
Mitigating Malware in Userland

by Kevin Mitnick,
Mitnick Security Consulting, LLC

Sponsored by:

AppSense[®]

www.AppSense.com

June 2006

Contents

A Changing World	3
Hackers at Work as Corporate Spies.....	3
Hacker Attacks: Common and Costly	6
The Profit-Oriented Attacker	7
Other Attack Modalities	9
Propagation by Social Engineering	11
Countermeasures	12
Conclusion: Mitigating Common Malware Attacks	17

A Changing World

Computer security is a cat-and-mouse game in which IT departments are continually adding measures to defend against the newest types of attack while attackers are ever busy peeking over the shoulder of the security companies to discover their latest strategies, and then working to develop still newer attacks that will be harder to defend against.

That part isn't new. What is new is the motive of the attackers. Until very recently—some experts say within the past year or two—almost all attacks were done for kicks and for bragging rights, which one hacker who allowed himself to be interviewed by the *New York Times* referred to as “street cred.” It's “Look what I did,” with the ultimate goal of an attack so widespread or devastating (or embarrassing to a major company) that it will be covered by CNN, which was and still is almost the hacker's equivalent of winning an Oscar.

Yet today the motive is quickly shifting to putting cash in the pocket—stealing Euros, rubles, shekels, yuans, rupees, dollars. And that attacker targeting your company may no longer be a teenager or group of teens pounding away on their keyboards at 3 A.M. but full-time workers in the employ of an eastern European mafia. Or working for one of your competitors.

Hackers at Work as Corporate Spies

Jacob Sachs recently received an e-mail offering a business proposal from a firm his company had partnered with in the past¹. He opened the proposal, read it, and quickly saw that it suggested a co-marketing arrangement that would conflict with one his company already had in place. He wrote a brief response that his company wasn't interested but would like to hear other suggestions for future cooperative deals. He deleted the incoming message and thought no more about it.

At another company, the VP for product development, Ruth Pollack, received a CD labeled “Marketing Report” indicating that it contained market intelligence about competitive firms; when she loaded the disk, she found it contained an assortment of newspaper and magazine articles, some of them not very current. “Nothing of value,” she thought. She started to file the CD but then decided to drop it into the trash.

In another city, Gadi Mayer was smiling from ear to ear. Jacob and Ruth had both fallen victim to his Trojan horse and backdoor spyware attacks. Soon he would be collecting valuable business intelligence from both their companies, intelligence that he would be paid very handsomely for.

Around the same time, similar scenes were being played out at more than twenty companies. These people were all victims, their companies were victims, the losses were huge, many people became very angry, and quite a number of others will be heading to prison.

The business proposals, marketing data, and other cover stories were phony, a masquerade, and their function only to induce the employees to open the attached file. When any of them did, a new piece of malware, combining a Trojan horse and a backdoor, was installed on the person's computer. The malware, known as the HotWar Trojan, left the company vulnerable to having documents stolen from corporate computers, and transferred using FTP to remote file-storage servers. But in this attack, the big payoffs were not going to hackers but to other companies: firms that were paying handsomely to gain intelligence on the plans of their competitors. Though the cases have not yet gone to trial, police believe that company CEOs had personal knowledge of the corporate spying that their firms were paying for.

It's possible the scheme might never have come to light but for the discovery by an author that text from one of his yet-to-be-published books was posted on an Internet site. He found that alarming because it was a book he was still in the process of writing, and he had never shared any of the material, not with anyone at all. Somehow his work had been electronically stolen from his computer. It would turn out the attack on his computer had been triggered by a family feud, ignited by ill feelings in the wake of an ugly divorce involving the author's daughter Natalia.

Months of investigation would lead to Michael Haephрати, Natalia's exhusband, who had developed the Trojan horse software to use for malicious payback aimed at his one-time father-in-law. Michael, 41, tall and athletic, with an unlined brow, his blonde hair shaved clean to the skull, could get work as a henchman to the villain in a James Bond movie. Yet the real villain in this story wasn't Michael but his second wife, Ruth, a blue-eyed blonde with a lean face and a mischievous smile that deceives.

It was Ruth who listened to the serpent and tasted the forbidden apple. She would later acknowledge, "Michael always told me, 'Don't do it. Don't get in touch with the investigators. I have a feeling they are misusing the system.'"

Ignoring his warnings and pleadings, she offered his software for sale. Probably she thought they were safe, since the couple and their six-year-old daughter were living in England, while Ruth negotiated sales of the malware to private investigator firms in their native land, far away.

The PI firms she selected were outfits that had major corporations as clients. The software was specifically designed to enable an attacker to steal documents from a target company. Apparently when the PI firms discussed the opportunity with their corporate clients, executives at a number of companies found the possibility so tempting that they turned a blind eye to the blatant illegality, and agreed to finance spying efforts on competitors.

Firms targeted in the attacks ranged from large, publicly traded companies, to an importer of Audis and Volkswagens, as well as a PR firm, a leading mobile-phone operator, and, curiously enough, even a mineral water company and a sausage company. (This is true; if I had made it up, I would have picked something that sounded more believable.)

Following the tip by the author, investigators spent months tracking down the Haephratis and tracing the use of their software. According to published reports, *tens of thousands* of documents were stolen. Police arrested senior executives of 15 corporations including a defense contractor and two subsidiaries of a state-owned telecommunications firm; the parent firm was in the process of being acquired but, according to press accounts, the buyers asked to renegotiate the sale as a result of the damage to the firm's reputation in the wake of the computer crimes.

The frightening part of this true story is that every one of the targeted companies was believed to have been fully protected with firewall and antivirus software. The malware, delivered mostly through e-mail messages or CDROMs sent through the mail, was loaded in a way invisible to the user. Once the program had infected a computer, the attacker was then able to monitor everything on that machine, as well as make changes in the application programs, and send documents and screenshots to one of a number of remote servers.

Also arrested were the private investigators involved in the attacks. The CEO of one of the PI firms, following his arrest, threw himself down a flight of stairs at the police station and landed in hospital in critical condition.

Hacker Attacks: Common and Costly

It's not news that attacks on the world's computer systems have become an everyday item, so common that only the notorious cases ever get media attention. But unless you've been checking the data recently, the extent of the problem may be an eye-opener.

I recently had a personal experience with this. For an article they were preparing to run, *USA Today* cooperated with me and a San Francisco marketing firm to set up some brand new, never-used computers, put them on line, and see how long it was before they were attacked. Even I was surprised by the result: it took *fewer than four minutes* before the first of the computers was compromised by a worm exploiting a published vulnerability. This worm turned out to be one small element of a botnet attack-compromising a network of computers that can then be controlled over a communications channel by an attacker or group of attackers to take actions in unison.

Robert M. Wood, the chief information security officer at the University of Southern California, in Los Angeles, says that the university's computer system receives hacker attempts to break in at a rate Mr. Wood estimates to be a half million times a day. Certainly every major university is a high profile target that is particularly tempting to a certain kind of hacker. Even so - have you checked lately on how many times a day your own company's computers are being scoped out by hackers? (If nothing else, you may find this a convenient number to put on the table during your next budget cycle.)

The 2005 *CSI/FBI Computer Crime and Security Survey* gives figures on the dollar value of company losses to computer crime that are enough to make even the most self-confident IT Department think twice. The *average* value of loss from unauthorized access to information suffered by companies reporting in the Survey was \$303,000 for the year, up from a comparatively meager \$52,000 just a year earlier. The average loss from the theft of proprietary information for responding companies was \$356,000.

According to the *2005 Malware Report*² published by Computer Economics, Inc., 2004 was the third most costly year on record for malware attacks, with an aggregate worldwide cost of \$14.2 billion. (Computer Economics uses "malware" in the broad sense, to include virus, worms, Trojan Horses, spyware, and adware, as well as phishing, pharming, and spamming.)

Since 2005, when Microsoft released its “malicious software removal tool” for Win2000, XP, and Windows Server 2003, the tool has been used over 2.7 billion times, on 270 million different computers. From user data, which Microsoft collects automatically, the company has reported that an astounding 5.7 million those computers were infected with malware³; (for references, see Endnotes on last page of this paper). The company emphasizes that its program needs to be used in conjunction with commercial malwareblocking software, not instead of it, but the results make clear that standard antivirus technologies, while essential, leaving gaping holes in the security architecture.

Fortunately, newer approaches are now available to block many of the types of malware that are slipping past the anti-virus programs.

The Profit-Oriented Attacker

The HotWar story shines a spotlight on this important shift in hacker activities, a shift that I believe all IT departments need to be concerned about. Clever hackers who enjoy causing chaos have always been a threat. But when there are financial rewards at stake, I believe we need to expect to see *customized* attacks becoming much more widespread, much more clever and insidious, and therefore much more dangerous. This is especially true when unscrupulous businessmen sign up attackers to gain a corporate advantage, as in the Haephtrati story above. These attacks have already taken a number of new forms.

Consider these examples -

The Botnet Denial-of-Service Attack⁴

The CEO of online satellite TV retailer Orbit Communication Corp., Jay Echouafni, became a fugitive after being accused of recruiting a business associate to organize an attack on competitive websites. Echouafni apparently offered the excuse that the other sites were using content stolen from his company’s site, and the other companies were aiming denial-of-service attacks at Orbit. The associate, who was also named in a criminal indictment, hired three hackers, who organized a botnet denial-of-service attack.

One of the targeted companies, a firm that markets satellite TV receivers, was unable to sell any product for two weeks. Altogether this Orbit competitor suffering a loss in revenues believed to total more than \$2 million.⁵

One feature of a botnet attack is that users of the computers being compromised for the attack are unaware that their machines are being used in this way. But it doesn't always work out like that. In one 2005 case, a hacker's effort to assemble a botnet led to malfunctions in a Seattle hospital, where automatic operating room doors failed to open, pagers were jammed, and computers in the Intensive Care Unit shut down. Fortunately, the hospital was able to shift to backup systems quickly enough to avoid danger to patients.

One expert calls botnets "the number one emerging Internet threat."⁶

Ransomware

The frustratingly ingenious attack called ransomware, which has been deemed the work of "cyber cat burglars," is designed to leave an individual victim or targeted company willing to pay a ransom to the attacker. Though not new, the approach was rarely used until 2005, and is still uncommon.

The attacker plants software on user computers that locates individual files, encrypts each file, and then deletes the original, unencrypted version. The targeted company is then contacted and a ransom demanded for the password or software utility that will allow recovery of the data. One 2005 ransomware attack, using a Trojan dubbed "Cryzip," left behind a message that began:

Your computer caught our software while browsing illegal porn pages, all your documents, text files, databases was archived with long enough password.

You can not guess the password for your archived files - password length is more than 10 symbols that makes all password recovery programs fail to bruteforce it (guess password by trying all 7 possible combinations).

Though the message was written by a semi-illiterate (or struggling English-as-Second-Language person), the software was powerful and effective. One security company warns of "the possible start of a trend of this type of malware."⁷

It's interesting to note that a new field has sprung up called cryptovirology, which has been defined as "the study of the applications of cryptography to malicious software." The goal is to study new types of malware attacks, in order to be able to discover vulnerabilities before the attackers do, and to develop safeguards against them.

Rootkits

A rootkit is a type of program designed to mask the existence of malware so that it appears hidden from routine malware searches by security software. The rootkit is designed to replace components of the operating system at the user or kernel level. This might be done, for example, by hooking an API so the rootkit filters particular information in a way that deceives the operating system; when an application program makes an inquiry of the operating system - what files exist, what directories exist, or what processes are running, for example - the operating system provides a response that does not include the presence of the malware. This is achieved by hiding processes, services, TCP/IP ports, files, directories, and other operating system properties.

Attackers are becoming sophisticated and extremely clever today in creating rootkits that are highly effective in masking their presence. Once a rootkit infects a computer, it may be nearly impossible for anyone other than a highly experienced specialist to detect its existence and remove it. The rootkit authors are becoming more adept at anti-detection methods. In fact, one rootkit developer was recently offering a version of his rootkit for sale, asserting that it would be undetected by many of the anti-rootkit security programs.

Other Attack Modalities

Stealth Trojan horse attacks

When you hear about viruses, worms, Trojan horses, and other malware, you naturally think of a massive attack launched in the wild, a massive attack which will fail to infect many destinations but will succeed in the eyes of the perpetrator if it manages to install on even a small percentage of the machines that it reached. This is the traditional mass malware attack.

These attacks quickly come to attention of antivirus manufacturers, which can quickly have a new signature update or other fix on line and being downloaded by subscribers, sometimes within minutes.

As attackers turn more to the profit motive, security experts are seeing a move toward attacks directed at a specific company. Instead of designing a new piece of malware and launching it widely, the attacker targets his new innovation with the goal of stealing data from a single firm, or conducting a ransomware attack. (Again, this may be for personal gain, or the attacker may be "on assignment" from a competitive firm or criminal organization.)

This approach is especially insidious, because an attack of this type does not so readily come to the attention of the antivirus companies. In fact, with such limited distribution, the security firms may never become aware of the particular malware variation. "Stealth attacks" are therefore much harder to protect against when the defense consists of protective software that functions by blocking known, previously identified malware.

Cell phone attacks

Although marketed only for keeping tabs on what your spouse is up to, the software package called Flexispy is in fact a powerful spyware product that can be used to obtain information from another person's cellphone. Of course, using it for malicious purposes requires access to the cellphone long enough to install the product, or sending the spyware installer utility via Bluetooth over and over again until it's accepted by the user to stop the annoying barrage of requests.

The company advertises that the product allows a user to read text messages that have been sent to and from the phone, as well as information on all incoming and outgoing calls, including the duration of the call and the telephone address-book listing showing who the call was to or from. The company brags that the owner of the telephone will never know they are being spied on.

This product, which sells for only \$50, could be used by stalkers, professional hackers and industrial spies, and probably, as well, by some of the world's law enforcement agencies (which likely have their own versions that can be installed over the air).

Propagation Mechanisms

Launching a malware attack involves two phases, analogous to delivering a round of explosives on an enemy installation: you have to build the cannon shell or bomb, but you also need an aircraft or artillery weapon to deliver it to the location.

The attacker needs to create his malware and also needs a propagation mechanism. Most of the methods are well known to experienced IT personnel; among them are -

1. E-mail — Sending the code embedded in an e-mail attachment, exploiting a vulnerability in Outlook or another e-mail client, or inducing a user to visit a particular web site that exploits a client-side vulnerability. The most widely used method employs social engineering to trick the unsuspected victim to run an executable.

2. Exploiting software flaws — Virus writers often use a recent vulnerability that exploits a systems service (which runs in the context of a privileged account). The attacker inserts arbitrary code that's ordinarily executed with administrator rights. This payload may download other malware code, cause a denial of service attack, or allow the attacker to control the computer over a communications channel such as IRC. Exploiting technical vulnerabilities is the method used by most of the notorious worms in recent years such as Code Red, Blaster, and SQL Slammer.
3. CD — Planting malicious code on a CD, with a hardcopy label that tempts the person receiving the CD to install it; the autorun feature, if it has not been disabled on the user's computer by the user or an administrator, loads the malware surreptitiously when the CD is put into a drive.
4. USB flash drives — Only specially manufactured USB drives with special firmware emulate a CDROM's autorun feature; a drive from the Japanese company Hagiwara is one that supports this feature.
5. Physical access — A person with access to company computers, such as an authorized visitor, a social engineer who infiltrates the company premises, or a disgruntled employee, installs malware on one or more computers. The likely malware in this instance would be a keystroke logger that acts like a computer wiretap.

Additional propagation vectors include using peer-to-peer networks for file sharing, downloaded freeware, and especially downloaded cracks for licensed software. The attacker can also exploit a weakness with Windows Wireless Configuration Service to trick the target into connecting to the attacker's wireless network. Once the connection is established, the attacker would then be able to perform what's called a "man-in-the-middle" attack that exploits a client-side vulnerability.

Propagation by Social Engineering

Social Engineering has been defined as "using influence and persuasion to deceive people by convincing them that the attacker is someone he is not, or by manipulation. As a result, the social engineer is able to take advantage of people to obtain information, or to persuade them to perform an action item, with or without the use of technology." (*The Art of Deception*, Mitnick & Simon, Wiley, 2002)

Social engineering can be used to steal data, but it can also be used to trick employees into an action that allows malware to be planted on their computer, as described earlier. Tactics include leaving a CD bearing a tempting label, such as "Payroll Salary History - Q1 2006" on the floor as if dropped, or sending an e-mail that appears to come from a friend or co-worker, containing an attachment with a title that seems too

tempting to pass up - "Here is the receipt for the order you recently placed." or "Here's a copy of your personal information that has been exposed on the Internet" or "Your online bank account has been compromised." Another attack might masquerade as an e-mail from Microsoft claiming the attachment is a critical security patch.

The common theme of these social engineering tactics is that they are based on a very powerful influence tactic known as *scarcity*. The main principal of scarcity is that people desire what they cannot have, wanting things that are rare or out of reach. Another application of this principle is the *fear of losing*. According to studies conducted by social scientists, people are more motivated to avoid loss than to realize a gain. Numerous malware attacks are based on one or the other of these methods of social influence.

While methods of blocking malware will continue to improve as technologies are developed, social engineering will always remain a substantial threat; apparently, finding a gullible employee is never too difficult. Therefore, it's important that technology be used whenever possible to take away the decision-making about the handling of sensitive information or action items from unsuspecting employees.

Countermeasures

The first line of defense for your company in protecting against malware is your antivirus software. It's a critical component. Antivirus packages use different tactics to identify potentially malicious software: signatures, heuristics, and virtual machine technology. In some cases, the virus writers use techniques to mask the malicious code by using packers, binders, compression, and encryption technologies.

While AV software is essential, it's important to recognize that it can only be effective against a new virus once the antivirus software companies become aware of the new attack, identify its signature, update their definitions package to include the new signature, and the new definitions are downloaded by your company and installed.

What's needed in addition are solutions that can stop **unknown malicious code**. The main available approaches are these -

Restricting administrative rights

Company employees other than IT personnel can have administrative rights restricted on their computers, establishing an environment in which no one is running with administrator rights unless doing administrative

functions. I believe that allowing user computers to run with administrator rights is one of the biggest reasons malware attacks are so successful.

Disabled active content (“Drive-by downloads”)

Another method of propagating malware is to use Active X controls that appear to be legitimate to the end-user. In older versions of Windows, the attacker can manipulate the text within the dialog box presented to the user to social engineer the target into accepting the malicious control.

Disabling active content features within the browser software will mitigate this threat; however, since this will greatly limit the user’s ability to surf the Net, an alternative is to set up particular security zones that allow active content for trusted sites.

Self-healing

Once infected with malware, the ***only sure way*** to secure the system is reinstalling the operating system from scratch. To avoid the need for this extreme measure, systems can be protected by the use of “self-healing” products, which monitor the system to detect any unauthorized changes to files, processes, services, or the registry. Once unauthorized changes are detected, the system is returned to the original state. Note, however, that virus writers also use the same self-healing technology to prevent their malware from being removed.

Multiple anti-spyware utilities

There are many anti-spyware software tools on the market. Unfortunately, using one product may not be sufficient; in general, each of these products successfully detects some spyware that the others do not. Consider using more than one anti-spyware product.

Digital signatures

One countermeasure to preventing execution of malicious code is to digitally sign all files containing executable code, such as authorized applications, scripts, and DLLs. Once the signatures are recorded, a software tool can check the signature prior to allowing execution of the code.

The algorithms most commonly used for computing digital signatures are based on MD5 or SHA-1 message digests. Depending on the third-party product being used, creating these signatures may be time

consuming. In selecting a product, it's important to evaluate how this feature is supported, to determine the overall burden of managing the effort.

Black List

When a user attempts to launch an application, open an e-mail attachment, or execute other code, prior to the execution of code, a third-party application could check a Black List for particular filenames or signatures. This is similar to the approach used by AV software to block known malware from executing.

However, using this approach successfully presents a difficult challenge since the enterprise must be extremely vigilant about updating the lists. This function is best served through the use of AV software, which, if desired, can be supplemented by manual updates and changes.

In certain circumstances, businesses may restrict particular users, groups, or computers from running particular application or scripts, even if not considered malware. However, another approach calls for hardening the workstations, which includes strictly controlling the permissions on files and directories.

White List

A White List is used to control what applications, scripts, DLLs, or other executable code is allowed to run. These White Lists can be granular enough to identify which specific applications can be executed by a particular user, group, computer, or organizational unit.

For small businesses, using a White List approach is one of the best available methods to prevent execution of malware based on attacks involving social engineering. Attacks involving exploiting system services or local kernel exploits would require other mitigation strategies.

Managing these lists can be quite burdensome and the effort does not scale well in large organizations. The significant challenge here is to develop a process that reduces the complexity of managing the lists.

Trusted Owners List

File ownership can be used as a basis for another method of blocking malware attacks. For example, in a mass-mailing attack designed to exploit the gullibility of the end-user, when a user is successfully duped into opening an attachment or downloading an application that contains malicious code, the file containing the

executable code will be designated by the operating system as being owned by the user who downloaded it.

Taking advantage of the file ownership property, a third-party software tool could check the file owner against a list of authorized software installers using a previously created Trusted Owners List—designating those people who are allowed to install applications and scripts. Only those programs owned by a Trusted Owner are authorized to be executed. This approach can reduce malware attacks targeted to general users.

Endpoint Blocking

New technologies developed in the past few years now make it possible to block malicious software *at the computer's endpoint*, thus ending concerns about employees bringing in software on external media such as CDROM, USB flash drives, PCMCIA hard drives, and external Firewire drives. Most desirable are software products that enable administrators to maintain granular control over the endpoints to prevent users from connecting any unauthorized devices.

Current approaches now make possible various system-wide “lock-down” steps to prevent the transfer of files or data to external devices such as a USB flash-drive. When desired, specified users can be exempted depending on the configuration of the access control software. While there is a tendency of corporate IT security personnel to think in terms of protecting mainly against the electronic intruder, it's important to remember that many instances of information theft are the work of disgruntled employees who have easy access not only to the data on their own computers but to data that can be transferred from corporate file servers, or the computer of another employee while that person is away from his or her desk.

Enabling DEP

Many notable worm attacks simply exploit a vulnerability that came into wide use after being published on the Internet. Worms such as Code Red, MS Blaster, Nimda, and SQL Slammer, among others, exploit buffer overflows to gain control of the targeted system.

While the issue of malware attacks that propagate through exploitation of vulnerabilities such as buffer overflows is beyond the scope of this paper, two defense strategies that are effective against many attacks of this type are worth mentioning —

- Eliminating unnecessary services, or at least using packet filtering to block access to services that do not need to be exposed in order to reduce the attack surface; and,

Enabling DEP (Data Execution Prevention) for all programs and services under Windows XP Pro SP2 and Windows 2003. DEP was intended to ensure that program instructions can only run in memory pages marked for execution, to block heap buffer overflow attacks. (While enabling DEP is still a valuable security measure, it's important to note that a method of defeating this protection has been developed and is, in fact, posted on the Internet.⁸)

Auditing security events

Standard security practices that involve monitoring and alerting of security events at the gateway are effective but fill only one part of the real-time alerting need. To remain fully on top of the moment-by-moment situation, security staff also needs to be alerted to events at individual workstations, such as an attempt to execute unauthorized applications or scripts, attempts to use unauthorized devices at the endpoints, or attempts to execute programs that do not exist on the White List.

Disabling application features

Some available approaches that provide added security need to be weighed against the degree to which they may produce a negative effect on employee productivity. One of these is the ability to disable certain features of a specified application. Using the least-privilege rule may prevent malicious end-users from abusing applications to obtain unauthorized access but may inconvenience employees to an unacceptable degree.

Conclusion: Mitigating Common Malware Attacks

Environments that allow users to run with local administrator rights on their work stations are playing with fire. Once a user running with administrator rights gets fooled into executing code, the game is over. So the #1 malware mitigation strategy for **users** is to make sure they never do their ordinary work using an administrator account. Other operating systems platforms such as Linux are managed better right out of the box. The **root** account (equivalent to a Windows administrator) is only used for administrative functions. Windows users should follow the same protocol.

In small environments, using a combination of granularly configured White Lists *in combination* with trusted ownership should substantially reduce the number of successful malware attacks propagated by means of social engineering (such as mass-mailing with malicious attachments, or hyperlinks to booby-trapped web sites that exploit client-side vulnerabilities.)

In medium to large environments, I recommend using a combination of digital signatures and trusted file ownership to reduce the burden of maintaining granularly configured White Lists.

In today's environment, IT departments cannot afford to rely on AV software and intelligent firewalls as the only means of protecting against malware attacks and the theft of data.

Attackers are constantly creating new malware and discovering new approaches. IT departments need to be embracing the most powerful protective measures available.

Sponsored by

www.AppSense.com

Endnotes and references:

- ¹. While the basic elements of this story are true, I have guessed at some of the details not yet made public. Some of the names are fictitious; the culprits are identified by their real names.
- ². Copyright 2006 Computer Economics, Inc.
- ³. http://blog.washingtonpost.com/securityfix/2006/06/microsoft_releases_malware_sta.html
- ⁴. A “bot” — short for robot — refers to a computer that has been surreptitiously commandeered by an attacker; when the attacker has amassed a sizeable number of bots, forming a “botnet,” he is able to control each system to start a barrage of traffic from the many bots to overload the target server, shutting it down for a period of time, resulting in a “denial of service.” Using botnets, attackers can send unsolicited email, commit point-and-click fraud, engage in trust fraud, or engage in other types of malicious activity. The attacker’s activities are ordinarily invisible to the owners of the computers, who can continue using their machines as usual, unaware of the actions being controlled by the hacker; this type of attacker is sometimes called a “botmaster” or “bot-herder.”
- ⁵. <http://www.securityfocus.com/news/9411>
- ⁶. <http://www.cnn.com/2006/TECH/internet/01/31/furst/>
- ⁷. http://www.lurhq.com/managed_security_services.html
- ⁸. <http://www.maxpatrol.com/defeating-xpsp2-heap-protection.pdf>