# Those ubiquitous viruses

Raphael Finkel
Computer Science Department
University of Kentucky
Lexington, KY

raphael@cs.uky.edu
raphael@ukma.bitnet

## Malicious logic

Programmers performing disruptive acts under the cloak of anonymity.

- **Worm**: A program that invades a computer via a network and then looks for other target computers. Each instance is called a *segment*.
    - **Host worm**: lives entirely on one computer, uses a network just for propagation.
    - **Network worm**: lives on many computers; the segments keep in touch.
    - **Chain letter**: A program that requires human interaction in order to spread, but is otherwise a worm.

  We will discuss the 1988 arpanet worm and the 1987 Christmas chain letter.

## Malicious logic, continued

- **Trojan horse**: a program that intentionally performs some undocumented, usually malicious, function. A Trojan horse can access files of victim that the designer could not get at.

  For example, a game might ask for your password (legitimate) and store it somewhere available to the author (illegitimate).

  Strictly speaking, a chain letter is usually a Trojan horse, and a virus makes infected programs Trojan horses.

    - **Wormhole** or **backdoor**: A program that allows the author to defeat its security provisions. We will discuss Thomson's login wormhole.

## Malicious logic, continued

- **Virus**: A program that can infect other programs by modifying them in such a way that they begin to include a version of the virus code and are themselves viruses.

  The computer virus has similarities to biologic viruses. It is small and splices copies of itself into other programs. When these programs run, viral code directs the computer to make additional viral copies and splice them into other programs.

  The original program then continues after a barely noticeable delay. Host facilities are subverted into producing copies of the foreign intruder. These hidden programs may also produce delays, noises, scrambling, or deletion of data.

  For example, an infected diskette transmits virus to main store, thence to all diskettes subsequently loaded.

## Arpanet Worm, 2 November 88

**What**: a program that exploits flaws in utility programs that allows it to break into those machines (that is, start unauthorized processes there) and copy itself, thereby infecting those machines.

**At risk**: All Vax and Sun computers on the Arpa Internet (perhaps 60,000 machines).

**Actual toll**: Many of the at-risk machines turned out to be immune (they were running slightly different versions of the utility programs). The best estimate is 6,000 machines affected.

**Quarantine**: Many machines disconnected from the network within 12 hours of the onset, protecting themselves temporarily but hampering the dissemination of bug fixes.

## Arpanet worm, continued

**Flaws attacked**: Network servers listen for connections from the network and respond to properly formed requests. They are essential to network use. Both flaws make faulty network servers start up command interpreters.

The *finger* server is used to see if a party is logged in on a machine. It failed to check if input from the network was too long. The worm called the finger server on a target machine and gave it a carefully prepared overlong message. The message overwrote part of the finger server and caused it to start up a command interpreter.

The *mail* server accepts incoming electronic mail. It accepted a debug command from the network, and then allowed the network to specify a program instead of a user name as the recipient of a message. The worm (at the other end of the network connection) specified the command interpreter as the program to run.

**Propagation**: After it starts a command interpreter, the worm calls for compilation of a simple communication program, which is then executed. The communication program brings the whole worm (in executable form) from its current infection site. The new worm is then started.

## Arpanet worm, continued

**Activities**: A new worm hides itself (erases its argument vector, removes disk files pertaining to it, periodically becomes a new process). It then looks for suitable victim machines and tries to infect them. Finally, it tries to crack passwords on its current site with a dictionary attack.

**Redundancy control**: Each worm periodically tries to connect to other local worms; if it succeeds, it terminates. A bug in this code made redundancy control practically worthless. The worm was easily detected by the enormous resources its many simultaneous instances consumed.

**Postscript**: Robert T. Morris, Jr. was convicted in February 1990, fined $10,000, and required to perform 400 hours of community service. An appeal 3/7/91 upheld the conviction; the U.S. Supreme court refused to hear a further appeal.

## The login wormhole

Ken Thomson was a codeveloper of Unix. He wanted to be able to log into any Unix anywhere.

He modified the source for *login* to accept his special password. He compiled *login* and then restored the original source.

He modified the source for the *compiler* to insert his modification any time it compiled *login*. He compiled the *compiler* and then restored the original source.

He modified the source for the *compiler* to insert his modification any time it compiled the *compiler*. He compiled the *compiler* and then restored the original source.

This wormhole persisted until a completely new compiler was installed.

## The Christmas chain letter (12/87)

| |
|---|
| From: A good friend of yours<br>Enclosed in this computer mail is a program.<br>Let it run and enjoy yourself!<br>A very happy Christmas and my best<br>wishes for the next year.<br>(Browsing the program is no fun; just run it.) |

Running the program has two effects:

- It displays a picture of a tree and holiday greetings.
- It uses the victim's mailing list to send copies of itself everywhere, signed by the victim.

**At risk**: IBM machines running VM/CMS connected via BITNET, EARN, Netnorth, and IBM's internal network VNET.

**Effect**: exponential load on processors and networks.

**Postscipt**: The same idea keeps showing up (CHRISTMA 12/5/88, BUL 3/8/89 in Turkey, DIR 11/25/89 at TECMTYVM, ORGASM 4/1/89, HEADACH 4/8/89). BITNET core sites now filter all variants out.

## Trojan horses

Jan 23, 1989, Issue #9 of The Dirty Dozen by Tom Sirianni and Sally Neuman listed 63 different programs on bulletin boards that are Trojan horses.

Examples:

CDIR.COM
: This program is supposed to give you a color directory of files on your disk, but it in fact will scramble your disk's file allocation table.

DROID.EXE
: This Trojan appears under the guise of a game. You are supposedly an architect that controls futuristic droids in search of relics. In fact, copies files to unexpected locations.

EGABTR
: Description says something like "improve your EGA display," but when run, it deletes everything in sight and prints, "Arf! Arf! Got you!"

## Typical Trojan horse warnings

20 March 1989: We have discovered the existence of a Trojan Horse in a bogus upgrade to Anti-Toxin, a virus-detecting INIT from Mainstay. The INIT, labelled as version 2.0 in the Get Info box, attempts to format your disk and rename it "Scored!".

12 December 1989: The “AIDS information diskette” from a corporation calling itself PC Cyborg was widely distributed to major corporations and PC user groups around the world. The diskette contained a highly destructive Trojan. After 90 reboots, the software would encrypt all data on the hard disk. People had to write decryption software to restore the data. Hundreds of systems were affected, including at the Chase Manhattan Bank and ICL Computers.

Postscript: 2 December 1991: Joseph L. Popp Jr., 39, was arrested in Cleveland and charged with blackmail, extradited to England, and charged with mailing 20,000 such disks from London about 11 December, 1989. Prosecutors there decided to drop the case in November, 1991 until Popp is deemed fit to stand trial.

## Viruses: why they are dangerous

- They can propagate with great speed; doubling time is about 2 months.
- They can spread via PC networks, floppies (especially), and bulletin boards (rarely).

The following initial reports of the WDEF Macintosh virus occurred within a few weeks of each other January 1990.

University of Michigan, University of North Carolina at Chapel Hill, Arizona State University, University of Oregon, University of Southern California, University of South Carolina, University of Rochester, Connecticut College, University of Miami, Illinois State University, Emory College, Texas A&M, Texas Christian University, Humber College in Toronto, Carnegie-Mellon University, Smith College (Massachusetts)

- They can be hard to detect because they avoid superinfection and can sit in object files and even in bad blocks.
- They are especially robust on PC’s, because there is no main-store or disk protection. An infected machine remains infected; viruses are persistent on floppies.

## Typical virus description — Ogre

Infects the boot record of any writable diskette or hard disk.

Classification: Boot record virus.

If the virus triggers then recognition is easy. Another method of recognizing it is the 8k of memory lost. You can detect infected diskettes by running Chkdsk. If you get 3K of bad sectors on a 360k diskette, that's a sign of Ogre [but could be Brain or something else].

When you boot from an infected diskette, the virus goes memory resident. While it is in memory, any disk that you access is liable to be infected. Ogre will replace the boot record with its own code, move the boot record further up the disk, add the rest of the Ogre code, and mark these sectors as bad in the file allocation table.

If you leave your computer on for 48 hours, and access the hard disk during the following hour, the virus triggers. It clears the screen, and puts up "Disk Killer -- Version 1.00 by COMPUTER OGRE 04/01/1989" in black characters on a white background. Then in yellow on green, it says "Warning !!", and two lines down "Don't turn off the power or remove the diskette while Disk Killer is Processing!". Then in bright red, and blinking, on black, it says "PROCESSING". By then it is too late. Your best course will be to re-initialize the disk and restore the latest backups.

Ogre was first sighted in the US, but we have also had a case in Ealing near London. Ogre is more infectious than Italian, as it can infect 80286 and 80386 machines, which Italian cannot. [Since this note was written, variants of Italian have been created that can.]

## What does a virus do?

- **Activation**: The execution of viral code. When the viral code is finished, it restores the program and invokes it.
- **Infection**: The virus placing a copy of itself in a target. Not all activations lead to infection. Some activations lead to many infections.
- **Logic/time bomb**: Code that checks for a certain condition/time and activates the payload. The conditions may have a random component. Typically, very few activations release the payload.
- **Payload**: The damage caused when the virus chooses to do damage. Some payloads are obvious and immediate; more dangerous are those that cause minor corruption that slowly builds up. Common payloads are displaying music or graphics, corrupting the boot record or files, formatting or overwriting the disk, affecting runtime operation of programs, and corrupting directory structures.

## What can a virus infect?

- **Boot record** (**BSI**, or **boot sector infector**): activated when computer is booted. (These tend to be the most successful viral programs.)
    - **Master boot record** (**MBR**; contains the hard-disk partition table) Most MBR viruses don't harm the partition table. Most save the true MBR elsewhere on the disk, sometimes corrupting other information or placing the true MBR at risk for later corruption.
    - **DOS boot record** (on both fixed and floppy disks)
- **File**: activated when an infected executable file (typically .COM, executed.
  Sometimes viruses infect overlays by mistake. File infectors often change the size of file, but some overwrite the file [Viper], or live in unused regions of .COM files [Proud] or in headers of .EXE files [Rat]. Even data files (to programs with interpreter capability, like spreadsheets) can be infected [MacMag].
- **File system**: changes information in directory entries so that each file seems to begin with the virus [Dir II].

## Other kinds of virus

- **Multipartite**: Able to infect both files and boot records [Tequila].
- **Companion**: A virus that attacks an executable file (say FOO.EXE) by installing itself as a new file with a name that takes execution priority (such as FOO.COM) or earlier in the execution path (such as FOO.COM). In that latter case, it is also called a **path** virus.
- **Dropper**: A program that installs a virus in the boot sector or in an executable file. Droppers are not themselves viruses; they do not spread.
- **Germ**: The result of direct assembly of the virus, not the result of an infection. Also called Generation 1.

## What techniques do viruses use?

- **Memory resident**: A virus that when activated remains in memory, able to continue to infect or to damage. Some leave memory after one infection [Anthrax].
- **Interrupt tracing**: A virus capable of finding the original interrupt handler for any interrupt. [First done in Yankee Doodle]
- **Stealth**: A memory-resident virus that redirects reads from the MBR to make it look uninfected [EXEbug] or that disinfects all files as they are read and re-infects files as they are written [512, 4096]. Some cannot even be detected in files that are read at sector (as opposed to file) level [Int13]. Stealth viruses were invented to circumvent integrity checkers.
- **Semi-stealth**: A virus that hides the fact that file size has increased [651], or that removes itself when inspected by a debugger [Yankee Doodle].

## More viral techniques

- **Fast infector**: A memory-resident virus that infects files as they are opened (or even as others are deleted). Fast infectors were invented to attack scanners by making them unwitting agents of viral spread.
- **Slow infector**: A memory-resident virus that infects executable files only when they are created or modified. Slow infectors circumvent activity monitors and integrity checking.
- **Sparse infector**: A virus that does not infect each time, to avoid detection or because it is too difficult for the virus to tell reliably if it has already infected the target. Sparse infectors try to reduce the likelihood of detection.
- **Armored**: Virus that uses tricks to make tracing, disassembling, and understanding their code more difficult [Whale].
- **Directed attack**: Virus that attempts to circumvent weaknesses of particular antiviral tools [Antimon].

## More viral techniques

- **Mutating/polymorphic**: Virus that can exist in many forms, with an encryption device or with extraneous instructions. Some have decriptors that can be scanned by wild-card strings. Others [V2P6] require special-purpose detection engines. Some [MtE family] are so polymorphic that it is quite difficult to build successful detection engines. Polymorphic viruses were invented to circumvent scanners.
- **Tunnelling**: Virus that can disable an activity monitor program it detects in memory or invoke monitored functions in a way that cannot be intercepted (such as by CALLs to the ROM BIOS). Tunnelling viruses were invented to circumvent activity monitors.
- **Spawning**: A virus that after it finishes its activation, creates and executes a fresh copy of the original program. Spawning viruses attempt to circumvent self-checkers.

## Virus-production software

A disturbing trend: Software has appeared that makes it easier to create viruses.

- The Mutation Engine (MtE 0.90-beta) is a library routine that any virus writer can link into the virus that causes each infection to be encrypted in a randomly selected way.
  Many scanners can detect MtE viruses, but often not completely successfully.
- The TPE was first reported around 3 January 1993. It has the same goals as MtE.
- The Virus Creation Laboratory (VCL 1.00) makes virus writing a matter of picking options from pull-down menus.
  Infection type, encryption, and so forth, can be selected, and a large range of nasty effects and conditions can also be selected. The package can make *.COM infecting viruses, companion viruses, trojans, and time-bomb object files. Luckily, VCL is full of bugs and practically useless.
- The Phalcom/Skism Mass-Produced Code Generator (PS-MPC) was first reported 6 November 1992.
  It generates quite primitive viruses, but they actually do work. It generates assembler code,

which can be readily modified to create variants.

## How do antivirals work?

- **Scanner**: A program that examines memory, files, and boot blocks for known viral scan strings. Can detect a virus before it spreads. Cannot detect previously unknown viruses. Some only look at likely locations in files and miss certain viruses. Recommended: AVTK, F-PROT.
- **Heuristic analyzer**: A program that examines programs for suspicious code by looking for generic activities common to actual viruses. Can often detect previously unknown viruses.
- Terminate-and-stay-resident program (**TSR**): A program that remains in memory so it can prevent the execution of programs infected with malicious code. Might include scanning and integrity checking of programs before they are run.
    - **Activity monitor** or **operation restrictor**: A TSR program that catches and prevents activities such as attempts to format the disk. Recommended: SECURE.

## More antiviral methods

- **Recognizer/Identifier**: A program that discovers a viral presence, or a program that tells exactly which strain of virus is present. Identification is necessary for disinfection.
- **Disinfecter**: A program that removes a virus from the memory, file, or boot block where it is found. Cannot be done for all viruses. Often it is better to replace the damaged file with a clean original. Recommended: F-PROT.
- **Encryption package**: A suite of programs that creates a non-standard environment to prevent successful viral spread.
- **Integrity** or **change checker**: A program that creates checksums on files and later verifies the integrity of the checksum. Can detect previously unknown viruses. Cannot detect companion viruses. Cannot detect a virus until it begins to propagate, and then cannot identify the virus (or even verify that a virus has caused the change). Cannot always detect droppers. Recommended: Untouchable, ASP Integrity Toolkit, Integrity Master.
    - **Integrity shell**: A TSR integrity checker that checks the integrity of programs being executed.

## More antiviral methods

- **Immunizer**: A program that modifies an executable file to allow that executable or some other program to recognize any change to the executable (and perhaps to reverse those changes). Discouraged, since it does not work with stealth viruses, some programs cannot be immunized without damaging them, and immunizing an infected file can hide the infection from scanners.
- **Self checker**: A program that checks various things before running, such as its own integrity (in memory and on disk), the integrity of its data files, and the potential presence in memory of TSR viruses. Most antiviral programs are self-checkers. There are software packages [Stealth Bomber 2.2] to make any program a self-checker. Self-checkers can often not be disinfected. If the self-check fails, the antiviral should quit, lest it inadvertantly spread the infection.
- **Decoy launcher**: A program that creates files on the disk and inspects them regularly to see if they have been infected.

## Other tools you might use

- **Disk utility**: A suite of programs useful for finding and fixing problems on the disk. Recommended: Explorer, PEEKA.

- Other utilities to modify the master boot record to check whether it has been infected, to disable boot from a floppy with Alt-Ctrl-Del, to check memory for suspicious anomalies. Recommended: FIXUTILS.

- **Archiver**: A program for backing up disks. Usable: ARC, ZOO, ZIP, ARJ

## Antiviral programs (partial list)

Amiga: BootX 4.50, Berserker 5.02, NoVirus 3.31, VirusChecker 6.06, VirusX, ZeroVirus III 1.15

Atari: VKILLER 3.84, Chasseur, Satrotan, VirusDie, VirusKil

Macintosh: AntiPan 1.5, AntiVirus 2.0 (MacTools), Disinfectant 2.9, GateKeeper 1.2.6, Interferon 3.1, Rival 1.1.9v, SAM 3.0.8SD, Symantec Anti-Virus 3.0, Vaccine, Virex 3.x, VirusDetective 5.0.4, VirusRX

MS-DOS: Antivirus [many have this name], ASP Integrity Toolkit, Central Point AntiVirus (CPAV) 1.3, Certus LAN 2.0, Control Room, Data Physician 3.1A, DISKSECURE 1.15A, Eliminator 1.17, F-PROT 2.07, HTScan 1.18, Gobbler-II 3.0, Integrity Master 1.31d, Mace Vaccine 3.0, McAfee 212e, Norton AntiVirus (NAV) 2.1 PC-Cillin 2.95L, PC-RX 1.1, Solomon AntiVirus Toolkit (AVTK) 5.56, Thunderbyte 6.25, Untouchable 1.13, Virus-Safe 1.12, Thurderbyte Scan 3.3, V-Analyst III 23.00.12, VACCINE 5.00, VET 7.06, Virus detection system (VDS) 2.10, Victor Charlie 5.0, Vi-Spy 9.0, Virex-PC (VIRx) 2.6, ViruCide 2.37, Virus Buster 3.93, VirusCURE 2.37 VIRSCAN (IBM) 2.2.3a, VIRUSCAN Suite 102, VirusSafe LAN 4.01
Available on networks at several sites, such as:

ftp.informatik.uni-hamburg.de (134.100.4.42)
phil.utmb.edu (129.109.9.22)
garbo.uwasa.fi (128.214.87.1)
risc.ua.edu (130.160.4.7)
urvax.urich.edu (141.166.36.6)

## Safe hex

- Never leave a diskette in drive A after file transfer has been done.
- Companies that use computers must have a written policy of what to do in case of a virus ourbreak.
- Provide an emergency box with clean write-protected bootable disks (with minimal software: CONFIG.SYS, COMMAND.COM, CHKDSK, FDISK; without windows or other fancy desk managers) and write-protected disks with anti-virus software.
- Use several antiviral techniques, including scanning (especially of new software before use), activity monitoring, and integrity checking.
  Don't trust any software you receive from outside, whether it be from a manufacturer, a bulletin board, or a friend.
- Backup often and systematically.
  But be careful. There are cases of 50MB of good disk overwritten by a blank disk and of boxes of IBM cards accidentally used for confetti. Date and label the backups.

## Safe network hex

- Set the access rights for applications drives (those with sharable executables) to *read only* for all users including administrators. Don't use software that cannot be installed in this manner.
- Don't provide drives where multiple users have write access. If you must, don't let *path* statements refer to such a drive and don't allow executable or batch files on them.
- Scan all applications before and after installation.
- Have all administrators maintain two accounts with separate passwords. Only one has privileges and is used only when necessary.
- Scan applications files daily and personal files occasionally.
- Scan administrator stations daily and other stations weekly. If an administrator moves to a new station, scan it before logging in.

## What do I do if I am infected?

- Don’t panic.
- Turn off the machine, then boot from a write-protected original system disk. (To ensure it was a clean boot, check your CMOS setup. If it indicates no floppy drive A:, reset it and reboot from floppy.)
- Make a copy of the virus onto a floppy for researchers to investigate.
- Remove, disinfect, or replace the minimal amount needed to remove the infection. If the virus has modified the MBR (but not the partition table), try FDISK /MBR (MS-DOS 5.0). It is almost never necessary to perform a low-level format.

## I have a new virus.

Send a sample to people in the following list:

Bontchev, Vesselin (researcher)
bontchev@fbihh.informatik.uni-hamburg.de
Chess, David (IBM)
chess@watson.ibm.com
Ducklin, Paul (developer)
duck@nuustak.csir.co.za
Goretsky, Aryeh (McAfee Associates)
mcafee@netcom.com
Greenberg, Ross (author of Virex-PC, Flushot)
72461.3212@CompuServe.COM
Jordan, Glenn (Virex-PC Development Team)
trent@rock.concert.net
Kuo, Jimmy (Norton AntiVirus Research)
cjkuo@ccmail.norton.com
Naggs, Anthony (developer)
xa329@city.ac.uk
Skulason, Fridrik (author of F-PROT)
frisk@complex.is
Yetiser, Tarkan (VDS Advanced Research Group)
tyetiser@ssw02.ab.umd.edu

Include details: (1) to whom you have sent samples, (2) how you sent it, (3) how you got the virus, (4) whether you have trouble clearing up the infection, (5) anything you have discovered about how the virus operates, (6) what kind of computer and operating system you are using.

## The good guys

Computer Emergency Response Team
cert@cert.sei.cmu.edu; (412)268-7090
Sample advisory, 29 Jan 1990: CERT has learned of and verified break-ins on several Internet systems in which the intruders have exploited a vulnerability in the Sun sendmail program. ...

Computer Virus Industry Association. 4423 Cheeney Street, Santa Clara, CA 95054-0253, (408) 727-4559

Virus Task Force of the Software Development Council

Virus Test Center, University of Hamburg
(bontchev@fbihh.informatik.uni-hamburg.de)

Vancouver Institute for Research into User Security. rslade@cue.bc.ca

ICSA Virus Research Center. Washington, DC; (202) 364-8252.

There is a yearly conference in New York in March (since 1988) on computer viruses and security and one in Europe in December (since 1989) on anti-virus research.

## The bad guys

Viruses have been created all over the world by dozens of anonymous (and a few known) people.

There is a Virus eXchange BBS (bulletin board) in Sofia, Bulgaria.

There is a Dutch Crackers group and a Chaos computer club.

> Anyone who uploads a new virus may then download all the viruses currently on file there. Many of these viruses are quite dangerous and are present in source code form. More recently, everyone has been allowed to download anything.

Similar BBSes exist in USA, Germany, Italy, Sweden, Czechoslovakia, UK, and Russia.

## How many viruses are out there?

- IBM PC (OS/2): none (1992)
- IBM PC (MS-DOS and similar): over 400 (in 1992). Many have several strains. There are about 1400 strains. About 100 strains/month are reported.
  Some of the common families: Italian (Bouncing Ball, Ping Pong, Vera Cruz), Austrian (Vienna, Unesco), New Zealand (Stoned, Marijuana), Cascade (second austrian, blackjack, falling tears), Oropax (Music), Den Zuk (Search, Venezuelan), Friday 13th (Jerusalem, Israeli, Hebrew University, sUMsDos; sURIV 3.01 variant), Ogre (Disk Killer), Fu Manchu, Dark Avenger, Vacsina, Fumble, Typo, 4096.

- Macintosh: 30 (in 1992); some have several strains.
  Some of the common families: MacMag (Peace, Drew), nVIR (nVIR A, nVIR B, Hpat and AIDS), Scores (Vult), INIT 29, Anti, Dukakis, WDEF, CDEF, MBDF

- Amiga: 150 (in 1992)
- Atari ST: 11 (in 1990)
- Apple II: 4 (in 1990)
- C-64: 1 (in 1992)
- Acorn Archimedes: 42 (in 1993); some have several strains.

## How are viruses named?

Viruses are given names based on messages they print, numbers of bytes they add to files, geographic location where they are first seen, or internal text.

There is a standard nomenclature for viruses, defined by CARO (Computer Anti-virus Researchers' Organization).

The full name of a virus consists of up to four parts, delimited by dots. The general format is

    family.group.variant.version

Examples:

- Stoned.Beijing.D
- Anti-Pascal_II.440.B
- Best_Wishes.1024.B8

## Why people write viruses

The virus writer likes causing damage and thinks it's funny. It gives a feeling of power.

— or —

The schoolkid who wrote the Stoned virus did it on a dare. Having written it, the consequences of unleashing it became a bit much to think about, so he made sure all copies were destroyed bar one which he kept at his house. Despite being under lock and key, his little brother and a couple of his friends thought it would be a huge joke to steal the disk and deliberately infect disks in a local computer store. This was fine, but after the initial laughs it proved impossible to trace all infected disks and the Stoned epidemic was born.

— or —

Curiosity, anger, desire for glory/peer approval.

## How people develop new strains

- Insignificant modifications to non-referenced area.
- Changes to text messages.
- Code patches (typically to fool scanners).
- Reassembly with a different assembler.
- Source code modifications, with no significant functionality changes (disinfectors must recognize).
- New features/changed functionality (such as encryption, bug corrections).
- Recompiling with a different compiler.
- Translation.
- Hybrids and code borrowing.

## References

Peter Denning, *Computer Viruses*, **American Scientist 76** 3, May-June 1988.

Peter Denning, ed., *Computers Under Attack*, Addison Wesley 1990. ISBN: 0-201-53067-8.

Eugene H. Spafford, *The internet worm program: an analysis*, Purdue Technical Report CSD-TR-823, November 29, 1988.

J.C. Van Winkel, "The phenomenon of computer viruses reviewed", 1989, NGI, Amsterdam. ISBN: 90-70621-29-0.

Lance J. Hoffman, ed. *Rogue Programs: Viruses, Worms, and Trojan Horses*, Van Nostrand Reinhold, 1990, ISBN 0-442-00454-0.

David Ferbrache, *A Pathology of Computer Viruses*, Springer-Verlag, 1992.

## More references

Eugene Spafford, Kathleen Heaphy, and David Ferbrache, *Computer Viruses: Dealing with Electronic Vandalism and Programmed Threats*, 1989, 109 pages. Published by ADAPSO (703-522-5055).

The National Research Council, *Computers at Risk — Safe Computing in the Information Age*, National Academy Press in Washington (1-800-624-6242), book code COMRIS, $19.95.

Vesselin Bontchev, *The Bulgarian and Soviet Virus Factories*.

Computer bulletin boards:

- net.comp.risks
- net.comp.virus

Many of the detailed accounts and much of the advice in this talk are taken, often verbatim, from articles on these bulletin boards.