 to $50, and many offer a free trial version. And most modern antivirus software either includes antispyware, or it can be added as an option.

Figure 5.7 shows the items found by running the WebRoot Spy Sweeper software on a system. These items can be selected for quarantine and removal.

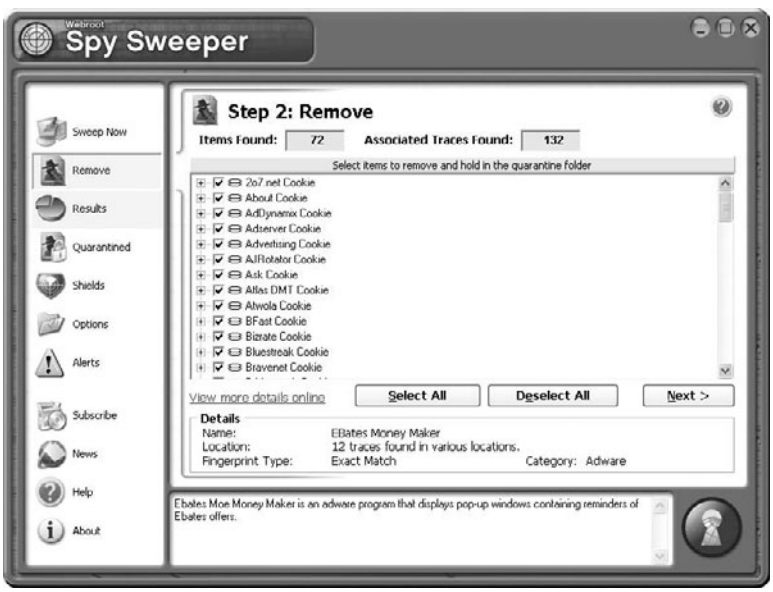

FIGURE 5.7 Items recommended for removal by Spy Sweeper.

Figure 5.8 shows the summary results after all of the items found have been quarantined. Note that the number of files scanned, the number of items removed, the date the full sweep was performed, as well as additional information is all detailed on this summary page. Each of the antispyware applications would provide similar results and each contain similar options for sweeping a system.

FIGURE 5.8 Summary results of a system sweep.

Summary

Clearly, there are a number of ways to attack a target system: by denial of service, virus/worm, Trojan horse, buffer-overflow attacks, and spyware. Each type of attack comes in many distinct variations. It should be obvious by this point that securing your system is absolutely critical. In the upcoming exercises, you will try out the antivirus programs by Norton and McAfee. There are so many ways for a hacker to attack a system that securing your system can be a rather complex task.

Another theme that is driven home throughout this chapter is that many, if not most, attacks are preventable. The exercises ahead will give you practice in figuring out how to prevent the Sasser and Sobig virus. In most cases, prompt and regular patching of the system, use of antivirus tools, and blocking unneeded ports would prevent the attack. The fact that so many systems do get infected is an indication of the very real problem of network professionals who are not skilled in computer security.

Test Your Skills

MULTIPLE CHOICE QUESTIONS

1. Which of the following is the best definition of virus?
   - A. Program that causes harm on your computer
   - B. Program used in a DoS attack
   - C. Program that slows down networks
   - D. Program that self-replicates
2. What is the most common damage caused by virus attacks?
   - A. Slowing down networks by the virus traffic
   - B. Deleting files
   - C. Changing the Windows Registry
   - D. Corrupting the operating system
3. What is the most common way for a virus to spread?
   - A. By copying to shared folders
   - B. By email attachment
   - C. By FTP
   - D. By downloading from a website

4. Which of the following is the primary reason that Microsoft Outlook is so often a target for virus attacks?

   A. Many hackers dislike Microsoft.
   B. Outlook copies virus files faster.
   C. It is easy to write programs that access Outlook's inner mechanisms.
   D. Outlook is more common than other email systems.

5. Which of the following virus attacks used a multimodal approach?

   A. Slammer virus
   B. Mimail virus
   C. Sobig virus
   D. Bagle virus

6. What factor about the Sobig virus made it most intriguing to security experts?

   A. It spread in multiple ways.
   B. It deleted critical system files.
   C. It was difficult to protect against.
   D. It was very sophisticated.

7. What was most interesting to security experts about the Mimail virus?

   A. It spread more rapidly than other virus attacks.
   B. It spread in multiple ways.
   C. It grabbed email addresses from documents on the hard drive.
   D. It deleted critical system files.

8. Which of the following reasons most likely made the Bagle virus spread so rapidly?

   A. The email containing it claimed to be from the system administrator.
   B. It copied itself across the network.
   C. It was a sophisticated virus.
   D. It was particularly virulent.

9. What made the Bagle virus so dangerous?

   A. It changed Windows Registry settings.
   B. It disabled antivirus software.
   C. It deleted key system files.
   D. It corrupted the operating system.

10. Which of the following is a way that any person can use to protect against virus attacks?

    A. Set up a firewall
    B. Use encrypted transmissions
    C. Use secure email software
    D. Never open unknown email attachments

11. Which of the following is the safest way to send and receive attachments?

    A. Use a code word indicating the attachment is legitimate
    B. Only send spreadsheet attachments
    C. Use encryption
    D. Use virus scanners before opening attachments

12. Which of the following is true regarding emailed security alerts?

    A. You must follow them.
    B. Most companies do not send alerts via email.
    C. You can trust attachments on security alerts.
    D. Most companies send alerts via email.

13. Which of the following is something a Trojan horse might do?

    A. Open a back door for malicious software
    B. Change your memory configuration
    C. Change ports on your computer
    D. Alter your IP address

14. What is a buffer-overflow attack?

    A. Overflowing a port with too many packets
    B. Putting more email in an email system than it can hold
    C. Overflowing the system
    D. Putting more data in a buffer than it can hold

15. What virus exploited buffer overflows?

    A. Sobig virus
    B. Mimail virus
    C. Sasser virus
    D. Bagle virus

16. What can you do with a firewall to help protect against virus attacks?
    A. There is nothing you can do on the firewall to stop virus attacks.
    B. Shut down all unneeded ports.
    C. Close all incoming ports.
    D. None of the above.
17. A key logger is what type of malware?
    A. Virus
    B. Buffer overflow
    C. Trojan horse
    D. Spyware
18. Which of the following is a step that all computer users should take to protect against virus attacks?
    A. Purchase and configure a firewall
    B. Shut down all incoming ports
    C. Use nonstandard email clients
    D. Install and use antivirus software
19. What is the primary way a virus scanner works?
    A. By comparing files against a list of known virus profiles
    B. By blocking files that copy themselves
    C. By blocking all unknown files
    D. By looking at files for virus-like behavior
20. What other way can a virus scanner work?
    A. By comparing files against a list of known virus profiles
    B. By blocking files that copy themselves
    C. By blocking all unknown files
    D. By looking at files for virus-like behavior

Exercises

EXERCISE 5.1: Using Norton AntiVirus

1. Go to the Norton AntiVirus website (www.symantec.com/downloads) and download the trial version of their software.
2. Install and run their software.
3. Carefully study the application, noting features that you like and dislike.

EXERCISE 5.2: Using McAfee AntiVirus

1. Go to the McAfee antivirus website (http://us.mcafee.com/root/package.asp?pkgid=100&cid=9901) and download the trial version of their software.
2. Install and run their software.
3. Carefully study the application, noting features you like and dislike.

EXERCISE 5.3: Preventing Sasser

1. Using resources on the Web or in journals, carefully research the Sasser virus. You may find that www.f-secure.com and Symantec's virus information center at www.sarc.com/avcenter/ are helpful in this exercise.
2. Write a brief essay about how it spread, what damage it caused, and what steps could be taken to prevent it.

EXERCISE 5.4: Preventing Sobig

1. Using resources on the Web or in journals, carefully research the Sobig virus. You may find that www.f-secure.com and Symantec's virus information center at www.sarc.com/avcenter/ are helpful in this exercise.
2. Write a brief essay about how it spread, what damage it caused, and what steps could be taken to prevent it.

EXERCISE 5.5: Learning about Current Virus Attacks

1. Using resources on the Web or in journals, find a virus that has been spreading in the last 90 days. You may find that www.f-secure.com and Symantec's virus information center at www.sarc.com/avcenter/ are helpful in this exercise.
2. Write a brief essay about how it spread, what damage it caused, and what steps could be taken to prevent it.

EXERCISE 5.6: Using Antispyware Software

1. Go to Spy Sweeper website (www.webroot.com/downloads) and download the trial version of the software.
2. Install and run the Spy Sweeper software.
3. Carefully study the application, exploring the options and noting features that you like and dislike.
4. Repeat this process to download and explore Adaware software (which is available from a variety of websites).
5. Assess which of these two antispyware applications would work best for your computer system.

Projects

PROJECT 5.1: Antivirus Policies

This activity can also work as a group project.

Considering what you have learned in this chapter and in previous chapters, as well as using outside resources, write an antivirus policy for a small business or school. Your policy should include technical recommendations as well as procedural guidelines. You may choose to consult existing antivirus policy guidelines that you find on the Web to give you some ideas. The following websites may be of some help to you in this project:

- www.sans.org/resources/policies/Anti-virus_Guidelines.pdf
- http://irmc.state.nc.us/documents/approvals/1_VirusPolicy.pdf

However, you should not simply copy their antivirus policies. Rather, you should come up with your own.

PROJECT 5.2: The Worst Virus Attacks

Using resources on the Web, books, or journals, find a virus outbreak that you consider to have been the worst in history. Write a brief paper describing this attack, and explain why you think it is the worst. Was it widely spread? How quickly did it spread? What damage did it do?

PROJECT 5.3: Why Write a Virus?

A number of hypotheses have been formed regarding why people write a virus. These hypotheses range from the frankly conspiratorial to the academically psychological. Taking whatever position you feel is most likely, write a paper explaining why you think people take the time and effort to write a virus.

Case Study

Chiao Chien manages IT security for a school. Given the wide range of people who use the school's computers, it is difficult for Chien to prevent virus attacks. Chien has a reasonably good budget and has installed antivirus software on every machine. He also has a firewall that has all unneeded ports blocked, and there is a school policy prohibiting the downloading of any software from the Web. Consider the following questions:

- How secure do you think Chien's network is from virus attacks?
- What areas has Chien not secured?
- What recommendations would you make to Chien?

Chapter 6

Techniques Used by Hackers

Chapter Objectives

After reading this chapter and completing the exercises, you will be able to do the following:

- Understand the basic methodology used by hackers
- Be familiar with some of the basic tools
- Understand the hacking mentality

Introduction

In the preceding five chapters, we have explored computer security and various security breaches. In this chapter we will be exploring the techniques that hackers use to commit computer crimes. Before we go any further, it is important that you realize that many hackers are not criminals. A hacker is a person who wants to understand a system, often by probing its weaknesses. There are even hackers that work for organizations, testing the organization's system security. This is called penetration testing. This is also often referred to as white hat hacking. The EC Council (www.eccouncil.org) even has a certification for this, the Certified Ethical Hacker. Then there are hobbyists, who simply like to probe systems, learning about them. There is a magazine for such people called *2600* (www.2600.com). However, there are people who use hacking techniques to breach systems to steal data, damage systems, or commit other cyber crimes. These people are usually referred to as black hat hackers or crackers.

The techniques presented in this chapter are not only presented to give the reader an understanding of how black hat hackers work, but also provide a method whereby a network administrator can perform a penetration test on his or her own network. By attempting some of these techniques on your network, you can assess your vulnerability. It should be pointed out that you should only do this once you are very comfortable with the techniques in this chapter, and only with permission from senior management.