NetworkCare
The knowledge behind the network

Lucent Technologies
Bell Labs Innovations



Worrisome New Windows Attacks

By Jay G. Heiser, Senior Security Consultant

Introduction

Remotely accessible computers have always attracted attack attempts and now new forms of hostile code are dramatically increasing the risk associated with connecting a Windows PC to the Internet. Early remote system strikes were waged exclusively through telephone lines using modems. As the Internet developed in the 90s, new levels of connectivity and convenience exposed computers to a rich array of new threats. For a number of years, Microsoft Windows systems avoided the compromise, hijacking, borrowing, and manipulation endured by other Internet hosts. Unfortunately, an increasing number of clever software attacks have been developed and targeted against Microsoft Windows systems

A typical example is an engineer whose home NT system was connected to the Internet through a cable modem. In December 1998, he noticed that his CD drawer was opening and closing by itself. Before he could recover from that surprise, a chat window appeared on his screen. Text appeared telling him that he was being controlled remotely over the Internet and asking him if he wanted to discuss it. This wasn't a *Twilight Zone* episode. Incidents like this are happening to Windows users every day.

The threat isn't limited to PC hijackings. Thousands of PCs were infected by the Melissa virus over a period of a few days. Melissa is more than just a virus; it can steal your data and send it to someone else outside of your organization. Explorer.zip, which targets and destroys specific data, uses e-mail to spread between organizations, and can then use Windows file sharing to infect additional systems even when mail has been disabled.

Attack Profiles

Every Internet host is threatened by joy riders looking for stimulation, and corporate espionage represents a serious but overlooked threat to many organizations. Some assailants are voyeuristic, and just enjoy seeing data that is not meant to be publicly shared. Others are excited by the challenge of system cracking. A vocal minority is politically motivated. However, most Internet-based attacks are relatively benign, and are not intended to steal or damage information. Unfortunately, a system administrator must always assume the worst and treat every attack as being capable of causing damage. Attackers target systems because they have the potential of offering something desirable to the attacker, and because the attacker has a reasonable expectation of finding exploitable vulnerabilities.

Many poorly administered UNIX hosts have ended up as playgrounds for system intruders. Multi-user systems, such as UNIX and VMS, usually run a number of complex network services that are

often buggy and at risk. These operating environments are completely integrated with the network, so an attacker who has cracked one remotely has access to the same Graphical User Interface (GUI) and powerful command line as a legitimate user. (In his doctoral thesis, *An Analysis of Security Incidents on the Internet:1989 – 1995*, John D. Howard showed that the object of most Internet-based attacks reported to the CERT was to gain a command prompt.) Attackers like to use cracked hosts as a springboard to attack other systems, either within a corporate WAN or on the Internet. By logging into a remote system, using that system to log in to another system, and so on, intruders can hide their locations effectively.

Microsoft Windows is a very different environment than UNIX. One person at a time accesses it directly at the console. It does not have X Windows, and the command-line environment is relatively weak. Windows vulnerabilities are harder to exploit in ways that are interesting to hackers. Internet denial-of-service attacks that cause system crashes have been common. However, once an attacker has knocked over a couple of file servers remotely, the thrill wears off. Denial-of-service attacks such as Teardrop are easy, but they offer no benefit to most system attackers. Windows servers with file sharing enabled over TCP/IP are usually vulnerable. Joy riders can potentially read and even alter files, but access to corporate file servers is usually not interesting to them. The lack of a powerful command-line environment has made Windows an unappealing target for unskilled and unmotivated intruders.

However, an increase in the sophistication of system exploitation software has reduced the need for expertise on the part of the attacker. A clear trend over the last 20 years, it has reached the point where the popular media has even noticed it. A recent *Doonesbury* storyline involved the hostile remote manipulation of PCs and used the term *kiddie scripts* to describe sophisticated tools created by more skillful programmers. These tools could be used by hacker wannabees to crack systems. A comic strip like this exaggerates life to make a point, but it isn't far from the truth of a NetBus attack. The concept of kiddie scripts has caught on within the Internet community. The term is evidence of widespread recognition that highly developed hacking code represents a growing threat. It no longer takes a rocket scientist to launch a successful attack on a Windows system across the Internet—anybody can do it. Convenient attack software has even automated the process of finding vulnerable target host—hackers just sit back and let the computers do the work.

Back Doors

A *back door* is an externally introduced mechanism that bypasses existing access controls to allow unintended system access. New malware attacks take advantage of existing Windows network services that were previously accessible only to programmers. They create back doors that make Windows systems easier and much more interesting to attack.

Attackers accomplish Windows remote takeovers by covertly inserting remote control software. Ostensibly developed as system administration tools, programs such as NetBus and BackOrifice have been proliferating across the Internet. These are two-part programs. A small, virtually unnoticeable server component is installed on the victim's PC, which the attacker accesses remotely through a convenient client GUI. The attacker uses the server as a bidirectional surreptitious channel to sidestep existing security mechanisms and access the compromised system remotely—all without the victim's knowledge.

The NetBus GUI provides a variety of useful strings to the remote puppeteer. Along with the ability to capture the screen, read and write any file, start and stop any program, or reboot the PC,

it also includes microphone access. So the attacker can eavesdrop on anything within earshot of the victim's PC. The ability to monitor and store keystrokes means that software-based encryption services can be subverted by capturing the user's access password when he or she types it. Not only can this defeat file encryption and network logins, but it can also undermine digital signatures. Did you receive that signed S/Mime or PGP message from the owner of the digital certificate, or an attacker who gained remote control of a PC and masqueraded as the legitimate owner? How does a merchant know if it was really you who ordered that stereo or someone who hijacked your PC and used your digital identity? If one of your remote access users is compromised by a back door, what will prevent the attacker from gaining access to your corporate WAN?

When a home system is hacked, it can result in an unfortunate loss of personal privacy, and possibly money. In a corporate setting, it is easy to envision a disenfranchised employee installing back doors clandestinely on corporate computers and gaining permanent unauthorized access to sensitive information including salaries, proposals, and trade secrets.

The Computer Emergency Response Team (CERT) at Carnegie-Mellon University continues to receive numerous reports from Internet Service Providers that hackers who are looking for exploitable instances of BackOrifice and NetBus are methodically scanning their IP address space. Systems using cable modems and DSL are popular targets because they are connected to the Internet for longer periods at a high bandwidth. Data isn't available on the number of usable back doors revealed through these scans, but it is reasonable to assume that the benefit to the attackers of finding exploitable PCs outweighs the effort required to run the automated scanning utility.

Data-Stealing Viruses

Computer security experts had long dreaded the widespread availability of hostile code capable of searching a victim's file system for specific data and then secretly transferring it back out to the attacker. The efficiency of such an attack tool made its creation inevitable. Viruses like this first appeared in the mid-1980s. In 1998, several of these exploits occurred on Windows, such as one reported in June in which a teenager stole 500 AOL passwords using an e-mailed Trojan horse.

In January 1999, the Caligula Microsoft Word macro virus emerged. Caligula searches infected PCs for Pretty Good Privacy (PGP) key rings. PGP is a public key-based e-mail security product that provides privacy by encrypting messages, and provides non-repudiation through digital signature. The recipient of a digitally signed message can know with a very high degree of certainty which private key was used to sign the message. Only if the private key is protected from compromise can the recipient be assured that the legitimate key owner actually signed the message. When the private key is stored on a hard drive, as it is with PGP, it is encrypted and accessed through a password entered by a user. If attackers can gain access to the encrypted private key—which is a component of the PGP key ring—then they can potentially decrypt it. If they are successful, they would then have the power to masquerade as the owner of the key, and the ability to decrypt any encrypted messages sent to the keyholder. Caligula dramatically demonstrates how a Word macro virus can steal specific data and then use FTP to transmit the data out to a remote attacker back through a firewall.

Currently, over 10,000 viruses have been identified but only a few hundred are actively circulating. The Melissa virus is unique in having disseminated itself so successfully. It had a dramatic effect on productivity, required hours of cleanup, and caused many e-mail outages. While downtime can be costly, it isn't necessarily Melissa's most dangerous effect. Melissa is designed to send infected

documents, which might contain sensitive or embarrassing information intended only for internal use, covertly via e-mail to 50 different addresses.

Reports of successful data theft using hostile code are rare, but information security statistics are always scarce because most organizations are hesitant to report attacks. Caligula and Melissa are not designed to be subtle; they are designed by egotistical hackers to attract as much attention as possible. Such attacks are annoying, but disloyal or otherwise highly motivated insiders cause the most costly security incidents. An industrial espionage agent isn't motivated by infamy, and successful spies remain that way by keeping a low profile. Professional thieves don't leave tracks, and they don't brag about their exploits. Organizations with data worth stealing should assume that motivated individuals would attempt to steal that data using the most efficient and least risky method. Computer-savvy industrial spies are aware of Melissa and know where to get its easily modifiable source code. Dozens of Melissa variants appeared within days of her initial appearance, so creating a stealthier version would be simple.

Getting Infected

Internal attackers can often install or exploit code on a PC when the owner isn't around; external attackers usually introduce hostile code through the network. Offered over the Internet as e-mail attachments, victims usually infect themselves by executing a Trojan horse. A Trojan horse is a file or program that purportedly offers some desirable functionality, but actually has an ulterior motive. Animated greeting cards are popular infection vectors, and a new back door is circulating as a screen saver. A bogus e-mail message, such as the one typically carrying the recent Explore.zip worm, is a form of Trojan horse.

Kiddie scripts are available to piggyback NetBus or BackOrifice on to any executable file that an attacker feels a victim would be willing to execute. In effect, this creates a customized Trojan horse. Attackers who are too lazy to trojanize the remote control server program themselves can just send potential victims a video game that is already prepared as a NetBus Trojan, such as "Whackamole.exe."

Prevention

What can you do to prevent these attacks from succeeding? While the sophistication of the attacks is increasing, the good news is that the appropriate response policies remain constant. Diligent adherence to procedures will prevent most hostile code infections. The first step in computer security is always to develop and implement a well-conceived policy establishing your organization's priorities and basic security rules. You can prevent hostile code infections through procedural and technical countermeasures; organizational policy must address both. Here are some steps you can take to ensure safe software:

- Train users not to double-click on e-mail attachments. Make them aware of the danger associated with accepting and using files from external sources.
- Run at least one anti-virus product on every desktop and keep its definition file current. Good AV products recognize the distinctive signatures of the demon exploits discussed above, and in case of accidental infection, can remove them from your systems. Unfortunately, anti-virus software products cannot scan for specific hostile code until the AV vendors become aware of

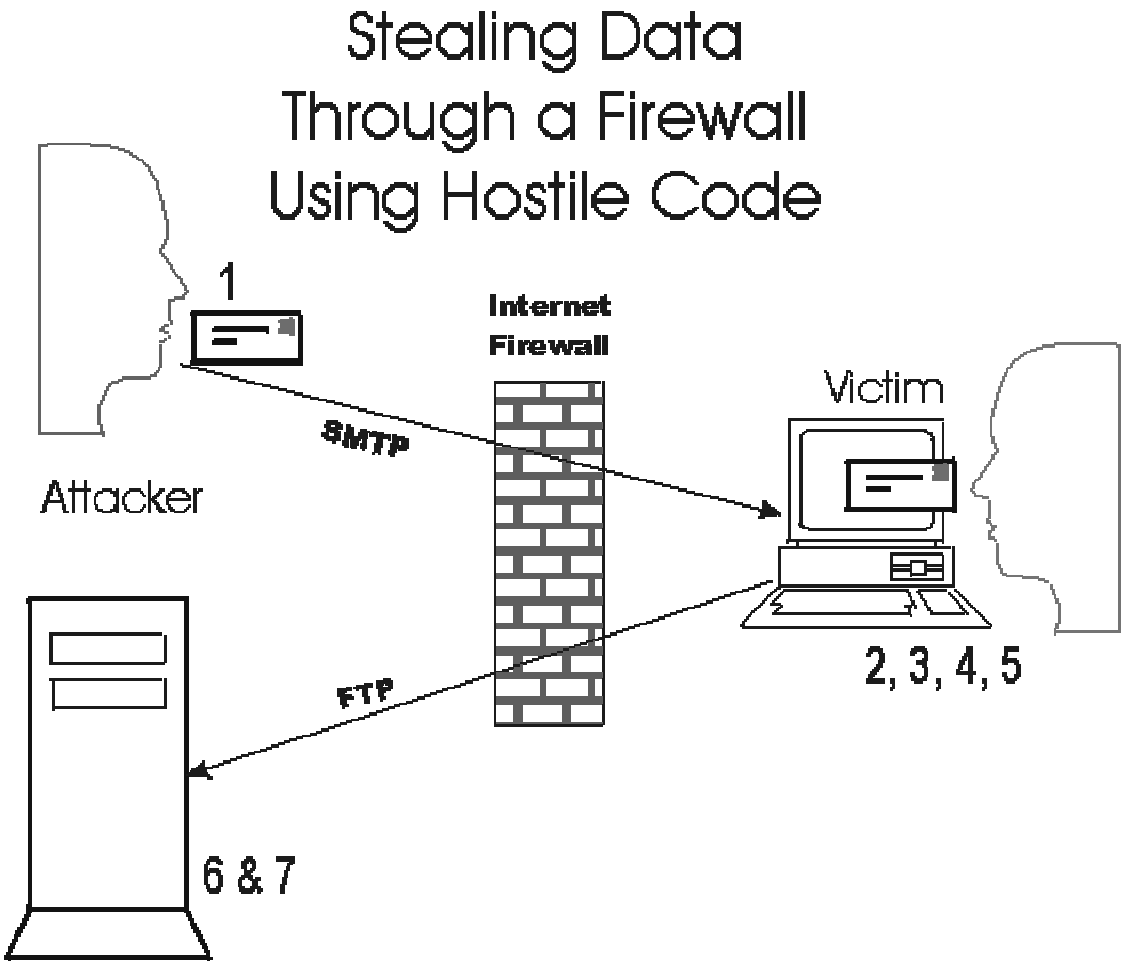
the new code and update their definition files. This latency prevents hostile code detectors from ever being 100 percent effective.

- Ensure that all MS-Word users turn on Macro Virus Protection.
- Use WordPad instead of MS-Word to view .doc files attached to e-mail.
- Use a virus wall to scan all incoming and outgoing messages, and use a different AV product than used on the desktop systems.
- Write-protect system configuration files and configure workstation shares carefully to prevent writing of configuration files such as win.ini over the network.
- Do not give NT administrator privileges to accounts used for routine office automation or development. Instead, reserve membership in the administrator's group for accounts used only when actually performing system maintenance.

A common but naïve belief is that a firewall is a total solution to Internet security concerns. Firewalls are network perimeter security devices that control access to specific network services—a limited task they perform well. Most Internet firewalls are configured to allow all incoming e-mail to reach a specific internal mail server. Typically, firewalls allow internal users to access the Internet as they see fit. A program running on a user's PC has the same outgoing access that the user has. Hostile code can be incorporated into e-mail messages, arriving on some hapless user's inbox as a Microsoft Word document or innocent-looking animated greeting card. If the user double-clicks and executes the hostile code, it will have full access to the user's PC, and will run with all the privileges of that user. Even worse, the hostile code will be behind the firewall. Because outgoing FP and e-mail are usually not controlled by the firewall, the malware can use one of these services to create a surreptitious channel and transfer stolen data back out to the Internet.

By controlling external access to internal systems, firewalls can prevent casual attackers from finding and subverting internal systems that may already be infected with the forms of remote control software spreading on the Internet today. More sophisticated back doors that install themselves and autonomously connect back out to the Internet would not be stopped by most existing firewalls.

The following diagram and accompanying text illustrate how hackers can steal data through a firewall.



1. Hostile user on Internet e-mails malware, Caligula in this example, to victim. Malware is usually in the form of a Trojan horse disguised as screen saver.
2. Victim receives e-mail and double-clicks on attachment.
3. Caligula is extracted covertly and placed on system.
4. Caligula searches for PGP key ring.
5. Caligula sends stolen data back out to an Internet server.
6. Stolen data arrives on hacker server.
7. Attackers can now attempt to crack the PGP key ring on their own system at their leisure. Once they have cracked the key ring, they will have access to the victim's private key, allowing them to decrypt PGP-mail sent to the victim and to masquerade as him or her by using the victim's private key to sign outgoing mail.

Also, firewalls are not designed to examine incoming data and determine whether it is executable or what it is likely to do. Virus wall products, like MimeSweeper, are application-level countermeasures that are effective at screening out hostile code from e-mail. These products can

run on the firewall or the mail server. However, it is usually most practical to use a stand-alone machine for mail filtering, locating it between the firewall and the organizational mail server. Operating as an e-mail proxy, virus walls open each message, check for attachments, unarchive them, and scan them for recognizable hostile code using a commercial AV product. Such products are efficient at unzipping attachments, but they cannot open encrypted messages. If an organization allows incoming encrypted attachments, they can only be scanned at the desktop. E-mail scanners should be considered as an augmentation to desktop control—not as a replacement. By using different AV products on the virus wall and the desktop, the combination of two different products in series provides you with a better detection rate than either product by itself.

PC anti-virus software real-time detection capabilities have proven effective at stopping the spread of recognized hostile code attached to e-mail messages. Running AV software in this mode does raise performance concerns, but the cost of faster hardware can be offset against the cost of downtime, cleanup, and loss of system integrity after a significant viral infection. Properly installed and configured, it will at least prevent users from infecting themselves with well-known attack software.

Given that no defense is perfect, it is prudent to anticipate failures. An effective security policy includes both attack detection and an attack response plan. Network security scanners can check for instances of running back door code as part of their routine network vulnerability scans. It is also possible to monitor hostile connection attempts with an intrusion detection system (IDS), allowing you to log and track incidents. Log review will paint a picture of attack frequency and patterns, providing an understanding of your organization's threat environment. Changes in attack patterns and prompt alerts from an intrusion detection system will let you know when it is necessary to increase your response posture. Commercial intrusion detection products are being modified to detect known back door attacks, but they cannot detect every hostile action. They are useful countermeasures, but they are just one component in a larger security architecture.

Conclusion

It is unfortunate that Microsoft Windows systems do not protect themselves against their own users. Eventually, all executable code will be signed digitally, providing a higher level of assurance that the code has not been modified and that its source is legitimate. This technology is already incorporated into the security architecture of Java 2.0 and ActiveX, but not yet widely used on the Internet. The secure file system provided with Windows 2000 will support digitally signed files and their verification at runtime. If this proves practical and easy to administer, it will go a long way toward preventing undesirable code from gaining a foothold.

Explore.zip, Melissa, and NetBus are symptoms of an evolved threat environment. Microsoft Windows users have had an easy ride on the Internet and avoided serious compromise. Unfortunately, given the regularity with which other operating environments have been hacked, and the simplicity and availability of powerful new Windows attacks, it is reasonable to expect an increase in the rate of successful Windows attacks. Don't wait to act until after a costly security incident. Take steps now to tighten up your Windows system configuration and educate your users.

The following links provide further information about hostile code.

Malware information pages:

- The Computer Virus Myths Page, http://kumite.com/myths/
- IBM's Anti-Virus Online http://www.av.ibm.com
- The WildList Organization International, http://www.wildlist.org/
- Back Orifice Resource Center, http://skyscraper.fortunecity.com/cern/600
- PCHelp: The Back Orifice Page, http://www.nwi.net/~pchelp/bo/bo.htm
- PCHelp: The NetBus Page, http://www.nwi.net/~pchelp/nb/nb.htm
- Ports used by Trojans, http://www.simovitz.com/nyheter9902.html
- Data Fellows virus news: http://www.datafellows.com/v-descs/
- Cert advisory warning about scans for back doors: CERT Summary CS-99-01
- "Computer-Communications Security Risks: Melissa is Just the Tip of a Titanic Iceberg" Peter Neumann's written testimony for the U.S. House Science Committee: http://www.csl.sri.com/~neumann/house99.html

AV product test sites:

- Virus Bulletin 100% Awards, http://www.virusbtn.com/100
- Virus Test Centre, http://agn-www.informatik.uni-hamburg.de/vtc
- Check-Mark certified anti-virus products, http://www.check-mark.com/
- ICSA certified anti-virus products, http://www.icsa.net/html/communities/antivirus/certification/certified_products/

For further information, see NetworkCare's website at http://www.lucent-networkcare.com, or call 1-877-369-1115 in the U.S. or 1-727-217-2303 outside the U.S.

NetworkCare is a servicemark and "The knowledge behind the network" is a registered trademark of Lucent Technologies Inc. All other trademarks and registered trademarks are properties of their respective holders.

Copyright © 2000, Lucent Technologies Inc. All rights reserved.