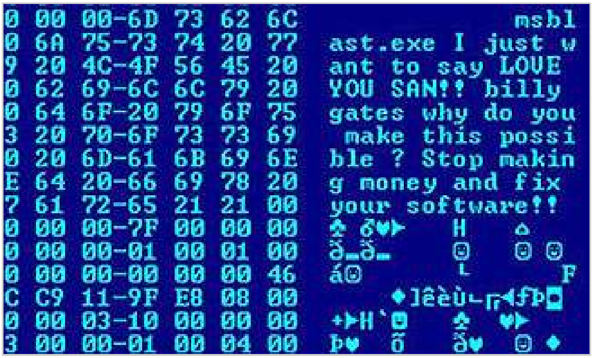
Computer virus

From Wikipedia, the free encyclopedia



```
0 00 00-6D 73 62 6C            msbl
0 6A 75-73 74 20 77   ast.exe I just w
9 20 4C-4F 56 45 20   ant to say LOVE
0 62 69-6C 6C 79 20   YOU SAN!! billy
0 64 6F-20 79 6F 75   gates why do you
3 20 70-6F 73 73 69    make this possi
0 20 6D-61 6B 69 6E   ble ? Stop makin
E 64 20-66 69 78 20   g money and fix
7 61 72-65 21 21 00   your software!!
0 00 00-7F 00 00 00   ♠ ó♥►    H    ⌂
0 00 00-01 00 01 00   ð▁ð▁     ☺    ☺ ☺
0 00 00-00 00 00 46   á☺       └      F
C C9 11-9F E8 08 00       ♦1êèù┴╓◄ƒP■
0 00 03-10 00 00 00   +►H`☻    ♠   ♥►
3 00 00-01 00 04 00   þ♥   õ   ð♥   ☺ ♦
```

Hex dump of the Blaster virus, showing a message left for Microsoft CEO Bill Gates by the virus programmer

A **computer virus** is a type of malware that, when executed, replicates by inserting copies of itself (possibly modified) into other computer programs, data files, or the boot sector of the hard drive; when this replication succeeds, the affected areas are then said to be "infected".[1][2][3] Viruses often perform some type of harmful activity on infected hosts, such as stealing hard disk space or CPU time, accessing private information, corrupting data, displaying political or humorous messages on the user's screen, spamming their contacts, or logging their keystrokes. However, not all viruses carry a destructive payload or attempt to hide themselves—the defining characteristic of viruses is that they are self-replicating computer programs which install themselves without the user's consent.

Virus writers use social engineering and exploit detailed knowledge of security vulnerabilities to gain access to their hosts' computing resources. The vast majority of viruses (over 99%) target systems running Microsoft Windows,[4][5][6] employing a variety of mechanisms to infect new hosts,[7] and often using complex anti-detection/stealth strategies to evade antivirus software.[8][9][10][11] Motives for creating viruses can include seeking profit, desire to send a political message, personal amusement, to demonstrate that a vulnerability exists in software, for sabotage and denial of service, or simply because they wish to explore artificial life and evolutionary algorithms.[12]

Computer viruses currently cause billions of dollars worth of economic damage each year,[13] due to causing systems failure, wasting computer resources, corrupting data, increasing maintenance costs, etc. In response, free, open-source anti-virus tools have been developed, and a multi-billion dollar industry of anti-virus software vendors has cropped up, selling virus protection to Windows users. Unfortunately, no currently existing anti-virus software is able to catch all computer viruses (especially new ones); computer security researchers are actively searching for new ways to enable antivirus solutions to more effectively detect emerging viruses, before they have already become widely distributed.[14]

Contents

Vulnerabilities and infection vectors

Software bugs

Because software is an often designed with security features to prevent unauthorized use of system resources, many viruses must exploit security bugs (security defects) in system or application software to spread. Software development strategies that produce large numbers of bugs will generally also produce potential exploits.

Social engineering and poor security practices

In order to replicate itself, a virus must be permitted to execute code and write to memory. For this reason, many viruses attach themselves to executable files that may be part of legitimate programs (see code injection). If a user attempts to launch an infected program, the virus' code may be executed simultaneously.[15]

In operating systems that use file extensions to determine program associations (such as Microsoft Windows), the extensions may be hidden from the user by default. This makes it possible to create a file that is of a different type than it appears to the user. For example, an executable may be created named "picture.png.exe", in which the user sees only "picture.png" and therefore assumes that this file is an image and most likely is safe, yet when opened runs the executable on the client machine.[16]

Vulnerability of different operating systems to viruses

The vast majority of viruses target systems running Microsoft Windows. This is due both to Microsoft's large market share of desktop users (over 95%), and to design choices in Windows that make it much easier for viruses to infect hosts running Windows. Also, the diversity of software systems on a network limits the destructive potential of viruses and malware.[17] Open-source operating systems such as Linux allow users to choose from a variety of desktop environments, packaging tools, etc. which means that malicious code targeting any one of these systems will only affect a subset of all users. However most Windows users are running the same set of applications, so viruses are able to rapidly spread amongst Windows systems by targeting the same exploits on large numbers of hosts.[4][5][6][18]

Theoretically, other operating systems are also susceptible to viruses, but in practice these are extremely rare or non-existent, due to much more robust security architectures in Unix-like systems (including Linux and Mac OS X) and to the diversity of the applications running on them.[19] There are no known viruses that have spread "in the wild" for Mac OS X.[20][21] The difference in virus vulnerability between Macs and Windows is a chief selling point, one that Apple uses in their Get a Mac advertising.[22]

While Linux (and Unix in general) has always natively prevented normal users from making changes to the operating system environment without permission, Windows users are generally not prevented from making these changes, meaning that viruses can easily gain control of the entire system on Windows hosts. This difference has continued partly due to the widespread use of administrator accounts in contemporary versions like XP. In 1997, researchers created and released a virus for Linux—known as "Bliss".[23] Bliss, however, requires that the user run it explicitly, and it can only infect programs that the user has the access to modify. Unlike Windows users, most Unix users do not log in as an administrator user except to install or configure software; as a result, even if a user ran the virus, it could not harm their operating system. The Bliss virus never became widespread, and remains chiefly a research curiosity. Its creator later posted the source code to Usenet, allowing researchers to see how it worked.[24]

Infection targets and replication techniques

Computer viruses infect a variety of different subsystems on their hosts.[25] One manner of classifying viruses is to analyze whether they reside in binary executables (such as .EXE or .COM files), data files (such as Microsoft Word documents or PDF files), or in the boot sector of the host's hard drive (or some combination of all of these).[26][27]

Resident vs. non-resident viruses

A *memory-resident virus* (or simply "resident virus") installs itself as part of the operating system when executed, after which it remains in RAM from the time the computer is booted up to when it is shut down. Resident viruses overwrite interrupt handling code or other functions, and when the operating system attempts to access the target file or disk sector, the virus code intercepts the request and redirects the control flow to the replication module, infecting the target. In contrast, a *non-memory-resident virus* (or "non-resident virus"), when executed, scans the disk for targets, infects them, and then exits (i.e. it does not remain in memory after it is done executing).[28][29][30]

Macro viruses

Many common applications, such as Microsoft Outlook and Microsoft Word, allow macro programs to be embedded in documents or emails, so that the programs may be run automatically when the document is opened. A *macro virus* (or "document virus") is a virus that is written in a macro language, and embedded into these documents so that when users open the file, the virus code is executed, and can infect the user's computer. This is one of the reasons that it is dangerous to open unexpected attachments in e-mails.[31][32]

Boot sector viruses

Boot sector viruses specifically target the boot sector/Master Boot Record (MBR) of the host's hard drive or removable storage media (flash drives, floppy disks, etc.).[26][33][34]

Stealth strategies

In order to avoid detection by users, some viruses employ different kinds of deception. Some old viruses, especially on the MS-DOS platform, make sure that the "last modified" date of a host file stays the same when the file is infected by the virus. This approach does not fool antivirus software, however, especially those which maintain and date cyclic redundancy checks on file changes.[*citation needed*]

Some viruses can infect files without increasing their sizes or damaging the files. They accomplish this by overwriting unused areas of executable files. These are called *cavity viruses*. For example, the CIH virus, or Chernobyl Virus, infects Portable Executable files. Because those files have many empty gaps, the virus, which was 1 KB in length, did not add to the size of the file.[35]

Some viruses try to avoid detection by killing the tasks associated with antivirus software before it can detect them.

[*citation needed*]

As computers and operating systems grow larger and more complex, old hiding techniques need to be updated or replaced. Defending a computer against viruses may demand that a file system migrate towards detailed and explicit permission for every kind of file access.[*citation needed*]

Read request intercepts

While some antivirus software employ various techniques to counter stealth mechanisms, once the infection occurs any recourse to clean the system is unreliable. In Microsoft Windows operating systems, the NTFS file system is proprietary. Direct access to files without using the Windows OS is undocumented. This leaves antivirus software little alternative but to send a read request to Windows OS files that handle such requests. Some viruses trick antivirus software by intercepting its requests to the OS. A virus can hide itself by intercepting the request to read the infected file, handling the request itself, and return an uninfected version of the file to the antivirus software. The interception can occur by code injection of the actual operating system files that would handle the read request. Thus, an antivirus software attempting to detect the virus will either not be given permission to read the infected file, or, the read request will be served with the uninfected version of the same file.[36]

The only reliable method to avoid stealth is to boot from a medium that is known to be clean. Security software can then be used to check the dormant operating system files. Most security software relies on virus signatures, or they employ heuristics.[*citation needed*]

Security software may also use a database of file hashes for Windows OS files, so the security software can identify altered files, and request Windows installation media to replace them with authentic versions. In older versions of Windows, file hashes of Windows OS files stored in Windows—to allow file integrity/authenticity to be checked—could be overwritten so that the System File Checker would report that altered system files are authentic, so using file hashes to scan for altered files would not always guarantee finding an infection.[*citation needed*]

Self-modification

See also: Self-modifying code

Most modern antivirus programs try to find virus-patterns inside ordinary programs by scanning them for so-called *virus signatures*. Unfortunately, the term is misleading, in that viruses do not possess unique signatures in the way that human beings do. Such a virus signature is merely a sequence of bytes that an antivirus program looks for because it is known to be part of the virus. A better term would be "search strings". Different antivirus programs will employ different search strings, and indeed different search methods, when identifying viruses. If a virus scanner finds such a pattern in a file, it will perform other checks to make sure that it has found the virus, and not merely a coincidental sequence in an innocent file, before it notifies the user that the file is infected. The user can then delete, or (in some cases) "clean" or "heal" the infected file. Some viruses employ techniques that make detection by means of signatures difficult but probably not impossible. These viruses modify their code on each infection. That is, each infected file contains a different variant of the virus.

Encrypted viruses

One method of evading signature detection is to use simple encryption to encipher the body of the virus, leaving only the encryption module and a cryptographic key in cleartext.[37] In this case, the virus consists of a small decrypting module and an encrypted copy of the virus code. If the virus is encrypted with a different key for each infected file, the only part of the virus that remains constant is the decrypting module, which would (for example) be appended to the end. In this case, a virus scanner cannot directly detect the virus using signatures, but it can still detect the decrypting module, which still makes indirect detection of the virus possible. Since these would be symmetric keys, stored on the infected host, it is in fact entirely possible to decrypt the final virus, but this is probably not required, since self-modifying code is such a rarity that it may be reason for virus scanners to at least flag the file as suspicious.[*citation needed*]

An old, but compact, encryption involves XORing each byte in a virus with a constant, so that the exclusive-or operation had only to be repeated for decryption. It is suspicious for a code to modify itself, so the code to do the encryption/decryption may be part of the signature in many virus definitions.[*citation needed*]

Some viruses will employ a means of encryption inside an executable in which the virus is encrypted under certain events, such as the virus scanner being disabled for updates or the computer being rebooted. This is called Cryptovirology. At said times, the executable will decrypt the virus and execute its hidden runtimes infecting the computer and sometimes disabling the antivirus software.

Polymorphic code

Polymorphic code was the first technique that posed a serious threat to virus scanners. Just like regular encrypted viruses, a polymorphic virus infects files with an encrypted copy of itself, which is decoded by a decryption module. In the case of polymorphic viruses, however, this decryption module is also modified on each infection. A well-written polymorphic virus therefore has no parts which remain identical between infections, making it very difficult to detect directly using signatures.[38][39] Antivirus software can detect it by decrypting the viruses using an emulator, or by statistical pattern analysis of the encrypted virus body. To enable polymorphic code, the virus has to have a polymorphic engine (also called mutating engine or mutation engine) somewhere in its encrypted body. See polymorphic code for technical detail on how such engines operate.[40]

Some viruses employ polymorphic code in a way that constrains the mutation rate of the virus significantly. For example, a virus can be programmed to mutate only slightly over time, or it can be programmed to refrain from mutating when it infects a file on a computer that already contains copies of the virus. The advantage of using such slow polymorphic code is that it makes it more difficult for antivirus professionals to obtain representative samples of the virus, because bait files that are infected in one run will typically contain identical or similar samples of the virus. This will make it more likely that the detection by the virus scanner will be unreliable, and that some instances of the virus may be able to avoid detection.

Metamorphic code

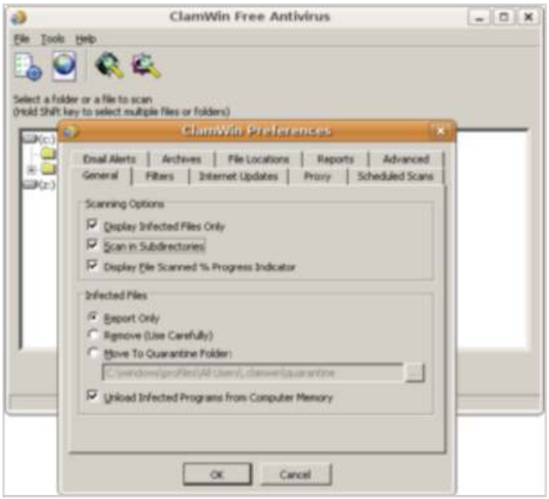
To avoid being detected by emulation, some viruses rewrite themselves completely each time they are to infect new executables. Viruses that utilize this technique are said to be metamorphic. To enable metamorphism, a metamorphic engine is needed. A metamorphic virus is usually very large and complex. For example, W32/Simile consisted of over 14,000 lines of assembly language code, 90% of which is part of the metamorphic engine.[41][42]

Countermeasures

See also: Vulnerability to malware and Anti-malware strategies

Antivirus software

Many users install antivirus software that can detect and eliminate known viruses when the computer attempts to download or run the executable (which may be distributed as an email attachment, or on USB flash drives, for example). Some antivirus software blocks known malicious web sites that attempt to install malware. Antivirus software does not change the underlying capability of hosts to transmit viruses. Users must update their software regularly to patch security vulnerabilities ("holes"). Antivirus software also needs to be regularly updated in order to recognize the latest threats. The German AV-TEST Institute publishes evaluations of antivirus software for Windows[43] and Android.[44]



Select a folder or a file to scan
(Hold Shift key to select multiple files or folders)

Scanning Options

☑ Display Infected Files Only
☑ Scan in Subdirectories
☑ Display File Scanned % Progress Indicator

Infected Files

● Report Only
○ Remove (Use Carefully)
○ Move To Quarantine Folder:

☑ Unload Infected Programs from Computer Memory

Screenshot of the open source ClamWin antivirus software running in Wine on Ubuntu Linux

Examples of Microsoft Windows anti virus and anti-malware software include the optional Microsoft Security Essentials[45] (for Windows XP, Vista and Windows 7) for real-time protection, the Windows Malicious Software Removal Tool[46] (now included with Windows (Security) Updates on "Patch Tuesday", the second Tuesday of each month), and Windows Defender (an optional download in the case of Windows XP).[47] Additionally, several capable antivirus software programs are available for free download from the Internet (usually restricted to non-commercial use).[48] Some such free programs are almost as good as commercial competitors.[49] Common security vulnerabilities are assigned CVE IDs and listed in the US National Vulnerability Database. Secunia PSI[50] is an example of software, free for personal use, that will check a PC for vulnerable out-of-date software, and attempt to update it. Ransomware and phishing scam alerts appear as press releases on the Internet Crime Complaint Center noticeboard.

Other commonly used preventative measures include timely operating system updates, software updates, careful Internet browsing, and installation of only trusted software.[51]

There are two common methods that an antivirus software application uses to detect viruses, as described in the antivirus software article. The first, and by far the most common method of virus detection is using a list of virus signature definitions. This works by examining the content of the computer's memory (its RAM, and boot sectors) and the files stored on fixed or removable drives (hard drives, floppy drives, or USB flash drives), and comparing those files against a database of known virus "signatures". Virus signatures are just strings of code that are used to identify individual viruses; for each virus, the anti-virus designer tries to choose a unique signature string that will not be found in a legitimate program. Different anti-virus programs use different "signatures" to identify viruses. The disadvantage of this detection method is that users are only protected from viruses that are detected by signatures in their most recent virus definition update, and not protected from new viruses (see "zero-day attack").[52]

A second method to find viruses is to use a heuristic algorithm based on common virus behaviors. This method has the ability to detect new viruses for which anti-virus security firms have yet to define a "signature", but it also gives rise to more false positives than using signatures. False positives can be disruptive, especially in a commercial environment.

Recovery strategies and methods

One can also reduce the damage done by viruses by making regular backups of data (and the operating systems) on different media, that are either kept unconnected to the system (most of the time), read-only or not accessible for other reasons, such as using different file systems. This way, if data is lost through a virus, one can start again using the backup (which will hopefully be recent).[*citation needed*]

If a backup session on optical media like CD and DVD is closed, it becomes read-only and can no longer be affected by a virus (so long as a virus or infected file was not copied onto the CD/DVD). Likewise, an operating system on a bootable CD can be used to start the computer if the installed operating systems become unusable. Backups on removable media must be carefully inspected before restoration. The Gammima virus, for example, propagates via removable flash drives.[53][54]

Virus removal

Many websites run by antivirus software companies provide free online virus scanning, with limited cleaning facilities (the purpose of the sites is to sell anti-virus products). Some websites—like Google subsidiary VirusTotal.com—allow users to upload one or more suspicious files to be scanned and checked by one or more antivirus programs in one operation.[55][56] Additionally, several capable antivirus software programs are available for free download from the Internet (usually restricted to non-commercial use).[57] Microsoft offers an optional free antivirus utility called Microsoft Security Essentials, a Windows Malicious Software Removal Tool that is updated as part of the regular Windows update regime, and an older optional anti-malware (malware removal) tool Windows Defender that has been upgraded to an antivirus product in Windows 8.

Some viruses disable System Restore and other important Windows tools such as Task Manager and Command Prompt. An example of a virus that does this is CiaDoor. Many such viruses can be removed by rebooting the computer, entering Windows safe mode with networking, and then using system tools or Microsoft Safety Scanner.[58] System Restore on Windows Me, Windows XP, Windows Vista and Windows 7 can restore the registry and critical system files to a previous checkpoint. Often a virus will cause a system to hang, and a subsequent hard reboot will render a system restore point from the same day corrupt. Restore points from previous days should work provided the virus is not designed to corrupt the restore files and does not exist in previous restore points.[59]

Operating system reinstallation

Microsoft's System File Checker (improved in Windows 7 and later) can be used to check for, and repair, corrupted system files.[*citation needed*]

Restoring an earlier "clean" (virus-free) copy of the entire partition from a cloned disk, a disk image, or a backup copy is one solution—restoring an earlier backup disk image is relatively simple to do, usually removes any malware, and may be faster than disinfecting the computer—or reinstalling and reconfiguring the operating system and programs from scratch, as described below, then restoring user preferences.[*citation needed*]

Reinstalling the operating system is another approach to virus removal. It may be possible to recover copies of essential user data by booting from a live CD, or connecting the hard drive to another computer and booting from the second computer's operating system, taking great care not to infect that computer by executing any infected programs on the original drive. The original hard drive can then be reformatted and the OS and all programs installed from original media. Once the system has been restored, precautions must be taken to avoid reinfection from any restored executable files.[*citation needed*]

Historical development

See also: Timeline of notable computer viruses and worms

Early academic work on self-replicating programs

The first academic work on the theory of self-replicating computer programs[60] was done in 1949 by John von Neumann who gave lectures at the University of Illinois about the "Theory and Organization of Complicated Automata". The work of von Neumann was later published as the "Theory of self-reproducing automata". In his essay von Neumann described how a computer program could be designed to reproduce itself.[61] Von Neumann's design for a self-reproducing computer program is considered the world's first computer virus, and he is considered to be the theoretical father of computer virology.[62]

In 1972 Veith Risak, directly building on von Neumann's work on self-replication, published his article "Selbstreproduzierende Automaten mit minimaler Informationsübertragung" (Self-reproducing automata with minimal information exchange).[63] The article describes a fully functional virus written in assembler language for a SIEMENS 4004/35 computer system.

In 1980 Jürgen Kraus wrote his diplom thesis "Selbstreproduktion bei Programmen" (Self-reproduction of programs) at the University of Dortmund.[64] In his work Kraus postulated that computer programs can behave in a way similar to biological viruses.

The first computer viruses

The Creeper virus was first detected on ARPANET, the forerunner of the Internet, in the early 1970s.[65] Creeper was an experimental self-replicating program written by Bob Thomas at BBN Technologies in 1971.[66] Creeper used the ARPANET to infect DEC PDP-10 computers running the TENEX operating system.[67] Creeper gained access via the ARPANET and copied itself to the remote system where the message, "I'm the creeper, catch me if you can!" was displayed. The *Reaper* program was created to delete Creeper.[68]

This is the MacMag virus 'Universal Peace', as displayed on a Mac in March of 1988.

In 1982, a program called "Elk Cloner" was the first personal computer virus to appear "in the wild"—that is, outside the single computer or lab where it was created.[69] Written in 1981 by Richard Skrenta, it attached itself to the Apple DOS 3.3 operating system and spread via floppy disk.[69][70] This virus, created as a practical joke when Skrenta was still in high school, was injected in a game on a floppy disk. On its 50th use the Elk Cloner virus would be activated, infecting the personal computer and displaying a short poem beginning "Elk Cloner: The program with a personality."

In 1984 Fred Cohen from the University of Southern California wrote his paper "Computer Viruses – Theory and Experiments".[71] It was the first paper to explicitly call a self-reproducing program a "virus", a term introduced by Cohen's mentor Leonard Adleman. In 1987, Fred Cohen published a demonstration that there is no algorithm that can perfectly detect all possible viruses.[72] Fred Cohen's theoretical compression virus[73] was an example of a virus which was not malware, but was putatively benevolent. However, antivirus professionals do not accept the concept of benevolent viruses, as any desired function can be implemented without involving a virus (automatic compression, for instance, is available under the Windows operating system at the choice of the user). Any virus will by definition make unauthorised changes to a computer, which is undesirable even if no damage is done or intended. On page one of *Dr Solomon's Virus Encyclopaedia*, the undesirability of viruses, even those that do nothing but reproduce, is thoroughly explained.[2]

An article that describes "useful virus functionalities" was published by J. B. Gunn under the title "Use of virus functions to provide a virtual APL interpreter under user control" in 1984.[74]

The first IBM PC virus in the wild was a boot sector virus dubbed (c)Brain,[75] created in 1986 by the Farooq Alvi Brothers in Lahore, Pakistan, reportedly to deter piracy of the software they had written.[76]

The first virus to specifically target Microsoft Windows, WinVir was discovered in April 1992, two years after the release of Windows 3.0. The virus did not contain any Windows API calls, instead relying on DOS interrupts. A few years later, in February 1996, Australian hackers from the virus-writing crew Boza created the VLAD virus, which was the first known virus to target Windows 95. In late 1997 the encrypted, memory-resident stealth virus Win32.Cabanas was released—the first known virus that targeted Windows NT (it was also able to infect Windows 3.0 and Windows 9x hosts).[77]

Even home computers were affected by viruses. The first one to appear on the Commodore Amiga was a boot sector virus called SCA virus, which was detected in November 1987.[78]

Viruses and the Internet

See also: Computer worm

Before computer networks became widespread, most viruses spread on removable media, particularly floppy disks. In the early days of the personal computer, many users regularly exchanged information and programs on floppies. Some viruses spread by infecting programs stored on these disks, while others installed themselves into the disk boot sector, ensuring that they would be run when the user booted the computer from the disk, usually inadvertently. Personal computers of the era would attempt to boot first from a floppy if one had been left in the drive. Until floppy disks fell out of use, this was the most successful infection strategy and boot sector viruses were the most common in the wild for many years.

Traditional computer viruses emerged in the 1980s, driven by the spread of personal computers and the resultant increase in BBS, modem use, and software sharing. Bulletin board–driven software sharing contributed directly to the spread of Trojan horse programs, and viruses were written to infect popularly traded software. Shareware and bootleg software were equally common vectors for viruses on BBSs.[*citation needed*] Viruses can increase their chances of spreading to other computers by infecting files on a network file system or a file system that is accessed by other computers.[79]

Macro viruses have become common since the mid-1990s. Most of these viruses are written in the scripting languages for Microsoft programs such as Word and Excel and spread throughout Microsoft Office by infecting documents and spreadsheets. Since Word and Excel were also available for Mac OS, most could also spread to Macintosh computers. Although most of these viruses did not have the ability to send infected email messages, those viruses which did take advantage of the Microsoft Outlook COM interface.[*citation needed*]

Some old versions of Microsoft Word allow macros to replicate themselves with additional blank lines. If two macro viruses simultaneously infect a document, the combination of the two, if also self-replicating, can appear as a "mating" of the two and would likely be detected as a virus unique from the "parents".[80]

A virus may also send a web address link as an instant message to all the contacts on an infected machine. If the recipient, thinking the link is from a friend (a trusted source) follows the link to the website, the virus hosted at the site may be able to infect this new computer and continue propagating.[*citation needed*]

Viruses that spread using cross-site scripting were first reported in 2002,[81] and were academically demonstrated in 2005.[82] There have been multiple instances of the cross-site scripting viruses in the wild, exploiting websites such as MySpace and Yahoo!.

See also

- Botnet
- Computer insecurity
- Crimeware

- Cryptovirology
- Multipartite virus
- Spam (electronic)
- Virus hoax
- Windows 7 File Recovery
- Windows Action Center (Security Center)

References

1. ^ Aycock, John (2006). *Computer Viruses and Malware*. Springer. p. 14. ISBN 978-0-387-30236-2.
2. ^ ***a b*** Dr. Solomon's Virus Encyclopedia, 1995, ISBN 1-897661-00-2, Abstract at http://vx.netlux.org/lib/aas10.html (archived version (https://web.archive.org/web/20110614105852/http://vx.netlux.org/lib/aas10.html))
3. ^ The term "virus" is also commonly, but erroneously, used to refer to other types of malware. "Malware" encompasses computer viruses along with many other forms of malicious software, such as computer worms, ransomware, trojan horses, keyloggers, rootkits, spyware, adware, malicious BHOs and other malicious software. The majority of active malware threats are actually trojans or worms rather than viruses.
4. ^ ***a b*** Mookhey, K.K. et al (2005). *Linux: Security, Audit and Control Features* (http://books.google.com/books?id=-kD0sxQ0EkIC&pg=PA128). ISACA. p. 128. ISBN 9781893209787.
5. ^ ***a b*** Toxen, Bob (2003). *Real World Linux Security: Intrusion Prevention, Detection, and Recovery* (http://books.google.com/books?id=_-1jwRwNaEoC&pg=PA365). Prentice Hall Professional. p. 365. ISBN 9780130464569.
6. ^ ***a b*** Noyes, Katherine (Aug 3, 2010). "Why Linux Is More Secure Than Windows" (https://www.pcworld.com/article/202452/why_linux_is_more_secure_than_windows.html). *PCWorld*.
7. ^ Skoudis, Edward (2004). "Infection mechanisms and targets" (http://books.google.com/books?id=TKEAQmQV7O4C&pg=PA31). *Malware: Fighting Malicious Code*. Prentice Hall Professional. pp. 31–48. ISBN 9780131014053.
8. ^ Aycock, John (2006). *Computer Viruses and Malware*. Springer. p. 27. ISBN 978-0-387-30236-2.
9. ^ Ludwig, Mark A. (1996). *The Little Black Book of Computer Viruses: Volume 1, The Basic Technologies*. pp. 16–17. ISBN 0-929408-02-0.
10. ^ Harley, David et al (2001). *Viruses Revealed*. McGraw-Hill. p. 6. ISBN 0-07-222818-0.
11. ^ Filiol, Eric (2005). *Computer viruses:from theory to applications*. Springer. p. 8. ISBN 978-2-287-23939-7.
12. ^ Bell, David J. et al, ed. (2004). "Virus" (http://books.google.com/books?id=5MFWZK0CSOQC&pg=PA154). *Cyberculture: The Key Concepts*. Routledge. p. 154. ISBN 9780203647059.
13. ^ "Viruses that can cost you" (http://www.symantec.com/region/reg_eu/resources/virus_cost.html).
14. ^ Kaspersky, Eugene (November 21, 2005). "The contemporary antivirus industry and its problems" (https://www.securelist.com/en/analysis/174405517/The_contemporary_antivirus_industry_and_its_problems). SecureLight.
15. ^ "Virus Basics" (http://www.us-cert.gov/publications/virus-basics#email). US-CERT.
16. ^ "Virus Notice: Network Associates' AVERT Discovers First Virus That Can Infect JPEG Files, Assigns Low-Profiled Risk" (http://www.woodboy.org/computing/first_virus_that_can_infect_jpegs.html). Retrieved 2002-06-13.
17. ^ This is analogous to how genetic diversity in a population decreases the chance of a single disease wiping out a population
18. ^ Raggi, Emilio et al (2011). *Beginning Ubuntu Linux* (http://books.google.com/books?id=5i-c2yms6tUC&pg=PA148). Apress. p. 148. ISBN 9781430236276.
19. ^ Worstall, Tim. "Is Unix Now The Most Successful Operating System Of All Time?" (http://www.forbes.com/sites/timworstall/2013/05/07/is-unix-now-the-most-successful-operating-system-of-all-time/). forbes.com. Retrieved 5/07/2013.
20. ^ Elmer-Dewitt, Phillip (September 2, 2009). "Why are there no Mac viruses" (http://tech.fortune.cnn.com/2009/09/02/why-are-there-no-mac-viruses/). *CNN / Fortune Tech*.
21. ^ There are, however, a variety of Trojans and other malware that exist for OS X, along with a variety of security vulnerabilities that can be exploited by attackers to gain unauthorized access to Mac systems. See for example: Sutter, John D. (22 April 2009). "Experts: Malicious program targets Macs" (http://www.cnn.com/2009/TECH/04/22first.mac.botnet/index.html). CNN.com. Retrieved 24 April 2009.; "Trojan virus tricks Apple Mac users to steal passwords" (http://www.telegraph.co.uk/technology/apple/9104229/Trojan-virus-tricks-Apple-Mac-users-to-steal-passwords.html). *The Daily Telegraph* (London). 2012-02-26. and "Malware Evolution: Mac OS X Vulnerabilities 2005–2006" (http://www.viruslist.com/en/analysis?pubid=191968025). Kaspersky Lab. 2006-07-24. Retrieved August 19, 2006.
22. ^ "Get a Mac" (http://www.apple.com/getamac). Apple. Retrieved 2012-07-15.

23. ^ "McAfee discovers first Linux virus" (http://math-www.uni-paderborn.de/~axel/bliss/mcafee_press.html) (Press release). McAfee, via Axel Boldt. 5 February 1997.
24. ^ Boldt, Axel (19 January 2000). "Bliss, a Linux 'virus'" (http://math-www.uni-paderborn.de/~axel/bliss/).
25. ^ Serazzi, Giuseppe & Zanero, Stefano (2004). "Computer Virus Propagation Models" (http://home.deib.polimi.it/zanero/papers/zanero-serazzi-virus.pdf). In Calzarossa, Maria Carla & Gelenbe, Erol. *Performance Tools and Applications to Networked Systems*. Lecture Notes in Computer Science. Vol. 2965. pp. 26–50.
26. ^ ***a b*** Avoine, Gildas et al. (2007). *Computer System Security: Basic Concepts and Solved Exercises* (http://books.google.com/books?id=UwrOhcgVhsMC&pg=PA21). EPFL Press / CRC Press. pp. 21–22. ISBN 9781420046205.
27. ^ Brain, Marshall; Fenton, Wesley. "How Computer Viruses Work" (http://www.howstuffworks.com/virus.htm). HowStuffWorks.com. Retrieved 16 June 2013.
28. ^ Grimes, Roger (2001). *Malicious Mobile Code: Virus Protection for Windows* (http://books.google.com/books?id=GKDtVYJ0wesC&pg=PA37). O'Reilly. pp. 37–38. ISBN 9781565926820.
29. ^ Salomon, David (2006). *Foundations of Computer Security* (http://books.google.com/books?id=d2RNQNUWPIkC&pg=PA47). Springer. pp. 47–48. ISBN 9781846283413.
30. ^ Polk, William T. (1995). *Anti-virus Tools and Techniques for Computer Systems* (http://books.google.com/books?id=laTf7a_jPy8C&pg=PA4). William Andrew (Elsevier). p. 4. ISBN 9780815513643.
31. ^ Grimes, Roger (2001). "Macro Viruses" (http://books.google.com/books?id=GKDtVYJ0wesC&pg=PA130). *Malicious Mobile Code: Virus Protection for Windows*. O'Reilly. ISBN 9781565926820.
32. ^ Aycock, John (2006). *Computer Viruses and Malware* (http://books.google.com/books?id=xnW-qvk1gzkC&pg=PA89). Springer. p. 89. ISBN 9780387341880.
33. ^ Anonymous (2003). *Maximum Security* (http://books.google.com/books?id=3jqBnS4b3EgC&pg=PA331). Sams Publishing. pp. 331–333. ISBN 9780672324598.
34. ^ Skoudis, Edward (2004). "Infection mechanisms and targets" (http://books.google.com/books?id=TKEAQmQV7O4C&pg=PA37). *Malware: Fighting Malicious Code*. Prentice Hall Professional. pp. 37–38. ISBN 9780131014053.
35. ^ "Computer Virus Strategies and Detection Methods" (http://www.emis.de/journals/IJOPCM/files/IJOPCM(vol.1.2.3.S.8).pdf). Retrieved 2 September 2008.
36. ^ Szor, Peter (2005). *The Art of Computer Virus Research and Defense* (http://books.google.com/books?id=XE-ddYF6uhYC&pg=PT285). Boston: Addison-Wesley. p. 285. ISBN 0-321-30454-3.
37. ^ Bishop, Matt (2003). *Computer Security: Art and Science* (http://books.google.com/books?id=pfdBiJNfWdMC&pg=PA620). Addison-Wesley Professional. p. 620. ISBN 9780201440997.
38. ^ Kizza, Joseph M. (2009). *Guide to Computer Network Security* (http://books.google.com/books?id=GyDM9kvo3MIC&pg=PA341). Springer. p. 341. ISBN 9781848009165.
39. ^ Eilam, Eldad (2011). *Reversing: Secrets of Reverse Engineering* (http://books.google.com/books?id=_78HnPPRU_oC&pg=PT216). John Wiley & Sons. p. 216. ISBN 9781118079768.
40. ^ "Virus Bulletin : Glossary – Polymorphic virus" (http://www.virusbtn.com/resources/glossary/polymorphic_virus.xml). Virusbtn.com. 2009-10-01. Retrieved 2010-08-27.
41. ^ Perriot, Fredrick; Peter Ferrie and Peter Szor (May 2002). "Striking Similarities" (http://securityresponse.symantec.com/avcenter/reference/simile.pdf) (PDF). Retrieved September 9, 2007.
42. ^ "Virus Bulletin : Glossary — Metamorphic virus" (http://www.virusbtn.com/resources/glossary/metamorphic_virus.xml). Virusbtn.com. Retrieved 2010-08-27.
43. ^ "Detailed test reports—(Windows) home user" (http://www.av-test.org/en/tests/home-user/). AV-Test.org.
44. ^ "Detailed test reports—Android mobile devices)" (http://www.av-test.org/en/tests/mobile-devices/android/). AV-Test.org.
45. ^ "Microsoft Security Essentials" (http://windows.microsoft.com/en-US/windows/products/security-essentials). Retrieved June 21, 2012.
46. ^ "Malicious Software Removal Tool" (http://www.microsoft.com/security/pc-security/malware-removal.aspx). Retrieved June 21, 2012.
47. ^ "Windows Defender" (http://www.microsoft.com/en-us/download/details.aspx?id=17). Retrieved June 21, 2012.
48. ^ Rubenking, Neil J. (Feb 17, 2012). "The Best Free Antivirus for 2012" (http://www.pcmag.com/article2/0,2817,2388652,00.asp). pcmag.com.
49. ^ Rubenking, Neil J. (Jan 10, 2013). "The Best Antivirus for 2013" (http://www.pcmag.com/article2/0,2817,2372364,00.asp). pcmag.com.
50. ^ Rubenking, Neil J. "Secunia Personal Software Inspector 3.0 Review & Rating" (http://www.pcmag.com/article2/0,2817,2406767,00.asp). PCMag.com. Retrieved 2013-01-19.
51. ^ "10 Step Guide to Protect Against Viruses" (http://www.bapcs.co.uk/10-step-guide-to-protect-against-viruses). Bits & PCs. Retrieved 16 June 2013.

52. ^ Zhang, Yu et al (2008). "A Novel Immune Based Approach For Detection of Windows PE Virus" (http://books.google.com/books?id=gakfOYC3RmIC&pg=PA250). In Tang, Changjie et al. *Advanced Data Mining and Applications: 4th International Conference, ADMA 2008, Chengdu, China, October 8-10, 2008, Proceedings*. Springer. p. 250. ISBN 9783540881919.
53. ^ "Symantec Security Summary — W32.Gammima.AG." http://www.symantec.com/security_response/writeup.jsp?docid=2007-082706-1742-99
54. ^ "Yahoo Tech: Viruses! In! Space!" http://tech.yahoo.com/blogs/null/103826
55. ^ "VirusTotal.com (a subsidiary of Google)" (https://www.virustotal.com/).
56. ^ "VirScan.org" (http://www.virscan.org/).
57. ^ Rubenking, Neil J. "The Best Free Antivirus for 2014" (http://www.pcmag.com/article2/0,2817,2388652,00.asp). pcmag.com.
58. ^ "Microsoft Safety Scanner" (http://windows.microsoft.com/en-US/windows7/how-do-I-remove-a-computer-virus).
59. ^ "Symantec Security Summary — W32.Gammima.AG and removal details." http://www.symantec.com/security_response/writeup.jsp?docid=2007-082706-1742-99&tabid=3
60. ^ The term "computer virus" was not used at that time.
61. ^ von Neumann, John (1966). "Theory of Self-Reproducing Automata" (http://cba.mit.edu/events/03.11.ASE/docs/VonNeumann.pdf). *Essays on Cellular Automata* (University of Illinois Press): 66–87. Retrieved June 10., 2010.
62. ^ Éric Filiol, *Computer viruses: from theory to applications, Volume 1* (http://books.google.com/books?id=CZGLFf6IhCIC&pg=PA19), Birkhäuser, 2005, pp. 19–38 ISBN 2-287-23939-1.
63. ^ Risak, Veith (1972), "Selbstreproduzierende Automaten mit minimaler Informationsübertragung" (http://www.cosy.sbg.ac.at/~risak/bilder/selbstrep.html), *Zeitschrift für Maschinenbau und Elektrotechnik*
64. ^ Kraus, Jürgen (February 1980), *Selbstreproduktion bei Programmen* (http://vx.netlux.org/lib/pdf/Selbstreproduktion%20bei%20programmen.pdf)
65. ^ "Virus list" (http://www.viruslist.com/en/viruses/encyclopedia?chapter=153310937). Retrieved 2008-02-07.
66. ^ Thomas Chen, Jean-Marc Robert (2004). "The Evolution of Viruses and Worms" (http://vx.netlux.org/lib/atc01.html). Retrieved 2009-02-16.
67. ^ Parikka, Jussi (2007). *Digital Contagions: A Media Archaeology of Computer Viruses* (http://books.google.com/books?id=yhe0w_j1iiQC&pg=PA50). New York: Peter Lang. p. 50. ISBN 978-0-8204-8837-0.
68. ^ Russell, Deborah & Gangemi, G.T. (1991). *Computer Security Basics* (http://books.google.co.uk/books?id=BtB1aBmLuLEC&pg=PA86). O'Reilly. p. 86. ISBN 0-937175-71-4.
69. ^ ***a b*** Anick Jesdanun (1 September 2007). "School prank starts 25 years of security woes" (http://www.nbcnews.com/id/20534084/#.UWgIGcrNOCY). CNBC. Retrieved April 12, 2013.
70. ^ "The anniversary of a nuisance" (http://www.cnn.com/2007/TECH/09/03/computer.virus.ap/).
71. ^ Cohen, Fred (1984), *Computer Viruses – Theory and Experiments* (http://all.net/books/virus/index.html)
72. ^ Cohen, Fred, An Undetectable Computer Virus (http://www.research.ibm.com/antivirus/SciPapers/VB2000DC.htm), 1987, IBM
73. ^ Burger, Ralph, 1991. *Computer Viruses and Data Protection*, pp. 19–20
74. ^ Gunn, J.B. (June 1984). "Use of virus functions to provide a virtual APL interpreter under user control" (http://portal.acm.org/ft_gateway.cfm?id=801093&type=pdf&coll=GUIDE&dl=GUIDE&CFID=93800866&CFTOKEN=49244432). *ACM SIGAPL APL Quote Quad archive* (ACM New York, NY, USA) **14** (4): 163–168. ISSN 0163-6006 (//www.worldcat.org/issn/0163-6006).
75. ^ "Boot sector virus repair" (http://antivirus.about.com/od/securitytips/a/bootsectorvirus.htm). Antivirus.about.com. 2010-06-10. Retrieved 2010-08-27.
76. ^ "Amjad Farooq Alvi Inventor of first PC Virus post by Zagham" (https://www.youtube.com/watch?v=m58MqJdWgDc). YouTube. Retrieved 2010-08-27.
77. ^ Grimes, Roger (2001). *Malicious Mobile Code: Virus Protection for Windows* (http://books.google.com/books?id=GKDtVYJ0wesC&pg=PA99). O'Reilly. pp. 99–100. ISBN 9781565926820.
78. ^ "SCA virus" (http://agn-www.informatik.uni-hamburg.de/catalog/amiga/html/scaorigi.htm). Virus Test Center, University of Hamburg. 1990-06-05. Retrieved 2014-01-14.
79. ^ "What is a Computer Virus?" (http://www.actlab.utexas.edu/~aviva/compsec/virus/whatis.html). Actlab.utexas.edu. 1996-03-31. Retrieved 2010-08-27.
80. ^ Vesselin Bontchev. "Macro Virus Identification Problems" (http://www.people.frisk-software.com/~bontchev/papers/macidpro.html). *FRISK Software International*.
81. ^ Berend-Jan Wever. "XSS bug in hotmail login page" (http://seclists.org/bugtraq/2002/Oct/119).
82. ^ Wade Alcorn. "The Cross-site Scripting Virus" (http://www.bindshell.net/papers/xssv/).

Further reading

- Burger, Ralf (16 February 2010) [1991]. *Computer Viruses and Data Protection*. Abacus. p. 353. ISBN 978-1-55755-123-8.
- Granneman, Scott (6 October 2003). "Linux vs. Windows Viruses" (http://www.theregister.co.uk/2003/10/06/linux_vs_windows_viruses/). *The Register*.
- Ludwig, Mark (1993). *Computer Viruses, Artificial Life and Evolution* (http://vx.netlux.org/lib/vml02.html). Tucson, Arizona 85717: American Eagle Publications, Inc. ISBN 0-929408-07-1.
- Mark Russinovich (November 2006). *Advanced Malware Cleaning* ***video*** (http://technet.microsoft.com/en-us/sysinternals/gg618529) (Web (WMV / MP4)). Microsoft Corporation. Retrieved 24 July 2011.
- Parikka, Jussi (2007). *Digital Contagions. A Media Archaeology of Computer Viruses*. Digital Formations. New York: Peter Lang. ISBN 978-0-8204-8837-0.

External links

- Viruses (http://www.dmoz.org/Computers/Security/Malicious_Software/Viruses/) on the Open Directory Project (DMOZ)
- Microsoft Security Portal (http://www.microsoft.com/security/)
- US Govt CERT (Computer Emergency Readiness Team) site (http://www.us-cert.gov/)
- 'Computer Viruses – Theory and Experiments' (http://all.net/books/virus/index.html) – The original paper by Fred Cohen, 1984
- Hacking Away at the Counterculture (http://www3.iath.virginia.edu/pmc/text-only/issue.990/ross-1.990) by Andrew Ross (On hacking, 1990)
- VX Heaven - the biggest library computer viruses (http://vxheaven.org/)

Retrieved from "http://en.wikipedia.org/w/index.php?title=Computer_virus&oldid=597780105"
Categories: Internet security | Computer viruses | Computer security exploits

- This page was last modified on 2 March 2014 at 08:03.
-