Name of Faculty: Sushil Kumar

Designation: Professor

Department: Information Technology

Subject: Basic Computer Engineering (BT-205)

Unit: IV

Topic: Viruses and Related Threats

Sem: II

Section: A4/A5

Year: Ist Year

VIRUSES AND RELATED THREATS

Perhaps the most sophisticated types of threats to computer systems are presented by programs that exploit vulnerabilities in computing systems.

What Are Computer Viruses and Related Threats?

Computer viruses are the most widely recognized example of a class of programs written to cause some form of intentional damage to computer systems or networks. A computer virus performs two basic functions: it copies itself to other programs, thereby infecting them, and it executes the instructions the author has included in it. Depending on the author's motives, a program infected with a virus may cause damage immediately upon its execution, or it may wait until a certain event has occurred, such as a particular date and time. The damage can vary widely, and can be so extensive as to require the complete rebuilding of all system software and data. Because viruses can spread rapidly to other programs and systems, the damage can multiply geometrically.

Related threats include other forms of destructive programs such as Trojan horses and network worms. Collectively, they are sometimes referred to as malicious software. These programs are often written to masquerade as useful programs, so that users are induced into copying them and sharing them with friends and work colleagues. The malicious software phenomena is fundamentally a people problem, as it is authored and initially spread by individuals who use systems in an unauthorized manner. Thus, the threat of unauthorized use, by unauthorized and authorized users, must be addressed as a part of virus prevention.

What Are the Vulnerabilities They Exploit?

Unauthorized users and malicious software may gain access to systems through inadequate system security mechanisms, through security holes in applications or systems, and through weaknesses in computer management, such as the failure to properly use existing security mechanisms. Malicious software can be copied intentionally onto systems, or be spread when users unwittingly copy and share infected software obtained from public software repositories, such as software bulletin boards and shareware. Because malicious software often hides its destructive nature by performing or claiming to perform some useful function, users generally don't suspect that they are copying and spreading the problem.

Why Are Incidents of Viruses and Related Threats On the Rise?

Viruses and related threats, while not a recent phenomena, have had relatively little attention focused on them in the past. They occurred less frequently and caused relatively little damage. For these reasons, they were frequently treated lightly in computer design and by management, even though their potential for harm was known to be great.

Computer users have become increasingly proficient and sophisticated. Software applications are increasingly complex, making their bugs and security loopholes more difficult to initially detect and correct by the manufacturer. In conjunction with these two factors, some brands of software are now widely used, thus their bugs and security loopholes are often known to users. With the widespread use of personal computers that lack effective security mechanisms, it is relatively easy for knowledgeable users to author malicious software and then dupe unsuspecting users into copying it.

Steps Toward Reducing Risk

Organizations can take steps to reduce their risk to viruses and related threats. Some of the more important steps are outlined below.

- Include the damage potential of viruses, unauthorized use, and related threats in risk analysis and contingency planning. Develop a plan to deal with potential incidents.
- Make computer security education a prerequisite to any computer use. Teach users how to protect their systems and detect evidence of tampering or unusual activity.
- Ensure that technically oriented security and management staff are in place to deal with security incidents.
- Use the security mechanisms that exist in your current software. Ensure that they are used correctly. Add to them as necessary.
- Purchase and use software tools to aid in auditing computing activity and detecting the presence of tampering and damage.

A Brief Overview on Viruses and Related Threats

The term computer virus is often used in a general sense to indicate any software that can cause harm to systems or networks. However, computer viruses are just one example of many different but related forms of software that can act with great speed and power to cause extensive damage - other important examples are Trojan horses and network worms. In this document, the term malicious software refers to such software.

Trojan Horses

A Trojan horse' program is a useful or apparently useful program or command procedure containing hidden code that, when invoked, performs some unwanted function. An author of a Trojan horse program might first create or gain access to the source code of a useful program that is attractive to other users, and then add code so that the program performs some harmful function in addition to its useful function. A simple example of a Trojan horse program might be a calculator program that performs functions similar to that of a pocket calculator. When a user invokes the program, it appears to be performing calculations and nothing more, however it may also be quietly deleting the user's files, or performing any number of harmful actions. An example of an even simpler Trojan horse program is one that performs only a harmful function, such as a program that does nothing but delete files. However, it may appear to be a useful program by having a name such as calculator or something similar to promote acceptability.

Trojan horse programs can be used to accomplish functions indirectly that an unauthorized user could not accomplish directly. For example, a user of a multi-user system who wishes to gain access to other users' files could create a Trojan horse program to circumvent the users' file security mechanisms. The Trojan horse program, when run, changes the invoking user's file permissions so that the files are readable by any user. The author could then induce users to run this program by placing it in a common directory and naming it such that users will program is a useful utility. After a user runs the program, the author can then access tile information in the user's files, which in this example could be important work or personal Information. Affected users may not notice the changes for long periods of time unless they are very observant.

An example of a Trojan horse program that would be very difficult to detect would be a compiler on a multi-user system that has been modified to insert additional code into certain programs as they are compiled, such as a login program. The code creates a trap door in the login program which permits the Trojan horse's author to log onto the system using a special password.

Whenever the login program is recompiled, the compiler will always insert the trap door code into the program, thus the Trojan horse code can never be discovered by reading the login program' source code. For more information on this example, see [thompson84].

Trojan horse programs are introduced into systems in two ways: they are initially planted, and unsuspecting users copy and run them. They are planted in software repositories that many people can access, such as on personal computer network servers, publicly-accessible directories in a multiuser environment, and software bulletin boards. Users are then essentially duped into copying Trojan horse programs to their own systems or directories. If a Trojan horse program performs a useful function and causes no immediate or obvious damage, a user may continue to spread it by sharing the program with other friends and co-workers. The compiler that copies hidden code to a login program might be an example of a deliberately planted Trojan horse that could be planted by an authorized user of a system, such as a user assigned to maintain compilers and software tools.

Computer Viruses

Computer viruses, like Trojan horses, are programs that contain hidden code which performs some usually unwanted function. Whereas the hidden code in a Trojan horse program has been deliberately placed by the program's author, the hidden code in a computer virus program has been added by another program, that program itself being a computer virus or Trojan horse. Thus, computer viruses are programs that copy their hidden code to other programs, thereby infecting them. Once infected, a program may continue to infect even more programs. In due time, a computer could be completely overrun as the viruses spread in a geometric manner.

An example illustrating how a computer virus works might be an operating system program for a personal computer, in which an infected version of the operating system exists on a diskette that contains an attractive game. For the game to operate, the diskette must be used to boot the computer, regardless of whether the computer contains a hard disk with its own copy of the (uninfected) operating system program. When the computer is booted using the diskette, the infected program is loaded into memory and begins to run. It immediately searches for other copies of the operating system program, and finds one on the hard disk. It then copies its hidden code to the program on the hard disk. This happens so quickly that the user may not notice the slight delay before his game is run. Later, when the computer is booted using the hard disk, the newly infected version of the operating system will be loaded into memory. It will in turn look for copies to infect. However, it may also perform any number of very destructive actions, such as deleting or scrambling all the files on the disk.

A computer virus exhibits three characteristics: a replication mechanism, an activation mechanism, and an objective. The replication mechanism performs the following functions:

- Searches for other programs to infect when it finds a program, possibly determines whether the program has been previously infected by checking a flag.
- Inserts the hidden instructions somewhere in the program
- modifies the execution sequence of the program's instructions such that the hidden code will be executed whenever the program is invoked

- Possibly creates a flag to indicate that the program has been infected

The flag may be necessary because without it, programs could be repeatedly infected and grow noticeably large. The replication mechanism could also perform other functions to help disguise that the file has been infected, such as resetting the program file's modification date to its previous value, and storing the hidden code within the program so that the program's size remains the same.

The activation mechanism checks for the occurrence of some event. When the event occurs, the computer virus executes its objective, which is generally some unwanted, harmful action. If the activation mechanism checks for a specific date or time before executing its objective, it is said to contain a time bomb. If it checks for a certain action, such as if an infected program has been executed a preset number of times, it is said to contain a logic bomb. There may be any number of variations, or there may be no activation mechanism other than the initial execution of the infected program.

As mentioned, the objective is usually some unwanted, possibly destructive event. Previous examples of computer viruses have varied widely in their objectives, with some causing irritating but harmless displays to appear, whereas others have erased or modified files or caused system hardware to behave differently. Generally, the objective consists of whatever actions the author has designed into the virus.

As with Trojan horse programs, computer viruses can be introduced into systems deliberately and by unsuspecting users. For example, a Trojan horse program whose purpose is to infect other programs could be planted on a software bulletin board that permits users to upload and download programs. When a user downloads the program and then executes it, the program proceeds to infect other programs in the user's system. If the computer virus hides itself well, the user may continue to spread it by copying the infected program to other disks, by backing it up, and by sharing it with other users. Other examples of how computer viruses are introduced include situations where authorized users of systems deliberately plant viruses, often with a time bomb mechanism. The virus may then activate itself at some later point in time, perhaps when the user is not logged onto the system or perhaps after the user has left the organization.

Network Worms

Network worm programs use network connections to spread from system to system, thus network worms attack systems that are linked via communications hens. Once active within a system, a network worm can behave as a computer virus, or it could implant Trojan horse programs or perform any number of disruptive or destructive actions. In a sense, network worms are like computer viruses with the ability to infect other systems as well as other programs. Some people use the term virus to include both cases.

To replicate themselves, network worms use some sort of network vehicle, depending on the type of network and systems. Examples of network vehicles include (a) a network mail facility, in which a worm can mail a copy of itself to other systems, or (b), a remote execution capability, in which a worm can execute a copy of itself on another system, or (c) a remote

login capability, whereby a worm can log into a remote system as a user and then use commands to copy itself from one system to the other.

The new copy of the network worm is then run on the remote system, where it may continue to spread to more systems in a like manner. Depending on the size of a network, a network worm can spread to many systems in a relatively short amount of time, thus the damage it can cause to one system is multiplied by the number of systems to which it can spread.

A network worm exhibits the same characteristics as a computer virus: a replication mechanism, possibly an activation mechanism, and an objective. The replication mechanism generally performs the following functions:

- searches for other systems to infect by examining host tables or similar repositories of remote system addresses
- establishes a connection with a remote system, possibly by logging in as a user or using a mail facility or remote execution capability
- copies itself to the remote system and causes the copy to be run

The network worm may also attempt to determine whether a system has previously been infected before copying itself to the system. In a multi-tasking computer, it may also disguise its presence by naming itself as a system process or using some other name that may not be noticed by a system operator.
The activation mechanism might use a time bomb or logic bomb or any number of variations to activate itself. Its objective, like all malicious software, is whatever the author has designed into it.
Some network worms have been designed for a useful purpose, such as to perform general "housecleaning" on networked systems, or to use extra machine cycles on each networked system to perform large amounts of computations not practical on one system. A network worm with a harmful objective could perform a wide range of destructive functions, such as deleting files on each affected computer, or by implanting Trojan horse programs or computer viruses.

Other Related Software Threats

The number of variations of Trojan horses, computer viruses, and network worms is apparently endless. Some have names, such as a rabbit, whose objective is to spread wildly within or among other systems and disrupt network traffic, or a bacterium, whose objective is to replicate within a system and eat up processor time until computer throughput is halted It is likely that many new forms wiU be created, employing more sophisticated techniques for spreading and causing damage.

The Threat of Unauthorized Use

In that computer viruses and related forms of malicious software are intriguing issues in themselves, it is important not to overlook that they are created by people, and are fundamentally a people problem. In essence, examples of malicious software are tools that people use to extend and enhance their ability to create mischief and various other forms of damage. Such software can do things that the interactive user often cannot directly effect, such as working with great speed, or maintaining anonymity, or doing things that require

programmatic system calls. But in general, malicious software exploits the same vulnerabilities as can knowledgeable users. Thus, any steps taken to reduce the likelihood of attack by malicious software should address the likelihood of unauthorized use by computer users.

References:

1. Bunzel, Rick; Flu Season: Connect, Summer 1988.
2. Denning, Peter J.; Computer Viruses; American Scientist, Vol 76, May-June, 1988.
3. Denning, Peter J.; The Internet Worm: American Scientist, Vol 77, March- April, 1989.
4. Federal Information Processing Standards Publication 73, Guidelines for Security of Computer Applications; National Bureau of Standards, June, 1980.