Computer Viruses

By Stuart Alden

Introduction

In the past decade, computer and networking technology has seen enormous growth. This growth however, has not come without a price. With the advent of the "Information Highway", as it's coined, a new methodology in crime has been created. Electronic crime has been responsible for some of the most financially devastating victimizations in society.

In the recent past, society has seen malicious editing of the Justice Department web page (1), unauthorized access into classified government computer files, phone card and credit card fraud, and electronic embezzlement. All these crimes are committed in the name of "free speech." These new breed of criminals claim that information should not be suppressed or protected and that the crimes they commit are really not crimes at all. What they choose to deny is that the nature of their actions are slowly consuming the fabric of our country's moral and ethical trust in the information age.

Federal law enforcement agencies, as well as commercial computer companies, have been scrambling around in an attempt to "educate" the public on how to prevent computer crime from happening to them. They inform us whenever there is an attack, provide us with mostly ineffective anti-virus software, and we are left feeling isolated and vulnerable. I do not feel that this defensive posture is effective because it is not pro-active. Society is still being attacked by highly skilled computer criminals of which we know very little about them, their motives, and their tools of the trade. Therefore, to be effective in defense, we must understand how these attacks take place from a technical stand-point. To some degree, we must learn to become a computer criminal. Then we will be in a better position to defend against these victimizations that affect us on both the financial and emotional level. In this paper, we will explore these areas of which we know so little, and will also see that computers are really extensions of people. An attack on a computer's vulnerabilities are really an attack on peoples' vulnerabilities.

Today, computer systems are under attack from a multitude of sources. These range from malicious code, such as viruses and worms, to human threats, such as hackers and phone "phreaks." These attacks target different characteristics of a system. This leads to the possibility that a particular system is more susceptible to certain kinds of attacks.

Malicious code, such as viruses and worms, attack a system in one of two ways, either internally or externally. Traditionally, the virus has been an internal threat (an attack from within the company), while the worm, to a large extent, has been a threat from an external source (a person attacking from the outside via modem or connecting network).

Human threats are perpetrated by individuals or groups of individuals that attempt to penetrate systems through computer networks, public switched telephone networks or other sources. These attacks generally target known security vulnerabilities of systems.

Many of these vulnerabilities are simply due to configuration errors.

Malicious Code

Viruses and worms are related classes of malicious code; as a result they are often confused. Both share the primary objective of replication. However, they are distinctly different with respect to the techniques they use and their host system requirements. This distinction is due to the disjoint sets of host systems they attack. Viruses have been almost exclusively restricted to personal computers, while worms have attacked only multi-user systems.

A careful examination of the histories of viruses and worms can highlight the differences and similarities between these classes of malicious code. The characteristics shown by these histories can be used to explain the differences between the environments in which they are found. Viruses and worms have very different functional requirements; currently no class of systems simultaneously meets the needs of both.

A review of the development of personal computers and multi-tasking workstations will show that the gap in functionality between these classes of systems is narrowing rapidly. In the future, a single system may meet all of the requirements necessary to support both worms and viruses. This implies that worms and viruses may begin to appear in new classes of systems. A knowledge of the histories of viruses and worms may make it possible to predict how malicious code will cause problems in the future.

Basic Definitions

To provide a basis for further discussion, the following definitions will be used throughout the report;

- *Trojan Horse* - a program which performs a useful function, but also performs an unexpected action as well;
- *Virus* - a code segment which replicates by attaching copies to existing executables;
- *Worm* - a program which replicates itself and causes execution of the new copy and
- *Network Worm* - a worm which copies itself to another system by using common network facilities, and causes execution of the copy on that system.

In essence, a computer program which has been infected by a virus has been converted into a "trojan horse". The program is expected to perform a useful function, but has the unintended side effect of viral code execution. In addition to performing the unintended task, the virus also performs the function of replication. Upon execution, the virus attempts to replicate and "attach" itself to another program. It is the unexpected and uncontrollable replication that makes viruses so dangerous. As a result, the host or victim computer falls prey to an unlimited amount of damage by the virus, before anyone realizes what has happened.

Viruses are currently designed to attack single platforms. A platform is defined as the combination of hardware and the most prevalent operating system for that hardware. As an example, a virus can be referred to as an IBM-PC virus, referring to the hardware, or a DOS virus, referring to the operating system. "Clones" of systems are also included with the original platform.

History of Viruses

The term "computer virus" was formally defined by Fred Cohen in 1983, while he performed academic experiments on a Digital Equipment Corporation VAX system. Viruses are classified as being one of two types: research or "in the wild." A research virus is one that has been written for research or study purposes and has received almost no distribution to the public. On the other hand, viruses which have been seen with any regularity are termed "in the wild." The first computer viruses were developed in the early 1980s. The first viruses found in the wild were Apple II viruses, such as Elk Cloner, which was reported in 1981 [Den90]. Viruses were found on the following platforms:

- Apple II
- IBM PC
- Macintosh
- Atari
- Amiga

These computers made up a large percentage of the computers sold to the public at that time. As a result, many people fell prey to the Elk Cloner and virus's similar in nature. People suffered losses in data from personal documents to financial business data with little or no protection or recourse.

Viruses have "evolved" over the years due to efforts by their authors to make the code more difficult to detect, disassemble, and eradicate. This evolution has been especially apparent in the IBM PC viruses; since there are more distinct viruses known for the DOS operating system than any other.

The first IBM-PC virus appeared in 1986 [Den90]; this was the Brain virus. Brain was a boot sector virus and remained resident in the computer until "cleaned out". In 1987, Brain was followed by Alameda (Yale), Cascade, Jerusalem, Lehigh, and Miami (South African Friday the 13th). These viruses expanded the target executables to include COM and EXE files. Cascade was encrypted to deter disassembly and detection. Variable encryption appeared in 1989 with the 1260 virus. Stealth viruses, which employ various techniques to avoid detection, also first appeared in 1989, such as Zero Bug, Dark Avenger and Frodo (4096 or 4K). In 1990, self-modifying viruses, such as Whale were introduced. The year 1991 brought the GP1 virus, which is "network-sensitive" and attempts to steal Novell NetWare passwords. Since their inception, viruses have become increasingly complex and equally destructive.

Examples from the IBM-PC family of viruses indicate that the most commonly detected viruses vary according to continent, but Stoned, Brain, Cascade, and members of the Jerusalem family, have spread widely and continue to appear. This implies that highly survivable viruses tend to be benign, replicate many times before activation, or are somewhat innovative, utilizing some technique never used before in a virus.

Personal computer viruses exploit the lack of effective access controls in these systems.

The viruses modify files and even the operating system itself. These are "legal" actions within the context of the operating system. While more stringent controls are in place on multi-tasking, multi-user operating systems (LAN Networks or Unix), configuration errors, and security holes (security bugs) make viruses on these systems more than theoretically possible. This leads to the following initial conclusions:

- Viruses exploit weaknesses in operating system controls and human patterns of system use/misuse;
- Destructive viruses are more likely to be eradicated and
- An innovative virus may have a larger initial window to propagate before it is discovered and the "average" anti-viral product is modified to detect or eradicate it. If we reject the hypothesis that viruses do not exist on multi-user systems because they are too difficult to write, what reasons could exist? Perhaps the explosion of PC viruses (as opposed to other personal computer systems) can provide a clue. The population of PCS and PC compatible is by far the largest. Additionally, personal computer users exchange disks frequently. Exchanging disks is not required if the systems are all connected to a network. In this case large numbers of systems may be infected through the use of shared network resources.

One of the primary reasons that viruses have not been observed on multi-user systems is that administrators of these systems are more likely to exchange source code rather than executables. They tend to be more protective of copyrighted materials, so they exchange locally developed or public domain software. It is more convenient to exchange source code, since differences in hardware architecture may preclude exchanging executables. It is this type of attitude towards network security that could be viewed as victim precipitation. The network administrators place in a position to be attacked, despite the fact that they are unaware of the activity. The following additional conclusions can be made:

- To spread, viruses require a large population of similar systems and exchange of executable software;

Destructive viruses are more likely to be eradicated;

- An innovative virus may have a larger initial window to propagate before it is discovered and the "average" anti-viral product is modified to detect or eradicate it.

Preventive Action

Although many anti-virus tools and products are now available, personal and administrative practices and institutional policies, particularly with regard to shared or external software usage, should form the first line of defense against the threat of virus attack. Users should also consider the variety of anti-virus products currently available.

There are three classes of anti-virus products: detection tools, identification tools, and removal tools. Scanners are an example of both detection and identification tools. Vulnerability monitors and modification detection programs are both examples of detection tools. Disinfectors are examples of a removal tools. A detailed description of the tools is provided below.

Scanners and disinfectors, the most popular classes of anti-virus software, rely on a great deal of a prior knowledge about the viral code. Scanners search for "signature strings" or use algorithmic detection methods to identify known viruses. Disinfectors rely on substantial information regarding the size of a virus and the type of modifications to restore the infected file's contents.

Vulnerability monitors, which attempt to prevent modification or access to particularly sensitive parts of the system, may block a virus from hooking sensitive interrupts. This requires a lot of information about "normal" system use, since personal computer viruses do not actually circumvent any security features. This type of software also requires decisions from the user.

Modification detection is a very general method, and requires no information about the virus to detect its presence. Modification detection programs, which are usually checksum based, are used to detect virus infection or Trojan horses. This process begins with the creation of a baseline, where checksums for clean executables are computed and saved. Each following iteration consists of checksum computation and comparison with the stored value. It should be noted that simple checksums are easy to defeat; cyclical redundancy checks (CRC) are better, but can still be defeated; cryptographic checksums provide the highest level of security.

Worms

The following are necessary characteristics of a worm:

- replication
- self-contained; does not require a host
- activated by creating process (needs a multi-tasking system)
- for network worms, replication occurs across communication links

A worm is not a Trojan horse; it is a program designed to replicate. The program may perform any variety of additional tasks as well. The first network worms were intended to perform useful network management functions [SH82]. They took advantage of system properties to perform useful action. However, a malicious worm takes advantage of the same system properties. The facilities that allow such programs to replicate do not always discriminate between malicious and good code.

Protecting a system against a worm requires a combination of basic system security and good network security. There are a variety of procedures and tools which can be applied to protect the system.

In basic system security, the most important means of defense against worms is the identification &authentication (I&A) controls, which are usually integrated into the system. If poorly managed, these controls become a vulnerability which is easily exploited. Worms are especially adept at exploiting such vulnerabilities; both the Internet and DECnet worms targeted I&A controls.

Add-on tools include configuration review tools (such as COPS [GS91] for UNIX systems) and checksum-based change detection tools. Design of configuration review tools requires intimate knowledge of the system, but no knowledge of the worm code.

Another class of add-on tools is the intrusion detection tool. This is somewhat analogous to the PC monitoring software, but is usually more complex. This tool reviews series of commands to determine if the user is doing something suspicious. If so, the system manager is notified.

One type of network security tool is the wrapper program. Wrapper programs can be used to "filter" network connections, rejecting or allowing certain types of connections (or connections from a pre-determined set of systems). This can prevent worm infections by "untrusted" systems. These tools do not protect a system against the exploitation of flaws in the operating system. This issue must be dealt with at the time of procurement. After procurement, it becomes a procedural issue. Resources are available to system managers to keep them abreast of security bugs and bug fixes, such as the CERT computer security advisories.

Another class of security tools which are widely employed today to protect a network against worms are the Firewall . The firewall system [GS91] protects an organizational network from systems in the larger network world. Firewall systems are found in two forms: simple or intelligent. An intelligent firewall filters all connections between hosts on the organizational network and the world-at-large. A simple firewall disallows all connections with the outside world, essentially splitting the network into two different networks. To transfer information between hosts on the different networks, an account on the firewall system is required.

Human Threats

Insiders, hackers and "phone phreaks" are the main components of the human threat factor. Insiders are legitimate users of a system. When they use that access to circumvent security, that is known as an insider attack. Hackers are the most widely known human threat. Hackers are people who enjoy the challenge of breaking into systems. "Phreakers" are hackers whose main interest is in telephone systems.

The primary threat to computer systems has traditionally been the insider attack. Insiders are likely to have specific goals and objectives, and have legitimate access to the system. Insiders can plant trojan horses or browse through the file system. This type of attack can be extremely difficult to detect or protect against.

The insider attack can affect all components of computer security. Browsing attacks the confidentiality of information on the system. Trojan horses are a threat to both the integrity and confidentiality of the system. Insiders can affect availability by overloading the system's processing or storage capacity, or by causing the system to crash.

These attacks are possible for a variety of reasons. On many systems, the access control settings for security-relevant objects do not reflect the organization's security policy. This allows the insider to browse through sensitive data or plant that trojan horse. The insider exploits operating system bugs to cause the system to crash. The actions are undetected because audit trails are inadequate or ignored.

Hackers

The definition of the term "hacker" has changed over the years. A hacker was once thought of as any individual who enjoyed getting the most out of the the system he was using. A hacker would use a system extensively and study the system until he became proficient in all its nuances. This individual was respected as a source of information for local computer users; someone referred to as a "guru" or "wizard." Now, however, the term hacker is used to refer to people who either break into systems for which they have no authorization or intentionally overstep their bounds on systems for which they do have legitimate access.

Methods used by hackers to gain unauthorized access to systems include:

- Password cracking
- Exploiting known security weaknesses
- Network spoofing
- "Social Engineering"

The most common techniques used to gain unauthorized system access involve password cracking and the exploitation of known security weaknesses. Password cracking is a technique used to surreptitiously gain system access by using another users account. Users often select weak password. The two major sources of weakness in passwords are easily guessed passwords based on knowledge of the user (e.g. wife's maiden name) and passwords that are susceptible to dictionary attacks (i.e. brute-force guessing of passwords using a dictionary as the source of guesses).

Another method used to gain unauthorized system access is the exploitation of known security weaknesses. Two type of security weaknesses exist: configuration errors, and security bugs. There continues to be an increasing concern over configuration errors. Configuration errors occur when a the system is set up in such a way that unwanted exposure is allowed. Then, according to the configuration, the system is at risk from even legitimate actions. An example of this would be that if a system "exports" a file system to the world (makes the contents of a file system available to all other systems on the network) , then any other machine can have full access to that file system. Security bugs occur when unexpected actions are allowed on the system due to a loophole in some application program. An example would be sending a very long string of keystrokes to a screen locking program, thus causing the program to crash and leaving the system inaccessible.

A third method of gaining unauthorized access is network spoofing. In network spoofing a system presents itself to the network as though it were a different system (system A impersonates system B by sending B's address instead of its own). The reason for doing this is that systems tend to operate within a group of other "trusted" systems. Trust is imparted in a one-to-one fashion; system A trusts system B (this does not imply that system B trusts system A). Implied with this trust, is that the system administrator of the trusted system is performing his job properly and maintaining an appropriate level of security for his system. Network spoofing occurs in the following manner. If system A trusts system B, and system C spoofs (impersonates) system B, then system C can gain otherwise denied access to system A. The system's integrity is compromised because it would allow a foreign system to mimic a friendly system, hence allowing access.

"Social engineering" is the final method of gaining unauthorized system access. People have been known to call a system operator, pretending to be some authority figure, and demand that a password be changed to allow them access. One could also say that using personal data to guess a user's password is social engineering.

Phone Phreaks

The "phone phreak" (phreak for short) is a specific breed of hacker. A phreak is someone who displays most of the characteristics of a hacker, but also has a specific interest in the phone system and the systems that support its operations. Additionally, most of the machines on the Internet, itself a piece of the Public Switched Network, are linked together through dedicated, commercial phone lines. A talented phreak is a threat to not only the phone system, but to the computer networks it supports.

There are two advantages of attacking systems through the phone system. The first advantage is that, phone system attack are hard to trace. It is possible to make connections through multiple switching units or to use unlisted or unused phone numbers to confound a tracing effort. Also by being in the phone system, it is sometimes possible to monitor the phone company to see if a trace is initiated.

The second advantage to using the phone system is that a sophisticated host machine is not needed to originate an attack nor is direct access to the network to which the target system is attached. A simple dumb terminal connected to a modem can be used to initiate an attack. Often, an attack consists of several hops, a procedure whereby one system is broken into and from that system another system is broken into, etc. This again makes tracing more difficult.

Configuration Errors and Passwords

Today, desktop workstations are becoming the tool of more and more scientists and professionals. Without proper time and training to administer these systems, vulnerability to both internal and external attacks will increase. Workstations are usually administered by individuals whose primary job description is not the administration of the workstation. The workstation is merely a tool to assist in the performance of the actual job tasks. As a result, if the workstation is up and running, the individual is satisfied.

This neglectful and permissive attitude toward computer security can be very dangerous. This user-attitude has resulted in poor usage of controls and selection of easily guessed passwords. As these users become, in effect, workstation administrators, this will be compounded by configuration errors and a lax attitude towards security bug fixes. To correct this, systems should be designed so that security is the default and personnel should be equipped with adequate tools to verify that their systems are secure.

Of course, even with proper training and adequate tools threats will remain. New security bugs and attack mechanisms will be employed. Proper channels do not currently exist in most organizations for the dissemination of security related information. If organizations do not place a high enough priority on computer security, the average system will continue to be at risk from external threats.

Internal Threats

System controls are not well matched to the average organization's security policy. As a direct result, the typical user is permitted to circumvent that policy on a frequent basis. The administrator is unable to enforce the policy because of the weak access controls, and cannot detect the violation of policy because of weak audit mechanisms. Even if the audit mechanisms are in place, the volume of data produced makes it unlikely that the administrator will detect policy violations.

Ongoing research in integrity and intrusion detection promise to fill some of this gap. Until these research projects become available as products, systems will remain vulnerable to internal threats.

Information Dissemination

The Forum of Incident Response and Security Teams (FIRST), an organization whose members work together voluntarily to deal with computer security problems and their prevention, has established valuable channels for the dissemination of security information. It is now possible to obtain security bug fix information in a timely fashion. The percentage of system administrators receiving this information is still low, but is improving daily.

Hackers continue to make better use of the information channels than the security community. Publications such as "Phrack" and "2600" are well established and move information effectively throughout the hacking community. Bulletin boards and Internet archive sites are available to disseminate virus code, hacking information, and hacking tools.

Conclusion

Poor administrative practices and the lack of education, tools, and controls combine to leave the average system vulnerable to attack. Research promises to alleviate the inadequate supply of tools and applicable controls. These controls, however, tend to be add-on controls. There is a need for the delivery of secure systems, rather than the ability to build one from parts. The average administrator has little inclination to perform these modifications, and no idea how to perform them. As long as this occurs, hackers, phreakers, and other malicious users will continue to prey on these systems.

Also, extensive connectivity increases system access for hackers. Until standards become widely used, network security will continue to be handled on a system by system basis. The problem can be expected to increasewithout appropriate security capabilities.

A promising note for the future does exist. Multiple sets of tools do not need to be developed in order to solve each of the potential threats to a system.

Many of the controls that will stop one type of attack on a system will be beneficial against many other forms of attack. The challenge is to determine what is the minimum set of controls necessary to protect a system with an acceptable degree of assurance.

System users and administrators must also educate themselves on a continuing basis. Only this way will they be able to remain current in methods of preventive action against hacker and electronic criminal activity activity. State educational institutions are receiving thousands of federal dollars each year to promote better computer and network understanding. It is their responsibility to promote this education throughout the schools and in the community. Society is still seeing the infancy of computers, not just in its general growth, but in its capabilities of controlling every facet of our normal lives. If we do not attempt to broaden our awareness of computer science, we will continue to become victims of electronic attacks. Hopefully, after reading this report you have a better understanding of what future lies before us and what we must do to keep its integrity intact.