

¿HOW VIRUSES WORKS?

A computer virus is a piece of self-replicating code attached to some other piece of code. This code can be harmless—for example, it might display a message or play a tune. Or it might be harmful and proceed to delete and modify files.

The virus code searches users' files for an uninfected executable program for which the user has security write privileges. The virus infects the file by putting a piece of code in the selected program file. When a program that is infected with a virus is executed, the virus immediately takes command, finding and infecting other programs and files.

Some viruses are "memory resident" viruses. When a user executes an executable file that is infected with this type of virus, the virus loads itself into memory and remains there even if the original program is shut down. Subsequent programs that are executed are infected with the virus until the computer is shut down or turned off. Some viruses have a "dormant" phase and will appear only at certain times or when certain actions are performed.

A **variant** is a virus that is generated by modifying a known virus. Examples are modifications that add functionality or evade detection. The term "variant" usually applies only when the modifications are minor. An example would be changing the trigger date from Friday the 13th to Thursday the 12th.

An **overwriting** virus will destroy code or data in the host program by replacing it with the virus code. It should be noted that most viruses attempt to retain the original host program's code and functionality after infection because the virus is more likely to be detected and deleted if the program ceases to work. A **non-overwriting** virus is designed to append the virus code to the physical end of the program or to move the original code to another location.

A **self-recognition** procedure is a technique whereby a virus determines whether or not an executable is already infected. The procedure usually involves searching for a particular value at a known position in the executable. Self-recognition is required if the virus is to avoid multiple infections of a single executable. Multiple infections cause excessive growth in size of infected executables and corresponding excessive storage space, contributing to the detection of the virus.

A **resident** virus installs itself as part of the operating system upon execution of an infected host program. The virus will remain resident until the system is shut down. Once installed in memory, a resident virus is available to infect all suitable hosts that are accessed.

A **stealth** virus is a resident virus that attempts to evade detection by concealing its presence in infected files. To achieve this, the virus intercepts system calls that examine the contents or attributes of infected files. The results of these calls must be altered to correspond to the file's original state. For example, a stealth virus might remove the virus code from an executable when it is read (rather than executed) so that an anti-virus software package will examine the original, uninfected host program.

An **encrypted** virus has two parts: a small decryptor and the encrypted virus body. When the virus is executed, the decryptor will execute first and decrypt the virus body. Then the virus body can execute, replicating or becoming resident. The virus body will include an encryptor to apply during replication. A **variably encrypted** virus will use different encryption keys or encryption algorithms. Encrypted viruses are more difficult to disassemble and study since the researcher must decrypt the code.

A **polymorphic** virus creates copies during replication that are functionally equivalent but have distinctly different byte streams. To achieve this, the virus may randomly insert superfluous instructions, interchange the order of independent instructions, or choose from a number of different encryption schemes. This variable quality makes the virus difficult to locate, identify, or remove.

A **reaserch** virus is one that has been written, but has never been unleashed on the public. These include the samples that have been sent to researchers by virus writers. Viruses that have been seen outside the research community are termed "in the wild."

How Are Computer Viruses Spread?

The following are necessary characteristics of a virus:

- It is able to replicate.
- It requires a host program as a carrier.
- It is activated by external action.
- Its replication ability is limited to the (virtual) system.

Computer viruses move from computer to computer by attaching themselves to files or boot records of disks and diskettes. These days it is not uncommon to find them in e-mail attachments and other programs that can be downloaded from the Internet.

A virus is a relatively passive agent that relies on ordinary users for its activation and propagation. It can travel from one file to another on the same computer if the infected file is executed, from computer memory to a file on disk, on a disk that is carried from one computer to another (some companies prohibit floppy drives, thereby preventing users from copying information onto their computers), on e-mail attachment executable files, and over a modem or network connection.

Damage that Viruses Cause

Viruses can destroy file allocation tables (FAT) and lead to the corruption of an entire file system, resulting in the need to fully reinstall and reload the system. Viruses also can create bad sectors on the disk, destroying parts of programs and files. They can decrease the space on hard disks by duplicating files. They also can format specific tracks on the disks or format the entire disk.

Viruses can destroy specific executable files and alter data in data files, causing a loss of integrity in the data. Viruses can cause the system to hang so that it does not respond to any keyboard or mouse movements.

Trojan Horses

Background

The term "Trojan horse" comes from a myth in which the Greeks gave a giant wooden horse to their foes, the Trojans, seemingly as a peace offering. After the Trojans dragged the horse inside the city walls of Troy, Greek soldiers sneaked out of the horse's hollow belly and opened the city gates, allowing their compatriots to pour in and capture Troy.

What Are Trojan Horses?

A Trojan horse is code hidden in a program such as a game or spreadsheet that looks safe to run but has hidden side effects. When the program is run, it seems to function as the user expects, but in actuality it is destroying, damaging, or altering information in the background. It is a program on its own and does not require a host program in which to embed itself. An example of a Trojan horse would be a Christmas executable that, when executed, pops up with an animated figure of Santa Claus and a caption saying "Merry Christmas." In the background, extra code could be deleting files or performing other malicious actions.

How Trojan Horses Are Spread

Trojan horses generally are spread through e-mail and exchange of disks and information between computers. Worms could also spread Trojan horses.

Damage Caused by Trojan Horses

The damage that Trojan horses cause is much the same as what a virus causes. Most of the time the users are unaware of the damage it is causing because of the Trojan horse's masking effect.

Worms

Background

Worms first were used as a legitimate mechanism for performing tasks in a distributed environment. Network worms were considered promising for the performance of network management tasks in a series of experiments at the Xerox Palo Alto Research Center in 1982. The key problem noted was worm management; controlling the number of copies executing at a single time.

Worms were first noticed as a potential computer security threat when the Christmas Tree Exec attacked IBM mainframes in December 1987. It brought down both the worldwide IBM network and BITNET. The Christmas Tree Exec wasn't a true worm. It was a Trojan horse with a replicating mechanism. A user would receive a Christmas card by e-mail that included executable (REXX) code. If executed, the program claimed to draw a Christmas tree on the display. That much was true, but it also sent a copy to everyone on the user's address lists.

The Internet Worm was a true worm. It was released on November 2, 1988. It attacked Sun and DEC UNIX systems attached to the Internet (it included two sets of binaries, one for each system). It utilized the TCP/IP protocols and vulnerabilities in sendmail, common application layer protocols, operating system bugs, and a variety of system administration flaws to propagate. Various problems with worm management resulted in extremely poor system performance and a denial of network service. It exploited operating system flaws and common system management problems.

What Are Worms?

The following are necessary characteristics of a worm:

- It is able to replicate.
- It is self-contained and does not require a host.
- It is activated by creating process (it needs a multitasking system).
- If it is a network worm, it can replicate across communication links.

A worm 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. They took advantage of system properties to perform useful actions. 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. Worms exploit flaws (that is, bugs) in the operating system or inadequate system management to replicate. Release of a worm usually results in brief outbreaks, shutting down entire networks.

Worms are programs that run independently and travel from computer to computer across network connections. Worms may have portions of themselves running on many different computers. Worms do not change other programs, although they may carry other code that does.

How Worms Affect Network Systems

Developing a worm requires a network environment and an author who is familiar not only with the network services and facilities, but also with the operating facilities required to support them once they've reached the computer. Protection against worm programs is like protection against break-ins. If an intruder can enter your computer, so can a worm program. If the computer is secure from unauthorized access, it should be secure from a worm program.

How Worms Are Spread

Worms are autonomous agents capable of propagating themselves without the use of another program or intervention or action by a user. Worms are found primarily on computers that are capable of multitasking and are connected by a network.

Damage that Worms Can Cause

Most worms disrupt services and create system management problems. Some worms scan for passwords and other loopholes and then send the information back to the attacker. In some cases worms can install Trojan horses or viruses that cause damage to the systems.

Macro Viruses

A macro virus is a virus that attaches itself to a spreadsheet worksheet, or is programmed into the spreadsheet. It also could be programmed into other products such as Word documents and Microsoft PowerPoint® presentations and so on.

Macro viruses are written in high-level languages like Visual Basic® for applications used by Microsoft Office products, Lotus scripting, WordPerfect macros, and so on. Macro viruses bypass integrity protection mechanisms for normal executables because macro viruses are embedded in the data file. Documents are widely exchanged by e-mail and therefore are a good medium for spreading a virus. Users opening a file may not even be aware of the fact that they are running a program. All instructions available for writing macros are also available to virus writers who now can hide viral code in a macro file.

An example of a macro virus is the Melissa macro virus. The Melissa macro virus was spread via e-mail. The virus was programmed into a Word document. When the document was opened, the macro virus would send a copy of it to the first 50 e-mail addresses from the global address list. This caused major e-mail systems to crash throughout the world and also saturated network bandwidth.