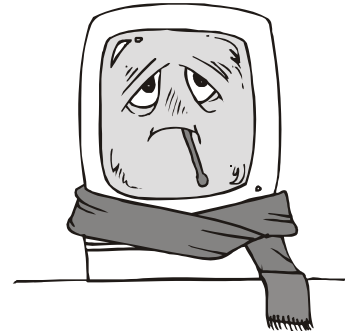


# I-LOVE-YOU: Viruses, Trojan Horses and Worms

## *Electronic threats, protection and potential disasters*

In this lecture various forms of malicious software will be examined and discussed in terms of a general threat which is likely to cause disasters (like the discussion of the Y2K bug). However, in difference to that bug we want to focus on intended threats meaning programs that were created to do something the user doesn't intend or control. From the popular press it might be concluded that viruses are the biggest security threat today. In fact, unintentional human errors (see Y2K) cause much more damage! However, a bug or error doesn't qualify as intended, malicious code! On the other hand, even though some viruses do not intentionally damage your data, in this context all viruses to be malicious software because they modify your programs without your permission with occasional disastrous results



## 1. History

The principle ideas of code, that is self-replicating goes back to about 1970. Today computer viruses are increasing at an unprecedented rate. In 1986, there was one known computer virus; three years later, that number had increased to six and by 1990, the total had jumped to 80. By November of that year, viruses were being discovered at the rate of one per week. Today, between 10 and 15 new viruses appear every day. In fact, from December 1998 to October 1999, the total virus count jumped from 20,500 to 42,000.

## 2. Defining the Terminology of Malicious Code

Viruses are one specific type of program written deliberately to cause harm to someone's computer or to use that computer in an unauthorised way. There are many forms of malicious software; sometimes the media calls all malicious software viruses, but it's important to understand the distinction between the various types. Before we are going to discuss the problems of malicious code, we are going to have a glance at the subcategories which are used today to specify what the code does.

**Viruses** are computer programs that are designed to spread themselves from one file to another on a single computer and eventually do some additional harm. The term virus arises because the infected program acts like a biological **host** being infected by a virus. The virus spreads from the infected file to other still healthy files in such a way that the virus code is executed when the infected program is executed. It is **self-replicating!**

A virus infects a file by attaching itself to the program and either destroys the program or coexists with it. Viruses are categorised again in terms where and how they infect (see Boot-Sector, Polymorphic, Stealth, Macro).

A virus might rapidly infect every application file on an individual computer, or slowly infect the documents on that computer, but it does not intentionally try to spread itself from that computer to other computers. This is done by humans sharing (infected) software by exchange of emails, floppies etc.

A **Logic Bomb** is a class of malicious code that goes off when a special condition occurs. Just like a real bomb, a logic bomb will lie dormant until triggered by some event. The trigger can be the number of times executed, a random number, or even a specific event such as deletion of an employee's payroll record. When the logic bomb is triggered it will usually do something unpleasant. A **Time Bomb** is a logic bomb triggered by a date or time.

As the name implies, a **Trojan Horse** program comes with a hidden surprise intended by the programmer but totally unexpected by the user. They are named after the Trojan horse which delivered by the Greek soldiers into the city of Troy. Likewise, a Trojan program is a delivery vehicle for some destructive code (such as a logic bomb or a virus) onto a computer. The Trojan program appears to be a useful program, but when a certain event occurs, it will attack the system in some way. Unlike viruses, Trojan Horses don't make copies of themselves, but they may contain a virus to be released in an event. An example would be a program that simulates the login-message of a system, grabs the username and password, remotely logs the user in (so s/he doesn't suspect a thing) but passes the username and password to a file or another person for malicious use.

A **Trapdoor** or **Backdoor** is a feature in a program (not necessary a virus) by which someone can access the program other than by the obvious -direct call or execution- typically with special privileges. Many programmers build such features into their programs either for maintenance purposes or to 'ensure their



employment'. An example would be a special PIN that allows access to everyone's transactions on an automated bank teller.

**Worms** are like viruses in that they do replicate themselves. However, instead of spreading from file to file, they spread from computer to computer via a network of some sort, infecting an entire system. For that purpose they rely on some transport mechanism or vehicle (shared disks, email,...). It may arrive in the form of a joke program or software of some sort.

Worms are insidious because they rely less (or not at all) upon human behaviour in order to spread themselves from one computer to others. The computer worm is a program that is designed to copy itself from one computer to another, leveraging some

network medium: e-mail, TCP/IP, etc. The worm is more interested in infecting as many machines as possible on the network, and less interested in spreading many copies of itself on a single computer (like a computer virus). The prototypical worm infects (or causes its code to run on) a target system only once; after the initial infection, the worm attempts to spread to other machines on the network. The so called *Morris ARPANET/INTERNET* "virus" was actually a worm. It created copies of itself through the *ARPA* network, eventually bringing the network to its knees. It did not infect other programs as a virus would, but simply kept creating copies of itself which would then execute and try to spread to other machines.

A **Rabbit** is defined as a virus or worm that replicates without bound with the intention to exhaust some computer resource (disk space, memory).

A **Joke** is a harmless program that causes various activities to display on your computer without causing any damage (e.g., an unexpected screen-saver).

A **virus Hoax** is an e-mail that is intended to scare people about a non-existent virus threat. Users often forward these alerts thinking they are doing a service to their fellow workers, but this causes lost productivity, panic and lost time. This increased traffic can soon become a massive problem in e-mail systems and cause unnecessary fear and panic.

Before we are going to envision possible disasters, we want to have a closer look on how viruses etc. operate.

### 3. Viruses, Trojan horses and Worms

#### 3.1 Viruses

Viruses are either **benign or malignant**. The majority of viruses are harmless and do no real damage to a computer or files. A benign virus might do nothing more than display a message at a pre-determined time or slow down the performance of a computer.

**Malignant viruses** cause damage to a computer system, such as corrupting files or destroying data. (These viruses don't corrupt the files they infect; that would prevent them from spreading. They infect, and then wait for a trigger date to do damage.) Just because a virus is classified as malignant does not mean that the damage it causes is intentional. Sometimes the damage is the result of poor programming or unintended bugs in the viral code.

A virus that has been found in more than one organisation or company is called an in the **wild virus**. Currently, **approximately 250 viruses exist in the wild**. Whether a virus is new or old, it can still be in the wild. A **zoo virus** can be found only within research labs and has not succeeded in moving into general circulation. The current census reports approximately **42,000+ zoo viruses**.



#### General Virus Behaviour

Viruses come in a great many different forms, but they all potentially have two phases to their execution, the **infection phase** and the **attack phase**:

1. When the virus executes it will **infect** other programs. What is often not clearly understood is precisely when it will infect the other programs. Some viruses infect other programs each time they are executed, other viruses infect only upon a certain trigger. This trigger could be anything; it could be a day or time, an external event on your PC, a counter within the virus etc. Some viruses are very selective about when they infect programs; this is vital to the virus's survival. If the virus infects too often, it is more likely to be

discovered before it can spread far. Virus writers want their programs to spread as far as possible before anyone detects them. This brings up an important point which bears repeating:

It is a serious mistake to execute a program a few times - find nothing infected and presume there are no viruses in the program. You can never be sure that the virus simply hasn't triggered its infection phase!

Many viruses go **resident** in the memory of your PC just as a terminate and stay resident. This means the virus can wait for some external event such as inserting a diskette, copying a file, or executing a program to actually infect another program. This makes these viruses very dangerous since it's hard to guess what trigger condition they use for their infection. Resident viruses frequently corrupt the system software on the PC to hide their existence.

2. **Payload:** This is the malicious activity that the virus performs. Not all viruses have payloads, but there are some that perform destructive actions.

**Payload trigger:** This is the condition that causes the virus to activate or drop its destructive payload. Some viruses trigger their payloads on a certain date (Time Bomb). Others might trigger their payload based on the execution of certain programs or the availability of an Internet connection (Worms).

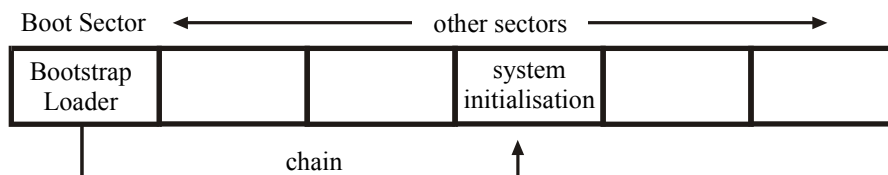
The second phase is the **attack phase**. Many viruses have unpleasant payloads, which may cause things such as deleting files or changing random data on your disk, simulating typos or merely slowing your PC down; some viruses do less harmful things such as playing music or creating messages or animation on your screen. Just as the virus's infection phase can be triggered by some event, the attack phase also has its own trigger. Viruses usually delay revealing their presence by launching their attack only after they have had ample opportunity to spread. This means that the attack may be delayed for years after the initial infection. The attack phase is optional, many viruses simply reproduce and have no trigger for an attack phase, hence no real payload. Does this mean that these are "good" viruses? No, unfortunately not! Anything that writes itself to your disk without your permission is stealing storage and CPU cycles. This is made worse since viruses which "just infect", with no attack phase, damage the programs or disks they infect. This is not intentional on the part of the virus, but simply a result of the fact that many viruses contain extremely poor quality code. One of the most common viruses, the *STONED* virus is not intentionally harmful. Unfortunately the author did not anticipate other than 360K floppy disks, with the result that the virus will try to hide its own code in an area on 1.2Mb diskettes which causes corruption of the entire diskette.

### 3.1.1 The different types of viruses

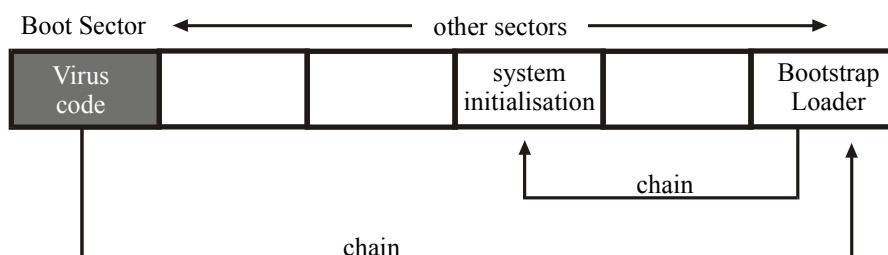
Now that we've examined general virus behaviour, let's take a closer look at the **major categories** of viruses and how they operate.

**Boot Sector Infectors.** All logical drives-hard disk and floppy-contain a boot sector, including disks that are not bootable. The boot sector contains specific information relating to the formatting of the disk and the data stored there. It also contains a small program called the boot program that loads operating system files. Under DOS, sectors are most commonly 512 bytes in length. These sectors are invisible to normal programs but are vital for correct operation of your PC. There are two types of system sectors found on DOS PCs, DOS boot sectors and partition sectors (also known as Master Boot Records or MBRs).

#### a) before infection



#### b) after infection



Boot sector viruses infect the boot program of the hard drive when an infected diskette is left in a floppy drive and the system is rebooted. These viruses plant themselves in your system sectors. When the computer reads and executes the boot sector program, the boot sector virus goes into memory and infects the hard drive. Later, when the user boots from the hard drive, the virus again gains control and can then infect each and every diskette used on the computer. Because every disk has a boot sector, computers can become infected by boot viruses on a "data disk" that has no programs or operating system.

System sector viruses (also commonly referred to as boot sector viruses) modify the program in either the DOS boot sector or the partition sector. Since there isn't much room in the system sector (only 512 bytes), these viruses often have to hide their code somewhere else on the disk (see graph).

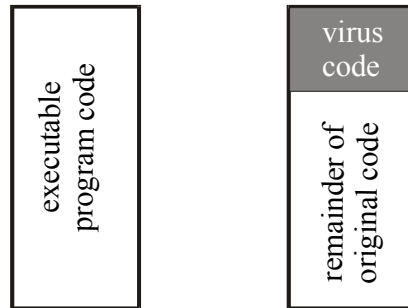
These viruses sometimes cause problems when this spot already contains data which is then overwritten. Some viruses, such as the *Pakistani BRAIN* virus mark the spot where they hide their code as having bad sectors. This is one reason to be alarmed if CHKDSK or Scandisk suddenly reports additional bad sectors on your disk. These viruses usually go **resident** in memory on your PC, and infect any floppy disk which you access. Simply doing a DIR on a floppy disk may cause it to be infected. Some viruses will infect your diskette as soon as you close the drive door. Since they are active in memory (resident), they can hide their presence. If *BRAIN* is active on your PC, and you use a sector editor to look at the boot sector of an infected diskette, the virus will intercept the attempt to read the infected boot sector and return instead a saved image of the original boot sector. You will see the normal boot sector instead of the infected version. Viruses which do this are known as **stealth viruses**. In addition to infecting diskettes, some system sector viruses spread by also infecting files.

**File Infectors.** These viruses attach themselves to or replace .COM and .EXE files, although in some cases they can infect files with the extensions .SYS, .DRV, .BIN, and OVL. This type of virus generally infects uninfected programs when they are executed with the virus in memory. In other cases, they infect programs when they are opened-using the DOS DIR command, for example-or the virus simply infects all of the files in the directory it was run from-a so-called **direct infector**.

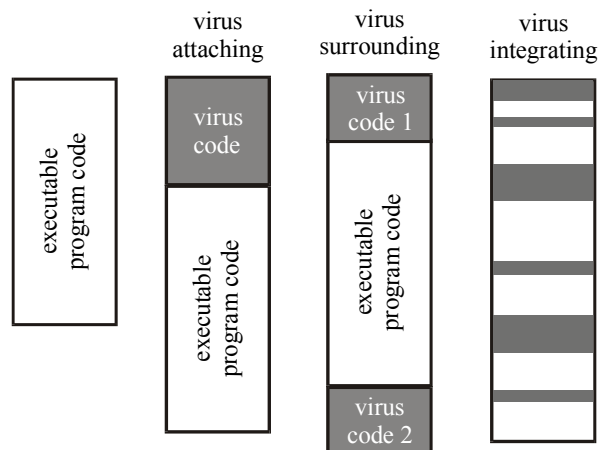
In terms of sheer number of viruses, these are the most common kind. The simplest file viruses work by locating a type of file that they know how to infect (usually a file name ending in ".COM" or ".EXE") and overwriting part of the program they are infecting (see graph on the right). When this program is executed, the virus code executes and infects more files. These overwriting viruses do not tend to be very successful since the overwritten program rarely continues to function correctly and the virus is almost immediately discovered.

The more sophisticated file viruses modify the program so that the original instructions are saved and executed after the virus finishes (see lower graph on the right). They can either do that by attaching to the file or by intercalating the original code. They can also use so-called stealth techniques (see later) to hide their presence. If you do a directory listing, you will not see any increase in the length of the file and if you attempt to read the file, the virus will intercept the request and return your original uninfected program to you.

a) before infection    b) after infection



Ways how viruses attach to files:



**Macro Viruses.** These type of viruses utilise the built-in macro language specific to some product such as *Microsoft Word*. Only users of this particular product can become infected by this type of virus. Typically files from *Microsoft Office* applications are attacked (e.g. *MS Word*, *MS Excel*, *MS Access*, etc.). The associated files (*MS Word* documents or templates and *MS Excel* spreadsheet files) are usually thought of only as data files so many people are surprised that they can be infected. But these files can contain programs (the macro language using *Visual Basic*, a BASIC like language) that are executed when you load one of these files into the associated product. The program inside of these files is interpreted by the *MS Office* application. What is now a language originally began as a very simple macro language that the user could use to combine keystrokes to automate some routine function. The macro language in these products has since grown substantially and now is a fully capable language. Almost all macro viruses (currently) are specific to the *MS Word* product.

What gives these viruses a chance to execute is the fact that *Microsoft* has defined special macros that will automatically execute. The mere act of opening an infected *MS Word* document or an infected *MS Excel* spreadsheet can allow the virus macros to be executed. (One simple prevention for this type of virus is to use the freely available (from *Microsoft*) viewer programs to rather than *MS Word* or *MS Excel* to view these type of files. Even *MS Access* database files (\*.mdb files) can contain macro viruses.

Macro viruses can **mutate** or become corrupted. A mutant macro virus is essentially a new virus with a different fingerprint, making it difficult to detect with existing fingerprints. In addition, macro viruses can also **mate** when they meet in the same document, creating a third macro virus that has elements of both parent viruses.

Whereas most viruses used to spread via floppy disks and program files, more infections occur now because of e-mail attachments and downloading from the Internet. According to the Virus Bulletin, eight of the top 10 reported viruses in March 1999 were macro viruses. In February 1999, that figure was even higher, with eight of the top 10 accounting for 83.9 % of all reported incidents.

### Other Virus Categories

Viruses, whether they are boot viruses, file viruses, or macro viruses, can employ none, one, or several of the following techniques to spread or conceal themselves.

**Multi-Partite Viruses.** Multi-partite viruses often infect multiple targets instead of just one type of file or disk. For example, they will infect both files and boot records on hard disks or both files and boot sectors on floppy disks.

**Polymorphic Viruses.** Polymorphic viruses mutate (change its byte pattern) to escape detection by anti-virus software. Both polymorphic file, boot sector, and macro viruses have been identified. They are also referred to as **Encrypted Viruses**. That is, it jumbles up its program code using encryption techniques (like ZIP) to make it difficult to detect. These type of viruses can't be detected with a simple pattern match as is possible with most viruses.

**Stealth viruses.** These viruses actively conceal themselves while they're running in memory. If the anti-virus program doesn't scan in memory for these viruses, it will completely miss them when scanning files. (e.g., if you try to read an infected file, it may appear uninfected.)

**Retro viruses.** These viruses are designed to actively attack anti-virus software. They're anti-anti-virus viruses! They'll try to delete anti-virus data files, corrupt anti-virus programs, and more.

## 3.2 Trojan Horses

A Trojan horse is an "apparently useful program containing hidden functions that can exploit the privileges of the user, with a resulting security threat. A Trojan horse does things that the program user did not intend.

Trojan horses rely on users to install them, or they can be installed by intruders who have gained unauthorised access by other means. Then, an intruder attempting to subvert a system using a Trojan horse relies on other users running the Trojan horse to be successful.

Any system can be affected by Trojan horses.

Hence it is a malicious, security-breaking program that is disguised as something benign, such as a directory lister, archiver, game or (in one notorious 1990 case on the Mac) a program to find and destroy viruses.

Users can be tricked into installing Trojan horses by being enticed or frightened. For example, a Trojan horse might arrive in email described as a computer game. When the user receives the mail, they may be enticed by the description of the game to install it. Although it may in fact be a game, it may also be taking other action that is not readily apparent to the user, such as deleting files or mailing sensitive information to the attacker.



### What does a Trojan horse do?

Trojan horses can do anything that the user executing the program has the privileges to do. This includes

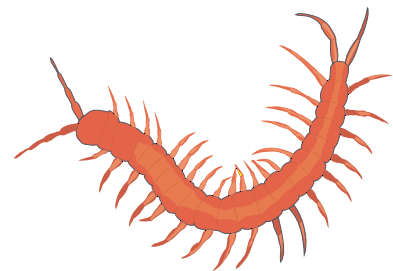
- deleting files that the user can delete
- transmitting to the intruder any files that the user can read
- changing any files the user can modify
- installing other programs with the privileges of the user, such as programs that provide unauthorised network access
- executing privilege-elevation attacks, that is the Trojan horse can attempt to exploit a vulnerability to increase the level of access beyond that of the user running the Trojan horse. If this is successful, the Trojan horse can operate with the increased privileges.
- installing viruses
- installing other Trojan horses

### 3.3 Worms

In the age of world-wide internet usage, worms are the threat to come. They might use email lists (like the *I-LOVE-YOU* worm) to spread and they might utilise Trojan horses to gain access to secure information or viral payloads to damage the system after spread to hide their trace.

The rise in Internet use is paralleled by an increase in Internet-borne malicious code carried by *Microsoft ActiveX* controls and *Sun Microsystems Java applets*. *ActiveX* or *Java* technology is downloaded to a user's hard drive and launched on the local computer, potentially with few security restrictions (in the case of *ActiveX*; *Java* is much more secure).

Although it has not yet happened, it is possible for virus writers to use *ActiveX* and possibly *Java* to introduce viruses, worms and Trojan horses onto a web-surfer's computer, turning Web pages into virus carriers. By simply surfing the Web, users could expose their computer to viruses spread via *ActiveX* controls, without downloading files or even reading e-mail attachments. The virus writers could then use the virus to access RAM, corrupt files, and access files on computers attached via a LAN, among other things.



## 4. The Danger

### Possible damages due to malicious software:

- **Causes system instability:** The malicious code might cause the computer to crash or to behave in an unexpected fashion.
- **Compromises security settings:** The malicious code might attempt to gain access to passwords or other system-level security settings. It might also search for openings in the Internet processing components of the computer to install a program on that system that could be controlled remotely by someone over the Internet.
- **Degrades performance:** The malicious code slows computer operations. This might involve allocating available memory, creating files that consume disk space, or causing programs to load or execute more slowly.
- **Deletes files:** The malicious code deletes various files on the hard disk. The number and type of files that might be deleted vary among viruses.
- **User loses confidence in system!** Causing decrease in productivity and efficiency.

### How Serious Are viruses?

Viruses are a problem but they are not the main thing you should be concerned about. There are many other threats to your programs and data that are much more likely to harm you than viruses. Problems such as hardware glitches, software conflicts, software bugs, and even typos are much more likely to cause undetected damage to your data than viruses. A well known anti-virus researcher once said that you have more to fear from a spilled cup of coffee than from viruses. While the growth in number of viruses now puts this statement into question, it's still clear that there are many more occurrences of data corruption from other causes than from viruses.

Because viruses have been deliberately written to invade and possibly damage your PC, they are the most difficult threat to guard against. It's pretty easy to understand the threat that disk failure represents and what to do about it, but the threat of viruses is much more difficult to deal with.

## Virus Myths

While viruses are capable of damaging systems, they **cannot do the following**:

1. Viruses don't infect files on write-protected disks.
2. Viruses don't infect compressed files. However, applications within a compressed file could have been infected before they were compressed. Some viruses are known to insert copies of themselves in already-created archives.
3. Viruses don't infect computer hardware such as monitors or computer chips; they only infect software. They can, however, damage certain types of hardware such as flash-memory.
4. *Macintosh* viruses don't infect DOS-based computer software, and vice versa. For example, the *Michelangelo* virus does not infect *Macintosh* applications. Again, an exception to this rule are the *Word* and *Excel* macro viruses, which infect spreadsheets, documents, and templates which can be opened by either *Windows* or *Macintosh* computers.
5. Viruses usually do not identify themselves as viruses, even after they do something destructive.



## Are Viruses Mostly Hype?

Unfortunately not! There is some confusion about this issue because some extreme claims have been made regarding numbers of viruses and how likely your software is going to become infected. During the *Michelangelo* media extravaganza in early 1991, some exaggerated figures were presented in the media which led some people to suspect that all viruses were nothing but hype. One company was quoted in *Information Week* that based on their reports, one out of four PCs was infected every month! You may also hear reports of there being from ten to thirty thousand different PC viruses with the number expected to double in six to nine months. So, are we faced with impending doom? No, not quite. The truth is viruses are very wide-spread but a relatively small number



(about one-hundred) account for 90% of all infections. Most of the twenty thousand viruses in virus-libraries are so poorly written that they will not spread in the real world. Many of these viruses are created by kids that can't even program. They use automated viruses creation programs that produce very poor quality viruses. These viruses are so obvious that they rarely spread in the wild. Still, viruses are a real threat that we can't afford to ignore. Viruses have been found on brand-new PCs, direct from the manufacturer, and on shrink-wrapped software, direct from the publisher. Viruses are not merely hype and no one is safe from potentially being infected. If you value your data and programs, you have to take some precautions.

## The new danger!

According to the International Computer Security Association (ICSA), diskettes are declining as a major source of virus infection, accounting for 68% of all reported infections in 1998 and 38% in 1999. Infections that spread through e-mail attachments-the source of macro viruses-increased from 32% in 1998 to 56% in 1999. E-mail attachments are the biggest source of macro viruses, while diskettes are the typical carrier for boot-sector viruses.

Understandably, an increase in viruses corresponds with an increase in the occurrence of virus infections. For example, a study by ICSA reports that the average rate of infection was 88 virus encounters per 1,000 computers during the month of February 1999 compared to only 32 per 1,000 for January 1998, and 14.9 virus encounters per 1,000 for January 1997. The study concludes that the figures show a "significant annual growth of approximately 20 encounters per 1,000 machines per month each year during that period.

The financial cost of virus infection, measured in cost per incident, has declined to \$2,454 in 1998 from \$8,100 in 1996, according to the ICSA study. The 1998 study also reports that complete recovery from an infection takes an average of 45.6 hours and 9.4 person-days of work. Often the cost is much more: one respondent to the study reported a cost of \$150,000 for a single incident. Clearly, viruses cause damage and waste time and manpower. What is not so clear is the extent of that damage. The ICSA study indicates that the reported costs of

virus infection would be much higher if related costs such as loss of business and lower productivity were taken into consideration.

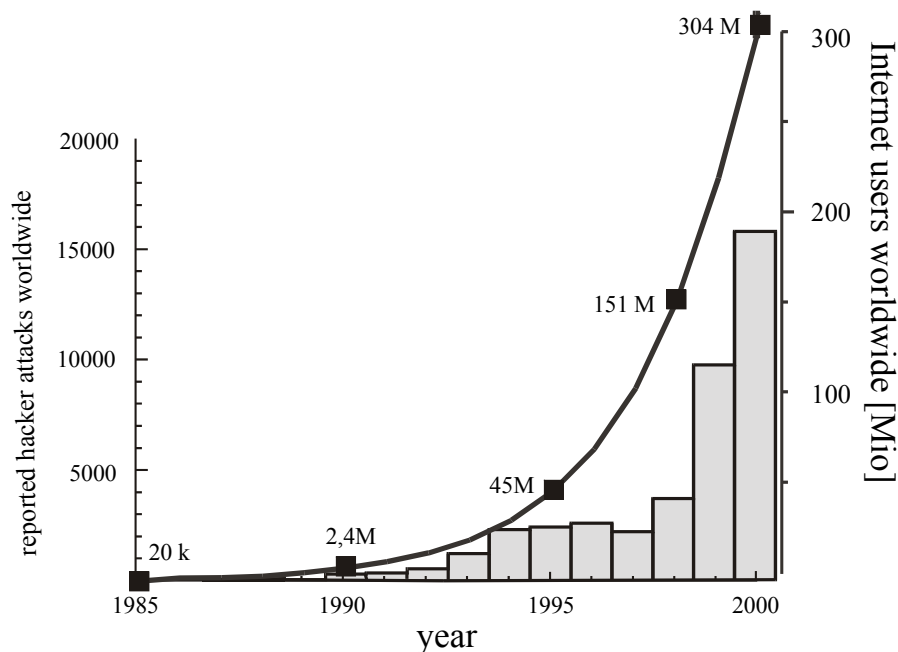
Causing everything from lost data to inaccessible files, computer viruses as well as worms and Trojan Horses are a drain on corporate bottom lines and employee patience. A rise in virus hoaxes, which can clog e-mail networks, can also result in downtime and lost productivity.

As the way people exchange electronic information changes, so does the nature of viruses. Today, viruses are moving away from platform dependency, forcing the old "binary" and boot-sector viruses into extinction. New viruses are able to migrate from *Windows 98* to *Windows NT* and back again. Script-based viruses and *Windows 32-bit* viruses represent the newest growth area.

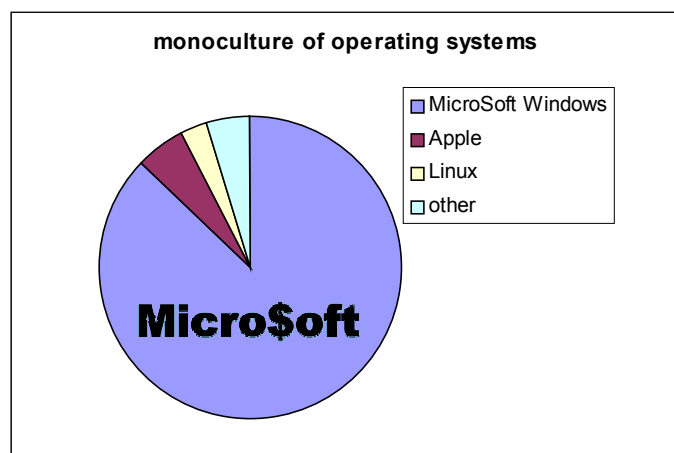
## The @-bomb?!

Here are some (fictious?) arguments why malicious code might (soon?) cause a real disaster:

- The increasing use of email and internet services provides therefore new dangers. Similar to the fact that 'new' biological viruses spread more rapidly by global mobility (cf. HIV).



- Another danger is the increasing dependence and inter-dependence of our life on computer-based services. All the possible threats discussed in circumstances of the Y2K-bug are also valid for new viruses. The distribution of electrical power, health services, transport guidance, banks etc. are susceptible to attacks of malicious code.
- The fact that the computer world is becoming increasingly a monoculture of I/O protocols (MIME, TCP/IP) based on -more or less- a single operating system (*MS Windows*) also increases the likelihood of fast virus spread with disastrous outcomes. (cf. The I-LOVE-YOU worm).
- Hacker and so-called 'script-kids' are getting more and more violent. The internet helps to spread ideas and even assembly kits to create viruses. Often frustrated teenagers start a hacker career for pure vandalism, while the old hacker scene was somewhat more interested in finding security leaks. The new motto might be: „I destroy therefore I am“.
- International terrorists and the organised crime might soon switch to this new arsenal of weapons and declare a 'cyber-war'.



- The military and its intelligence services might already consider or even use viruses as a new weapon.

## 5. Protection

### Virus Control

Viruses can be controlled at the desktop, the file server, the gateway, and on e-mail servers. Desktop and server anti-virus applications allow for virus scan and detection on an on-going and periodic basis, as well as each time a file is downloaded or a computer is booted. More and more, computer users have anti-virus software running full-time in the background, scanning all files and diskettes the moment they are accessed. As macro viruses proliferate, scanning e-mail attachments at the desktop is critical. To protect networks, monitoring attachments at the e-mail gateway is just as important (firewalls).

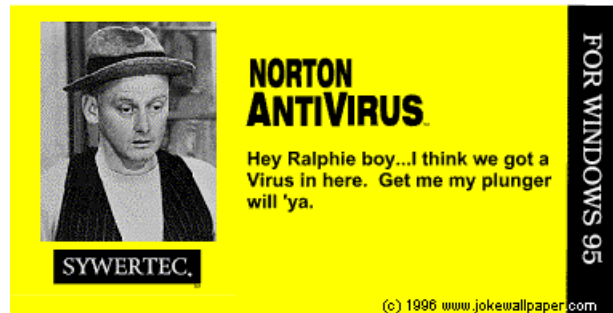


### Quick Virus Guidelines

It's important to keep viruses in perspective. They are but one threat to your data and programs. They need not be regarded as mysterious and they are quite easy to understand. Here are a few tips

to keep in mind when considering viruses:

- Make sure you run a **virus-scanner** (better two different brands) routinely (every week or so, depending on your computer's contact with the 'environment').
- You can only get a virus by executing an infected program or booting from an infected diskette. Any diskette can be infected by a boot sector virus, even non-bootable diskettes.
- You cannot (yet) get a virus simply by being on the internet, or an online service. You will only become infected if you download an infected file and execute that file. (It's important to understand that *Microsoft Office* files act as executable programs since they can contain macro programs that are executed when 'open' the file; so, to be safe, a *MicroSoft Word* document or *Excel* Spreadsheet should not be opened with the actual *Microsoft* application but rather with a viewer program such as those available from the *Microsoft* web site or simply disregarded.)
- Most viruses are transferred by booting from an infected diskette (e.g. *Stoned*, *Form*, *Stealth-B*, *AntiExe*, *Monkey*). Remove diskettes from your floppy drive as soon as you are through with the diskette. If your CMOS permits it, change your boot order to boot from your hard disk first. If you don't know what CMOS is, check the manual for your PC; there is normally an option when you boot your PC to hit a specific key to enter CMOS set-up. This allows you to change many options on your PC.
- Make sure you have at least two **backups** for all of your files. Backups are essential not only to safely recover from virus infections, but also to recover from the other threats to your data.
- Be sure to **check all new software** for viruses. Even shrink-wrapped software from a major publisher may contain a virus.



### References:

1. P. Pfleeger: „*Security in Computing*“ 2<sup>nd</sup> ed., Prentice Hall 1997 (ISBN: 0-13-185794-0)
2. Alexander: "*The underground guide to computer security*" Addison-Wesley Publ., 1996 (QA 76.9.A25)
3. Caelli: "*Information security handbook* " Basingstoke, Macmillan, 1991 (QA 76.9.A25, 5 copies)
4. <http://www.symantec.com>
5. <http://www.datafellows.com/v-descs/>
6. <http://nsi.org/Computer/threats.html>



## Appendix A: Today's Most Frequently Reported Viruses

Although there are now more than fifty thousand different viruses, only a small number of viruses account for most of the infections. The following viruses account for 98 % of all viruses reported.

Following is a list of the top reported viruses as of July 1999:

1. W97M.Melissa. W97M.Melissa.A is a typical **macro virus** (incl. features of a **Trojan horse** and **worm**) that has an unusual payload. When a user opens an infected document, the virus attempts to e-mail a copy of this document to up to 50 other people, using *Microsoft Outlook*. The virus turns off security protection upon opening an infected document in *MS Word 2000*, disabling the macro prompt the next time the document is opened. The virus infects *MS Word 97* and *MS Word 2000* documents by adding a new macro module named Melissa.
2. Worm.ExploreZip. First found in Israel, this **worm** contains a malicious payload, utilising *MAPI* commands and *MS Outlook* on *Windows* systems to propagate itself. The worm e-mails itself out as an attachment with the filename "zipped\_files.exe"; the body of the e-mail message appears to come from a known e-mail correspondent. The worm determines the recipient by going through received messages in the user's inbox. Once the attachment is executed, the worm copies itself to the user's directory and modifies the WIN.INI file so that the program is executed each time Windows is started. The worm then utilises the user's e-mail client to harvest e-mail addresses in order to propagate itself. When executed, the worm also searches through the C through Z drives of the user's computer system and selects a series of files (of any file extension) to destroy by making them 0 bytes long. This can result in non-recoverable data and operable computers.
3. WM.CAP.A, alias WordMacro/Cap.A. This *MS Word* **macro virus** consists of 10 macros, all stored in encrypted form. WM.CAP.A has a stealth feature that hides the menu item from the Tools menu and the Templates menu item from the File menu when the NORMAL.DOT file is infected. This prevents the user from checking the list of macros contained in the document or template, and hides the macros. It has no intentional payload or trigger.
4. W97M.Ethan.A, alias Ethana. This is a *Word 97* **macro virus** that inserts its viral code into the beginning of the "ThisDocument" *Visual Basic* module. While closing an infected document, the virus, with 30% chance, modifies several fields in the File Summary Information menu. This macro virus also removes the temporary text file "C:\CLASS.SYS" that most W97M.Class variants use.
5. W97M.Marker. This is a common **macro virus** with a unique payload that adds its viral code to the "ThisDocument" *Visual Basic* module. It also uses a randomly named temporary text file while infecting. This macro virus will keep the date/time of the infection and user information. When the payload in the virus activates on the 1<sup>st</sup> of the month, it will upload this information to an FTP site.
6. PictureNote.Trojan, alias Trojan Horse, Backdoor.Note, Picture.exe, and URLSnoop. This is a malicious program that is often identified and referred to as a **Trojan Horse**. It does not have the capability to spread like a virus. The program is sent through Internet e-mail as an e-mail attachment named PICTURE.EXE. When this file is executed, it ultimately searches for *America Online* user information on that computer, possibly stealing the user's *AOL* password information.
7. Happy99.Worm, alias Trojan.Happy99 and I-Worm.Happy. This is a **worm** program, not a virus. The program file is usually sent as an e-mail attachment or an article attachment. When executed, the program shows a fireworks display as it copies itself as SKA.EXE and extracts a DLL that it carries as SKA.DLL into the Windows\System directory. It also modifies WSOCK32.DLL and copies it into WSOCK32.SKA. This allows the worm routine to be triggered when a connect or send activity is detected. When such online activity occurs, the modified code loads the worm's SKA.DLL. This DLL creates a new e-mail or a new article with UUENCODED HAPPY99.EXE inserted into the e-mail or article. It then sends this e-mail or posts this article.
8. XM.Laroux, alias ExcelMacro/Laroux, Excel.Laroux, and Laroux. This virus is the first working *Excel* **macro virus** found in general circulation. The actual virus code consists of two macros called Auto\_Open and Check\_Files. The macros are stored in a hidden datasheet named "laroux." When an infected spreadsheet is opened, the Check\_Files macro copies the worksheet with the virus code into a spreadsheet file stored in the *Excel* startup directory named "Personal.xls." This enables the infection of all other spreadsheets opened or created on the infected system in the future. XM.Laroux contains no deliberately destructive payloads; it exists only to replicate.

9. W95.CIH, alias Chernobyl and CIH.Spacefiller. CIH is a very destructive **virus** with a payload that destroys data. It infects 32-bit *Windows 95/98/NT* executable files but is only capable of functioning under *Windows 95/98*. When an infected program on a *Windows 95/98* machine is run, it becomes resident in the computer's memory. An infected system, therefore, must be rebooted from a clean system disk before scanning with an anti-virus product. The virus includes two payloads; the first is designed to overwrite the hard disk with random data and the second tries to cause permanent damage to the computer by attacking the Flash BIOS.
10. W97M.Class, alias Class.Poppy. This **polymorphic W97M macro virus** does not add a new *Visual Basic* module; instead, it adds viral code to the "ThisDocument" VB module which, by default, is always in *Word97* document/template. Most variants have a payload that displays messages on certain dates of the year.

## Appendix B: Examples of malicious software

**W95.CIH virus:** alias: Chernobyl, PE\_CIH, Win95.CIH,

Infection length: Up to 1KB

Payload: Destroys data and possible damage to CMOS, triggered on April 26th commemorating Chernobyl (the anniversary of the April 26, 1986 Soviet nuclear disaster)

The W95.CIH virus, often referred as the CIH or the Chernobyl virus, was first discovered in June of 1998 in Taiwan. According to the Taipei authorities, the CIH virus was written by a 24 year old man named Chen Ing-hau (note the name of the virus derived from his initials).

CIH is a very destructive virus with a payload that destroys your data. On April 26, 1999, the payload triggered for the first time, and caused many computer users to loose their data. CIH is a virus that infects 32-bit *Windows 95/98/NT* executable files but only capable to function under *Windows 95/98*. When an infected program on a *Windows 95/98* machine is run, the virus will become resident in computer's memory. This means that an infected system must be rebooted from a clean system disk before scanning with an anti-virus product. If this is not done, the virus will infect every file that the anti-virus product scans. Files infected by CIH may have the same size as the original files because of CIH's unique mode of infection. The virus will search for empty, unused spaces in the file. Next it will break itself up into smaller pieces and inserts them in these unused spaces.

In Korea, it was estimated as **many as one million** computers were affected resulting in more then **\$250 million in damages**. You are only at risk if you are not using antivirus software with up-to-date virus definition files. In the U.S., Symantec's technical support confirmed over 500 reports from home users that experienced the destructive payload because they were not using any antivirus software. However, only one report was received from corporate users in the U.S. Most corporate users had already purchased antivirus software or updated their virus definitions due to the Melissa virus outbreak in late March 1999.

In April of 2000, although the virus is rather old, it is still believed the virus is in the wild and may cause damage to PC users that are using very outdated virus definitions or not using antivirus software.

**Happy99:** alias: Trojan.Happy99, I-Worm.Happy, W32.Ska, Happy00

Infection length: 10,000 bytes

HAPPY99.EXE is a **worm** program, not a virus. This program has reportedly been received through email spamming and USENET newsgroup posting. The file is usually named HAPPY99.EXE and appears as an attachment to an email or article. When executed, the infected program opens a window entitled "Happy New Year 1999 !!!" and shows a firework display to disguise its installation. This worm sends itself to other users when the infected computer is online. Doesn't cause further damage.

**W97.Melissa.A:** alias: Mailissa

This is a **macro virus** which has an unusual payload which makes it also a **worm**. When a user opens an infected document, the virus will attempt to e-mail a copy of this document to up to 50 other people, using *Microsoft Outlook*.

The virus turns off the security protection upon opening an infected document in *MS Word 2000*. This disables MS Word 2000 macro prompt the next time the document is opened.

It infects a MS Word 97 and MS Word 2000 document by adding a new module named Melissa. Although there is nothing unique in the infection routine of this macro virus, it has a payload that utilises *MS Outlook* to send an attachment of the infected document being opened.

As its primary payload, the virus will attempt to use *Microsoft Outlook* to e-mail a copy of the infected document to up to 50 other people. When a user opens or closes an infected document, the virus first checks to see if it has done this mass e-mailing once before, by checking the following registry key:

"HKEY\_CURRENT\_USER\Software\Microsoft\Office\" as "Melissa?" value.

If this key has a value "Melissa?" set to the value "...by Kwyjibo", then the mass e-mailing has been done previously from the current machine. The virus will not attempt to do the mass mailing a second time, if it has already been done from this machine.

If it does not find the registry entry, the virus does the following:

1. Open *MS Outlook*.
2. Using MAPI calls, it gets the user profile to use *MS Outlook*.
3. It creates a new e-mail message to be sent to up to 50 addresses listed in the *user's MS Outlook* address book.
4. It gives the email message a subject line:  
"Important Message From USERNAME",  
where USERNAME is taken from *MS Word* setting.
5. The body of the email message is:  
"Here is that document you asked for ... don't show anyone else ;-)"
6. It attaches the active document (the infected document being opened or closed) to the email message.
7. It sends the e-mails.

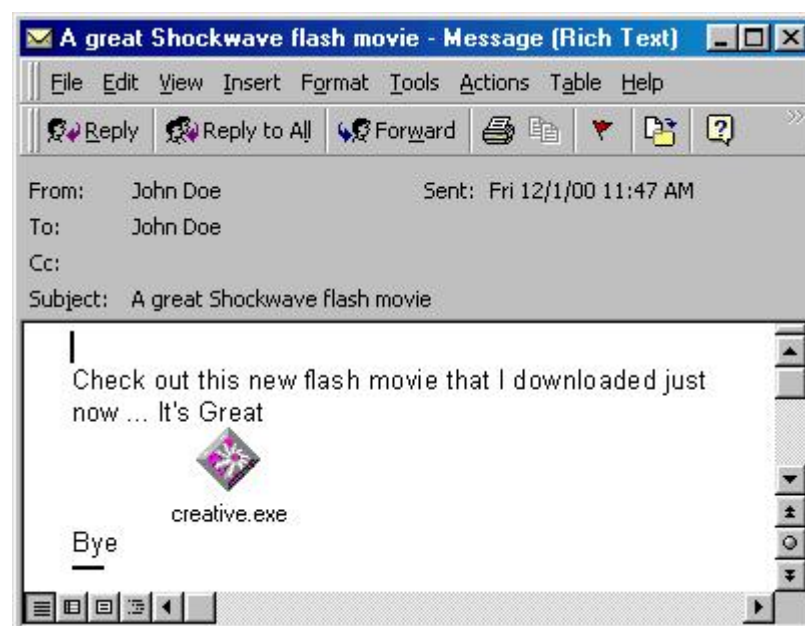
There is a second payload which triggers once an hour, at the number of minutes past the hour corresponding to the date (i.e., on the 16th of the month, the payload triggers at 16 minutes after every hour). If an infected document is opened or closed at the appropriate minute, this payload will insert the following sentence into the document:

" Twenty-two points, plus triple-word-score, plus fifty points for using all my letters. Game's over. I'm outta here."

Note that the virus will also infect other documents on the user's machine, using the normal infection mechanisms of macro viruses, even if the user does not have *MS Outlook*. So, it is potentially possible for a new document from any user's machine to be e-mailed to other people .

**Creative:** alias: Prolin, Shockwave, W32/Prolin@mm,TROJ\_SHOCKWAVE, TROJ\_PROLIN

Prolin is an e-mail **worm** that spreads itself using *MS Outlook*. The worm itself is a *Windows* EXE file about 37Kb long written in *VisualBasic*. The worm uses the standard "Melissa"-like way of spreading itself: it opens *MS Outlook's* address book, gets e-mail addresses from there and sends its copies to these addresses. The infected messages look like the graph below.



The worm then sends a notification message to his author and informs him about another infected computer:



Then the worm installs itself twice to system. The second copy is specially placed in auto-run directory, so it will be activated during every Windows session. The worm has a dangerous payload. It scans all available disk drives, gets ZIP, MP3, and JPG files and renames them.

For example, BGAMEX.JPG and DATA.ZIP are moved to:

C:\BGAMEX.JPG change atleast now to LINUX

C:\DATA.ZIP change atleast now to LINUX

The worm also creates a text file "messageforu.txt" in root C:\ folder writes some text to there and adds a list of renamed files to the end:

Hi, guess you have got the message. I have kept a list of files that I have infected under this. If you are smart enough just reverse back the process. i could have done far better damage, i could have even completely wiped your haddisk. Remember this is a warning & get it sound and clear... - The Penguin

C:\WINDOWS\SYSTEM\OOBE\IMAGEX\BGAMEX.JPG

C:\BACKUP\DATA.ZIP

## VBS.LoveLetter.A, I-LOVE-YOU worm

Discovered on: August 31, 2000, 29 versions of this worm are identified so far

This worm appears to originate from Manila, Philippines and was written by the frustrated student Onel de Guzman. He actually wrote an essay about such a strategy, which was rejected by his teacher.

It has wide-spread distribution and infecting millions of computers in 2000 (some news agencies were infected, newspapers couldn't be printed, Belgian bank tellers stopped working, 80% of the US government computers were infected, incl. Pentagon and the foreign ministry). The damage was estimated to amount 7 billion pounds world wide.

This worm sends itself to email addresses in the *Microsoft Outlook* address book and also spreads itself into Internet chatrooms via *mIRC*. This worm overwrites files on local and remote drives, including files with the extensions

.vbs, .vbe, .js, .jse, .css, .wsh, .sct, .hta, .jpg, .jpeg, .wav, .txt, .gif, .doc, .htm, .html, .xls, .ini, .bat, .com, .mp3, and .mp2.

The contents of these files will be replaced with the source code of the worm, thus destroying the original contents. The worm will also append the extension '.vbs' to each of these files. For example, the file image.jpg will become image.jpg.vbs. However, files with .mp2 and .mp3 extensions will merely be hidden from the user's view and not actually destroyed. It also tries to download a password-stealing Trojan horse program from a Web site (disabled by now).

The subject of the message is: ILOVEYOU

The body of the message is: kindly check the attached LOVELETTER coming from me.

Attached to the message is the file: Love-letter-for-you.txt.vbs

**Deep Throat:** alias Win32.DeepThroat

Deep Throat is a hacker's remote administration tool (**Trojan horse**, example of advanced hacker techniques), much like the infamous **Back Orifice** and NetBus tools. Deep Throat allows a hacker to access data and gain control over some *Windows* functions on remote system.

Deep Throat tool has client and server parts. The server part is installed on a remote system to be accessed. The server part can be dropped to a TEMP directory with a random name by a special dropper. On execution the server part installs itself to Windows directory and it will be executed automatically during next Windows startup.

The server part hides its process name in *Windows* task manager. Access to the running server part file is denied by *Windows* so it can't be removed easily while *Windows* is running.

The client part allows to control the remote computer system where the server part is installed and active. The client part has a dialog interface which allows to perform tricks on remote system and to receive/send data, text and other information (some features are not implemented in version 1.0).

Below is the list of Deep Throat features:

1. Open and close CD-ROM tray
2. Show a message box on remote system with optional text
3. Hide or show Windows taskbar
4. Start FTP server on port 21 - upload/download files (not implemented in v1.0)
5. Capture screen to JPG image and receive it from remote system
6. Open optional URLs with browsers on remote system
7. Turn monitor on/off - send powersave mode commands
8. Steal passwords
9. Run any program on target system
10. Run any program on target system in invisible mode
11. Reboot Windows
12. Port scanner - to scan for computers with DT server running
13. Ping remote system - to check if there's a running DT server
14. Get remote system info



The client and server parts use TCP/IP protocol to communicate with each other. The client part has an option to scan a range of IP addresses to search for active server part and connect to it.

## Appendix C: A HACKERS' GLOSSARY

- **back door:** In the security of a system, a hole deliberately left in place by designers or maintainers. May be intended for use by service technicians. Synonym: trap door.
- **bit bucket:** The universal data sink. Discarded, lost or destroyed data is said to have gone to the bit bucket. Sometimes amplified as The Great Bit Bucket in the Sky.
- **cracker:** One who breaks security on a system. Coined by hackers in defence against journalistic misuse of the term "hacker." The term "cracker" reflects a strong revulsion at the theft and vandalism perpetrated by cracking rings. There is far less overlap between hackerdom and crackerdom than most would suspect.
- **deep magic:** An awesomely arcane technique central to a program or system, esp. one that could only have been composed by a true wizard. Many techniques in cryptography, signal processing, graphics and artificial intelligence are deep magic.
- **hacker:** 1. A person who enjoys exploring the details of programmable systems and how to stretch their capabilities. 2. One who programs enthusiastically. 3. A person who is good at programming quickly. 4. An expert at a particular program, as in 'a Unix hacker'. 5. [deprecated] A malicious meddler who tries to discover sensitive information by poking around. The correct term for this sense is "cracker."

- **KISS Principle:** "Keep It Simple, Stupid." Often invoked when discussing design to fend off creeping featurism and control development complexity. Possibly related to the marketroid maxim, "Keep It Short and Simple."
- **kluge:** 1. A clever programming trick intended to solve a particularly nasty case in an expedient, if not clear, manner. 2. Something that works for the wrong reason.
- **lots of MIPS but no I/O:** A person who is technically brilliant but who can't seem to communicate with human beings effectively. Technically it describes a machine that has lots of processing power but is bottlenecked on input-output.
- **munge:** 1. [derogatory] To imperfectly transform information. 2. A comprehensive rewrite of a routine, data structure or whole program. 3. To modify data in some way that the speaker doesn't need to go into right now.
- **netiquette:** The conventions of politeness recognized on Usenet, such as avoidance of cross-pointing to inappropriate groups and refraining from commercial pluggery outside the biz groups.
- **phreaking:** 1. The art and science of cracking the phone network (so as, for example, to make free long-distance calls). 2. By extension, security-cracking in any other context (especially, but not exclusively, on communications networks).
- **raster burn:** Eyestrain brought on by too many hours of looking at low-res, poorly tuned or glare-ridden monitors, esp. graphics monitors.
- **RTFM** [Acronym for 'Read The F..... Manual.'] 1. Used by gurus to brush off questions they consider trivial or annoying. 2. Used when reporting a problem to indicate that you aren't just asking out of randomness: "Yes, I RTFM first."
- **security through obscurity:** (alt. security by obscurity) A hacker term for vendors' favorite way of coping with security holes -- namely, ignoring them; documenting neither any known holes nor the underlying security algorithms; or trusting that nobody will find out about them, and that people who did find about them won't exploit them. This "strategy" never works for long.
- **sneaker:** An individual hired to break into places in order to test their security; analogous to "tiger team."
- **spaghetti code:** Code with a complex and tangled control structure, esp. one using many GOTOs, exceptions or other 'unstructured' branching constructs. Pejorative. The synonym kangaroo code has also been reported, doubtless because such code has so many jumps in it.
- **vaporware:** Products announced far in advance of any release.
- **voodoo programming:** The use by guess or cookbook of an obscure or hairy system, feature or algorithm that one does not truly understand. The implication is that the technique may not work, and if it doesn't, one will never know why.
- **Vulcan nerve pinch:** The keyboard combination that forces a soft-boot or jump to ROM monitor (on machines that support such a feature). On many micros this is Ctrl-Alt-Del; on Suns, L1-A; on some Macintoshes, it is ! Also called the "three-finger salute."
- **wedged:** 1. To be stuck, incapable of proceeding without help. This is different from having crashed. If the system has crashed, it has become totally nonfunctioning. If the system is wedged, it is trying to do something but cannot make progress. 2. Often refers to humans suffering misconceptions.
- **wetware:** 1. The human nervous system, as opposed to computer hardware or software. 2. Human beings (programmers, operators, administrators) attached to a computer system, as opposed to that system's hardware or software.
- **wizard:** A person who knows how a complex piece of software or hardware works; esp. someone who can find and fix bugs quickly in an emergency. Someone is a hacker if he or she has general hacking ability, but is a wizard only if he or she has detailed knowledge.
- **zipperhead:** A person with a closed mind.

