SOPHOS

Sophos Reference Guide 1998/1999

Sophos Reference Guide

1998/1999

ISBN 0 9513420 8 8

Eleventh edition, published in June 1998 by Sophos Plc.

Sophos Plc • The Pentagon • Abingdon • OX14 3YP • England

Email enquiries@sophos.com • http://www.sophos.com/

Tel +44 1235 559933 • Fax +44 1235 559935

Edited by Michael Lewis and Paul Oldfield, Sophos Plc, Abingdon.

9 8 7 6 5 4 3 2 1

Part # su000017/980526

Contents

Data security 15

Sophos Data Security products 159

Foreword

The *Sophos Reference Guide* provides information to help computer users protect their data. It has five sections:

Data security

This introduces the principles of data protection and discusses encryption, authentication, secure erasure of data, and access control. It also includes chapters on Internet security and the millennium bug.

Computer viruses

This deals with computer viruses and measures for their prevention, detection and elimination. It also discusses other forms of software attack, as well as virus hoaxes and scares.

Sophos Data Security products

This details the Sophos Data Security range, including encryption toolkits, secure erasure utilities and disk authorisation systems.

Sophos Anti-Virus products

This introduces Sophos Anti-Virus, including InterCheck, Sophos’ unique client-server system for on-access virus detection, and discusses its deployment on large networks. This section also includes briefings on the different versions of Sophos Anti-Virus for various platforms.

Additional information

This section details security standards and sources of further information.

Data security

N

Introduction to data security

Data security techniques

Security of data can be achieved by ensuring its **Confidentiality**, **Integrity** and **Availability (CIA)**.

Various methods such as **encryption**, **authentication**, **secure erasure, access control** and **anti-virus measures** can be used to achieve these aims.

Aims Methods	Confidentiality	Integrity	Availability
Encryption	✔	✔	
Authentication		✔	
Secure erasure	✔		
Access control	✔	✔	✔
Anti-virus measures		✔	✔

Methods of ensuring confidentiality, integrity and availability of data

Encryption means rearranging or disguising information so that it cannot be understood by an unauthorised person. Methods for doing this have ranged from simple substitution codes, used as early as Etruscan times, to highly sophisticated modern computer-based techniques. Encryption can be used very effectively to protect data held on computer systems or transmitted between them. See the 'Data encryption' chapter for more information.

Authentication is a technique used between the sender and receiver to validate the source and the text of a message. Disputes over the contents of a message can be avoided, and forgeries, tampering or transmission errors are always detected. Authentication is also a powerful tool in ensuring the integrity of stored data; it is used to combat program corruption, viruses and Trojan horse attacks. See the 'Authentication' chapter for more information.

Secure erasure is used to remove file contents completely from magnetic media such as disks. Deleting files using standard system commands often only removes the file name from the directory, leaving the actual contents of the file on the disk. See the 'Secure erasure' chapter for more information.

Access control is the technique for preventing an intruder from accessing computer resources. In its simplest form it is **physical access control** by means of a lock on the door of the computer room or a lock on the computer itself. **Logical access control** refers to the separation of the users' processes and data, as well as the separation between the users and the operating system. See the 'Access control' chapter for more information.

Anti-virus measures ensure that computer viruses do not corrupt or delete data. They include prevention, detection and recovery. See the 'Introduction to computer viruses' chapter and the chapters that follow for more information.

Dangers to stored information

Much of the information stored on computer systems is confidential. Typical examples include company finances, payrolls, lists of contacts, proprietary programs and technical information.

Personal computer resources need protection against unauthorised modification, disclosure or destruction of information. The potential damage is often great - for example if an employee were to read or, worse still, modify the company payroll file.

Large computer systems generally use 'logical' access control mechanisms, such as login passwords and different user areas on the disk. These offer good protection against casual attempts to access unauthorised information, but can usually be overcome relatively easily by experienced programmers or system operators.

A good answer to problems such as these is to protect information by encrypting it. Encryption means scrambling it into a coded form in which it cannot be read or used, even by 'hackers' and computer specialists.

When the integrity, rather than confidentiality, of programs or stored data is of importance, authentication, a mathematical technique for detecting changes in information, can be used to protect them.

Dangers to information in transit

When information of any sort is communicated, it is vulnerable.

Taking the example of an email, the dangers can be:

- **Accidental**, e.g. a confidential email being read by an unauthorised user (on an unattended PC for example), or being delivered to the wrong recipient.

- **Intentional**, e.g. if the email is deliberately intercepted, or its contents have been falsified for fraudulent purposes.

Such dangers threaten not only email, but also normal written documents, financial computer networks, etc.

Cryptographic techniques can be used to protect communicated information against:

- **Interception** (eavesdropping) by unauthorised or hostile third parties. If the information were encrypted it would be meaningless without the correct encryption key.
- **Falsification**, for example: alteration, deletion or addition of data (tampering); changing the apparent origin of data (forgery); using previously

Applicability of methods	Stored data	Transmitted data
Encryption	✔	✔
Authentication	✔	✔
Secure erasure	✔	
Access control	✔	
Anti-virus measures	✔	✔

Protecting stored and transmitted data

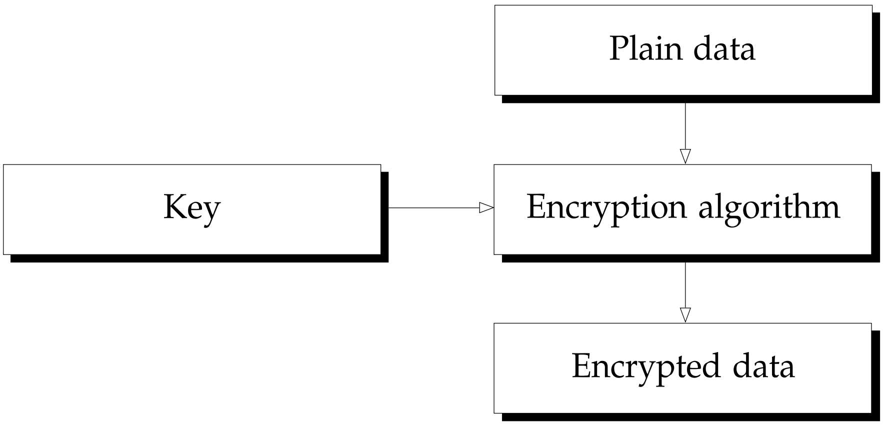
transmitted or stored data again; falsifying an acknowledgement; errors in transmission. Authentication can be used to validate the source and content of the information.

Data encryption

What is encryption?

Encryption (or scrambling) means rearranging or disguising information so that it cannot be understood by an unauthorised person. Simple encryption based on letter substitution was already used as early as Etruscan times, while more sophisticated systems, such as the famous 'Enigma' machine, were used in the Second World War. Since the 1970s the advance of digital computers has made possible tremendous improvements in the security of encryption methods: these are described in the 'Encryption algorithms: DES and RSA' section.

Modern encryption techniques perform their scrambling using a **key** chosen at random from a very large number of possibilities. This key can be thought of as a type of password to access the information.



Data encryption

What is decryption?

The reverse process to encryption, i.e. recovering the original data from the encrypted version, is called **decryption**. If decryption is carried out using any key other than the correct one, the result is meaningless.

How secure is encryption?

Encryption can be used very effectively to protect data held on computer systems, or transmitted between them, from unauthorised access. It is generally applied only to information which really needs protecting, rather like keeping only confidential documents in the safe.

Good encryption, used well, is **the** most secure way of protecting confidential data against unauthorised disclosure. It is a more fundamental protection than the access control systems found on many computers, which are often relatively simple for ‘hackers’ or computer specialists to defeat.

Cracking an encryption code (i.e. finding out the original confidential data from the encrypted version) is a very difficult problem, governed by powerful laws of mathematical intractability.

Modern encryption methods are designed in such a way that it is not possible, even knowing the precise technique which was used, to trace back the manipulations which have taken place and thus ‘unravel’ the encryption. This is because the basic tool, the encryption method, works in conjunction with an unpredictable key, as already described.

However, the **overall** security provided by such a tool depends very much upon the way it is used. The important issues are how the encryption keys are chosen, to whom they are made known, and so on. Encryption offers no protection if the relevant key becomes known to an unauthorised person. The problems of choosing, distributing and changing keys are known collectively as **key management**.

Key management

Encrypted information is safe as long as the key used for encryption is safe. There should exist a stringent set of rules on how and when the keys are created, where they are stored, who has access to them etc.

This is like choosing the combination for the lock on a safe and ensuring that it is not disclosed. Encryption keys are usually long words, phrases or numbers which are as difficult as possible to guess. In some systems, the key is generated as a random number.

When encrypted information is transmitted between two parties, e.g. as an email, the key has to be communicated from the sender to the recipient in a secure way. In such circumstances, key management can be made much simpler and safer by using a **public key** system. This approach is used in the Sophos product PUBLIC.

Key management is simpler (though no less important) if the key does not have to be communicated between two parties. When encrypting data stored on disk, for example, the person who decrypts the data is usually the same person who encrypted it, and the special problem of key communication does not arise.

Choosing keys and passwords

Most data security systems require their users to choose keys (for encryption of data) or passwords (usually for access to part of a system). In either case, similar rules apply.

Experience shows that most people's choices are highly insecure; they tend to choose short words such as names, which are easy to guess. According to a survey, the most commonly used passwords are 'PASSWORD', 'PASS', 'FRED', 'LOVE', 'SEX', 'GOD', 'GENIUS', 'HACKER' and 'SECRET'.

Here are some fundamental rules for choosing passwords and keys:

DON'Ts

- DON'T choose familiar words or phrases such as first names, surnames, telephone numbers, dates of birth, car registration numbers etc.
- DON'T write your key or password down in an obvious place like the inside of your drawer, underneath the computer keyboard, or on a floppy disk used for storing encrypted files.
- DON'T use a new password or key which depends on the old one.
- DON'T tell anyone else your password.
- DON'T choose a password which is fewer than 8 characters long.

DOs

- DO choose a long, obscure phrase, rather than a short and obvious word.
- DO make your choice as random as possible.
- DO try to mix upper and lower case characters as well as numbers.
- DO change your keys as often as possible, but not in a predictable pattern.
- DO change a key or password immediately if you suspect someone may have seen you typing it in.

Software or hardware encryption?

Encryption can generally be performed either by a program running on a general-purpose computer ('software encryption'), or by a special, dedicated electronic device ('hardware encryption'). While the end result is identical, there are nevertheless considerations which can make one of the two methods more suited to particular applications.

Hardware devices typically encrypt information 10 to 100 times faster than software implementations of the same method. This is the main advantage of hardware; there are some applications in which speed is essential. Hardware devices are also easier to protect physically, but are expensive. Almost all such devices still need driving software to be usable.

Software is generally less expensive than hardware. It is more flexible and easier to integrate into computer-based communication systems. It is also much easier to upgrade and only marginally more prone to attack by an intruder. Its main disadvantage is generally the lower encryption speed.

If the encryption algorithms used for hardware and software encryption are identical, hardware devices and software encryption can be used interchangeably. Data can be encrypted using hardware and decrypted using software and vice versa.

The choice between hardware and software demands careful evaluation of the requirements.

Encryption algorithms: DES and RSA

There are two well-established standard encryption algorithms (i.e. rules which specify the method of encryption), both used in Sophos products.

DES (which is an abbreviation for the 'United States National Bureau of Standards Data Encryption Standard') is a highly secure algorithm for encrypting or decrypting 64 bits of data on the basis of a 56-bit key. DES is very widely used, particularly in the banking world. It is known as a 'symmetric' algorithm because it encrypts and decrypts data using the **same** key.

Not only are there more than 70,000,000,000,000,000 possible DES keys, but there is no known way of cracking DES other than trying all these keys until the correct one is found. The technical term for this is ‘exhaustive key searching’. For a general-purpose computer, this is not a practical task: it would take hundreds of thousands of years. However, if the search is divided up and allocated to a very, very large number of machines (as has been done for demonstration purposes), a key can be cracked within months.

The **RSA algorithm** was invented by Rivest, Shamir and Adleman in 1976, and represents one of the most important advances in cryptography in recent years. Its special property is that it uses **different** keys for encryption and for decryption; in technical terms it is a ‘public key’ or ‘asymmetric’ cipher. A user can make their encryption key widely known, but keep their decryption key secret. Anyone can therefore encrypt a message to send to that user, but only that user will be able to decrypt and understand it. There is no need to agree a secret key between sender and recipient. This property makes the RSA algorithm particularly suitable for **communicating** sensitive information as well as for computing **digital signatures**.

Unlike DES (for which the numbers of key and data bits are precisely specified), the RSA algorithm can be implemented using any number of bits. The greater the number, the more secure the cipher, but the longer it takes to compute. A respectable RSA implementation will use 512 bits, which gives extremely high security while remaining a reasonable task for a computer to calculate. Cracking RSA involves finding the decryption key from the encryption key by factorising huge integer numbers. For 512-bit RSA keys, this would take tens of thousands of years even on a supercomputer.

The DES algorithm is described in the Federal Information Processing Standards Publication Number 46 of 15th January 1977, published by the U.S. Department of Commerce / National Bureau of Standards, and in ISO Standard 8731 (Part 1), dated 1st June 1987.

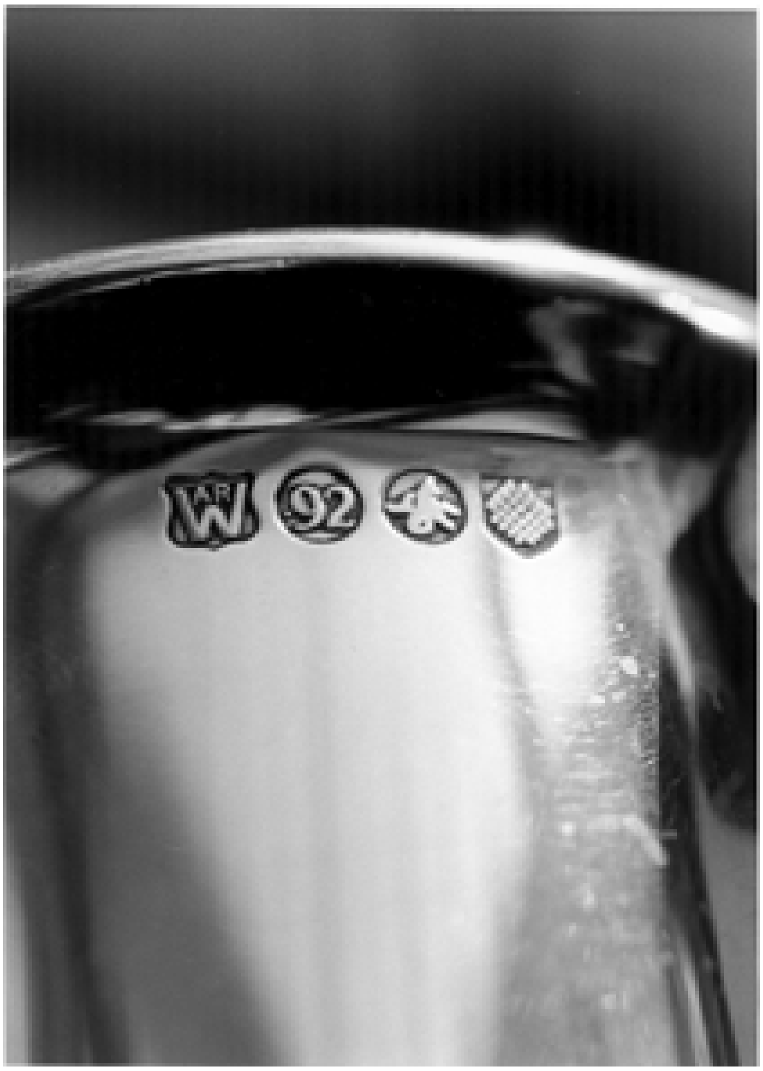
Encryption algorithms: SPA and MDH

Sophos has developed two proprietary encryption algorithms, **SPA** and **MDH**, which can be used as alternatives to DES and RSA respectively.

SPA, like DES, is a symmetric algorithm for encrypting and decrypting 64-bit data blocks. Unlike DES, however, it uses a 64-bit key rather than a 56-bit key. There is no known method of cracking SPA, other than exhaustive key searching. Due to the greater key length, an exhaustive search attack on SPA would take 256 times as many encryption operations as an equivalent attack on DES. Unlike DES, SPA was designed specifically for efficient operation in software. As a result, software SPA implementations are 5 to 10 times as fast as the most optimised software DES implementations, and can approach the speed of hardware DES.

MDH is a highly secure asymmetric algorithm which can be used as an alternative to RSA. Like RSA, it is an arithmetically based block cipher which can be implemented using any number of bits. For most applications, 512-bit MDH is recommended. MDH offers significant advantages over RSA in terms of security: because it is inherently immune to factorisation attacks; and because MDH allows full use of the available key space, whereas for an equivalent implementation, only a much smaller set of values can be used as RSA keys. The MDH algorithm is a standard option in Sophos' product PUBLIC.

SPA and **MDH** were developed in the UK as part of Sophos' cryptographic R&D programme. Both algorithms can be customised for special applications or closed user groups.



AR
W
92

Authentication

What is authentication?

Authentication is the technique used between the sender and receiver of a message to validate its source and its contents. Authentication can also be used for establishing the validity of stored data.

The need for authentication

The need for authentication is clear in banking applications. Suppose that XYZ Bank receives a message from ABC Bank, requesting the transfer of money from one account to another. XYZ Bank has to establish the origin of the message and also that the amounts and account numbers are correct. It could, for example, telephone ABC bank, speak to the sender and verify all the details. However, a better way of authenticating a message would be to append a unique code which only the sender at ABC Bank could have calculated. Such a code is known as a Message Authentication Code (MAC).

A MAC is usually a 32-bit (4 byte) value appended to the message. The uniqueness of the MAC results from the one-way function used to calculate it.

ISO standards 8730 and 8731

ISO (International Organisation for Standardization) standard 8730-1986 (Banking - Requirements for message authentication (wholesale)) specifies methods to be used for protecting the authenticity of wholesale messages passing between financial institutions, by means of a Message Authentication Code.

ISO standard 8731-1986 (Banking - Approved algorithm for message authentication) specifies **two** approved authentication algorithms, which meet the requirements specified in ISO 8730.

- ISO 8731/1 describes the Data Encryption Algorithm (DEA). This is identical to the Data Encryption Standard (DES).
- ISO 8731/2 describes the Message Authenticator Algorithm (MAA), specifically designed for high speed authentication using software. The ISO 8731/2 algorithm produces a 32-bit (4 byte) MAC.

Both the DEA (ISO 8731/1) and the MAA (ISO 8731/2) are used in Sophos products.

ANSI standard X9.9

ANSI (American National Standards Institute) standard X9.9-1986 (Financial Institution Message Authentication (Wholesale)) specifies methods for authenticating messages.

The algorithm for authenticating messages is the Data Encryption Algorithm (DEA) and X9.9 specifies the way a MAC is constructed using the DEA. The modes of operation of DEA are described in ANSI X3.106-1983 (Modes of Operation for the Data Encryption Algorithm), while the algorithm itself is described in ANSI X3.92-1981 (Data Encryption Algorithm). Note that the DEA is identical to DES.

Sophos' product VACCINE uses the ANSI X9.9 standard, in conjunction with SPA, as its normal method of constructing MACs for stored data.

Digital signatures

A **digital signature** is a way of authenticating a message using a public key cipher. Like the authentication methods described in ISO 8730/8731 and ANSI X9.9, it uses a MAC calculated from the contents of the message. However, unlike these two methods, the uniqueness of a digital signature **does not depend on the secrecy of a mutually agreed key**. Digital signatures are a further step towards secure message authentication, beyond the MACs of ISO 8730/8731 and ANSI X9.9.

A message bearing a digital signature cannot be altered in any way without the signature becoming invalid. The signature is also invalid if the message was actually sent by anyone other than the claimed sender.

This means that disputes over the contents of a message can be avoided, and that forgeries, tampering or transmission errors are **always** detected. For example, the sender of a message cannot claim that they did not actually send the message, if it can be shown that the message bears a valid digital signature. Also, the recipient of the message cannot alter its contents (e.g. the sum on a payment order) and still claim that it is genuine.

Digital signatures are typically constructed and checked using RSA or MDH combined with DES or SPA.

A message can be signed digitally **without** being encrypted. Often the first concern is to guarantee the precise contents and origin of a message rather than maintain confidentiality.

Secure erasure

What is secure erasure?

Secure erasure is the complete removal of file contents from magnetic media such as disks.

Just as confidential documents should be shredded or burned (rather than simply thrown away) confidential files on disk should be positively removed and overwritten (rather than simply deleted).

Insecure file deletion is a widespread risk to confidential data.

Why file deletion can be insecure

Every computer system has a facility for deleting unwanted files, usually using a command such as 'DEL', 'RM' or 'ERA'. On most systems, these commands merely remove the file's name from the directory, but leave the actual contents of the file untouched on disk.

Furthermore, multi-tasking systems and even normal PC packages use temporary storage areas on disk, which can be abandoned when the programs finish executing, leaving confidential data on the disk without informing the user. While not being easily accessible to the normal user with normal system commands, the data can be retrieved and examined with little difficulty.

Secure erasure programs

Since file deletion is insecure, confidential files should be erased positively with special shredding programs.

All Sophos packages automatically shred the unwanted files which may have contained confidential information. In addition, the Sophos utilities SHRED and PURGE can be used as required for completely secure and irretrievable deletion of any unwanted data.

Emerging erasure standards

In certain countries, there are already official or semi-official standards for secure erasure of magnetic media.

Laboratory studies have shown that under certain circumstances it is possible to read back information which has only been overwritten once, due to physical effects such as residual magnetism. For very confidential information, therefore, the preferred approach is to ensure that each storage bit is changed in polarity a number of times before finally being overwritten with harmless data.

Sophos has introduced three erasure security levels:

1. 'Quick' security level (1 overwrite); overwrite with a pattern.
2. 'Government' security level (3 overwrites followed by a verify); overwrite with all 1s, overwrite with all 0s, overwrite with a pattern, verify the pattern.
3. 'Military' security level (7 overwrites followed by a verify); overwrite with all 1s, overwrite with all 0s, overwrite with all 1s, overwrite with all 0s, overwrite with all 1s, overwrite with all 0s, overwrite with a pattern, verify the pattern.

These levels cover the full range of official erasure requirements. In many cases, a single overwrite is adequate. For greater protection, the ‘Government’ security level guarantees that each storage bit changes polarity at least once, while the ‘Military’ level guarantees at least five polarity changes for every bit.

All Sophos products support these three security levels where appropriate. The final overwrite patterns allow instant visual verification that the disk contains no confidential information.

Note: There are no formal standards for secure erasure of magnetic media in the UK.

Access control

What is access control?

Access control is the prevention of unauthorised access of computer resources.

Physical access control refers to a lock on the door of the computer room or on the computer itself.

Logical access control refers to the separation of the users' processes and data, as well as the separation of the users and the operating system.

On mainframes, which have traditionally had a number of users, logical access control has evolved as a part of the operating system.

On PCs, it has been an afterthought. However, new operating systems such as Windows NT provide built-in separation of users as well as the definition of user privileges on an individual user or group basis.

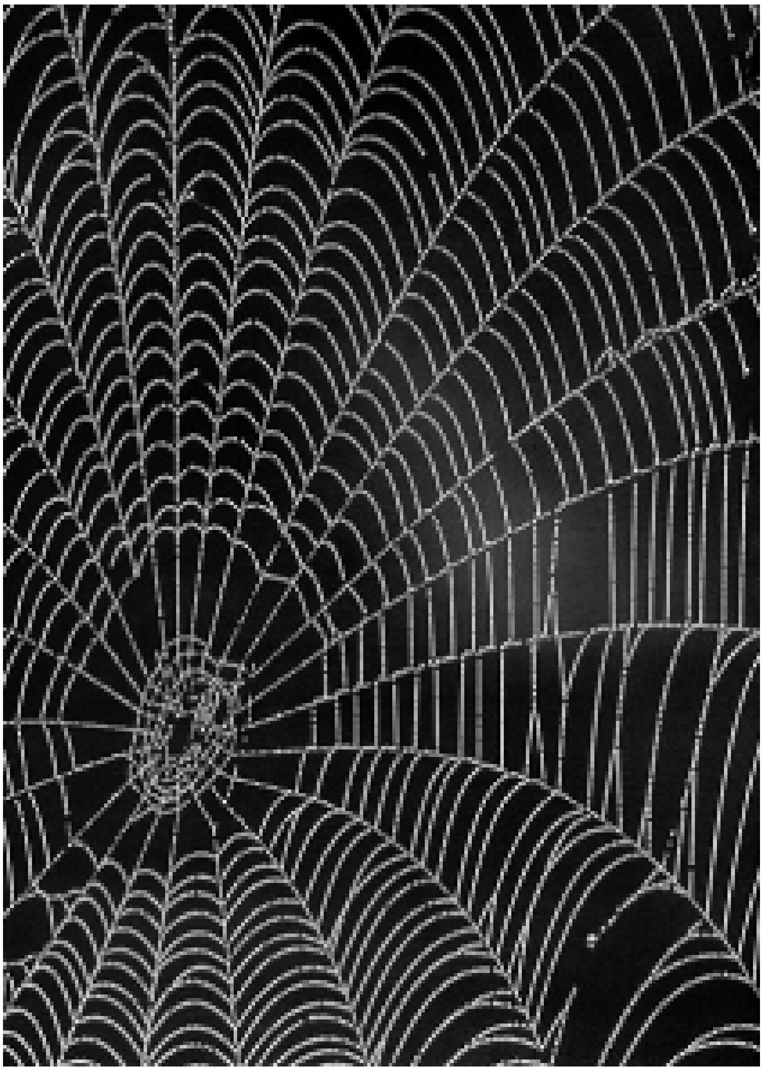
Requirements of a PC access control system

- The system should prevent booting from floppy disks. If it does not, an unauthorised user could use the system from a floppy disk, completely bypassing security mechanisms.
- Each user should be identified by a combination of an ID and a password. Passwords should not be echoed (displayed on the screen) when typed. The system should enforce a minimum number of

characters in a password (say 6 or 8). The passwords should have expiry dates, and the user should be forced to change a password on expiry. The system should not allow the re-use of the same or a previous password.

- The clock providing time and date should be separate from the DOS clock. If this is not the case, one cannot rely on packages which allow access at certain times of day because anybody can change the DOS clock setting. The provision of a separate clock is, of course, only possible in hardware.
- A user should be assigned specific times of day and days of the week during which they are allowed to log on.
- A user should have a set of privileges defined by a security manager, such as which directories they are allowed to access, which programs to run etc.
- An audit trail should be kept of which users have logged on and when. This should preferably be stored in an encrypted form and in such a way that it cannot be erased.
- The user should be able to leave the PC in a locked state, which can only be unlocked by using a correct password. The screen should be blanked when the PC is locked. This facility allows the user to leave their desk without fear that somebody will tamper with their PC.
- The user should be able to blank the screen by touching a single key. This prevents onlookers from seeing any confidential information on the screen.
- A keyboard inactivity monitor should be provided, locking out the PC if nothing has been typed for a predetermined period of time. Even if the PC is left unattended and unlocked, it should lock itself automatically.

- There should be a set of routines for the security manager to create new accounts, change privileges etc. Specific user passwords should be exclusively controlled by the user.
- Automatic encryption of sensitive files should be provided. Ideally, this would be using a recognized encryption algorithm such as DES. Faster algorithms are often not secure, and should be verified by an independent *bona fide* organisation.
- It is important that the package does not interfere with hardware and software on the PC, which is a potential problem with access control packages which reach ‘deep’ into the operating system.



Security on the Internet

The Internet is a network of networks connecting many millions of computers and allowing data to be transferred between them. As such, it can be an unsafe place.

Internet security problems

There are security weaknesses in all manner of user and kernel-level software on all operating systems. This section documents the most obvious, the most well-known, and those most likely to be the subject of attempted exploits.

Password guessing

Passwords and pass-phrases are by far the most common way for a legitimate user to prove to a computer that they are who they claim to be. The importance of the security of the passwords should never be underestimated; if they are compromised, the only solution is to reset them all.

The passwords on most modern computer systems are stored in the form of a **one-way hash**. When a password is set, a number is computed using the string entered as the key. Future attempts to enter the password are hashed, and compared with the stored value. It is not possible (computationally impossibly expensive) to compute the string from the hash value, i.e. it is a one-way system. The most popular form of password-cracking, therefore, is **password guessing**.

Password guessing involves the attacker obtaining a list of the encrypted passwords for a machine. These are often stored in common locations, e.g. in the file etc/passwd on UNIX machines. They can examine this list at their leisure, only contacting the computers again when they believe they have access.

Software to test encrypted strings against the stored password hash is easily available on the Internet. A favourite technique is the **dictionary attack**, which uses words in a dictionary, because research suggests that between 20% and 40% of passwords are English words. Such software also tests various encodings of the account name and the real name of the account holder. These techniques often display an alarmingly high hit rate.

Administrators should run regular attempts to crack passwords on their own systems, and force all those which are cracked to be changed. If the administrator can crack a password, others can too.

Rules for choosing a secure password are well-known, but often ignored. See the section on 'Choosing keys and passwords' in the 'Data Encryption' chapter.

The most secure system is one using so-called 'one-time' passwords, i.e. passwords that change every time they are used. Not only are these practically immune to guessing, they are certainly immune to 'password sniffing' (see below).

Password sniffing

Even the most securely-chosen password is useless if it is transmitted across an insecure network in a manner that can easily be intercepted and read. For example, many users avail themselves of the Internet's support for the ever-present telnet protocol to log in to remote computers. When they do this, their password is sent across the Internet unencrypted. Any attacker who has access to the network en route can 'sniff' this information from the network and hence obtain the password.

It can be extremely difficult to survive on the Internet without transmitting unencrypted passwords. However, the careful user must resist, and must be extremely careful not to be trapped into giving their password to a remote machine.

In summary, **never allow passwords to be transmitted across an insecure network in plaintext**. This encompasses a wide variety of protocols, including telnet, FTP, and the rlogin family.

Address spoofing

Much Internet-related software makes security decisions based upon the source address (i.e. the sender) of information reaching it. It is possible (although by no means trivial) to fake this information in packet headers and this will fool such software.

Fortunately, it is possible to configure routers and firewalls to eliminate this type of attack. Informing the router that any packet reaching it from the outside with a network address of a machine on the inside should be discarded, and vice-versa, removes the problems.

Domain inserting

Another difficult, but perfectly possible, attack involves subversion of routing protocols and/or direct attacks on name servers.

Attacks have been demonstrated where the network address of a registered domain (for example 'sophos.com') is changed. In this way, the attacker's machines will receive all information which was intended for the victim's computers.

A related risk arises from security problems in various versions of the BIND (Berkeley Internet Name Domain) package. This package, from which comes the version of 'named' (the software that handles requests to resolve machine names to IP addresses) used by almost all UNIX machines on the Internet, is often updated, and administrators should keep abreast of the latest developments.

Connection hijacking

Attacks have been seen which involve taking over an existing connection between two computers. Specifically, information can be inserted into the TCP stream as the data travels between the two computers.

This attack is an enhancement on address spoofing, allowing data to be modified or added to in transit. The solution is to use encrypted data streams, so that the attacker cannot insert data without first breaking the encryption.

Rlogin

The Berkeley 'r' protocols (rlogin, rcp, rsh, etc.) allow users to define a list of machines from which connections will be accepted **without password verification**. This is extremely dangerous, as a hacker only has to break one machine, and they have automatic access to the rest.

Electronic mail

Electronic mail can give rise to a number of security problems.

The most famous exploitation of bugs in mail delivery systems was the 1988 *Internet Worm*. This program exploited well-known bugs extant in the mail servers of the time, which have since been closed. However, new exploits are regularly discovered in the UNIX mail delivery system *sendmail*, and mailing list management software has also had its fair share of weaknesses.

The administrator should disable all the mail features that they do not need, and limit those which they do to the bare minimum.

Email forgery

Email forgery is, in some ways, more serious. Any first year computer science student soon finds out how to send mail with the sender's address faked, as this capability is a feature of the mail protocol.

Spam

Spam is now a serious problem. Spam is the name given to a message delivered to a large number of recipients and/or newsgroups, usually advertising primitive, and often illegal, pyramid schemes, pornographic Web sites, or any of a large variety of products and services. Many spam messages are sent out with fake return information, usually a non-existent address. It is usually possible to work out which computer a message came from by examining the headers on the message in some detail. However, the relevant headers are usually hidden from the user by modern PC mail systems.

Email and viruses

Email can also facilitate the transmission of computer viruses (see the 'Viral infiltration routes and methods' section of the 'How infections spread' chapter).

NFS, NIS, and RPC

All three of these were created by Sun Microsystems. NFS stands for **Network File System**, NIS for **Network Information Service**, and RPC for **Remote Procedure Call**. All three have been implicated in various security breaches, but fortunately there is usually no need to allow traffic of any of these protocols into the network from the outside: simply block them at the router. It is frequently necessary to run these systems internally (particularly NFS, which is now the standard method for UNIX machines to share file systems with one another).

X Window System

The X Window System is the universal graphical front end often seen on machines running UNIX. Many of its network aware features have proved very insecure. Once a hacker has used vulnerabilities within the X Window System to gain access, they can also use X features to watch the screen of the machine under attack, and grab and replace keystrokes.

As is well-documented, X servers listen on TCP ports starting at 6000, and incrementing with the number of screens and sessions: as a minimum, it is wise to block ports 6000 to 6005.

Ping of death

'Ping' is a very useful utility that tests the basic network connectivity between two computers, i.e. whether one computer can 'see' another one across the network. However, in 1996 it was found that ping packets much larger than normal, but still within the size allowed by the specification, would cause many types of UNIX to crash, halt, reboot, or run very slowly for several minutes (hence the name 'ping of death').

Machines running Windows 95 or Windows NT can generate these packets without difficulty, and many versions of programs to allow UNIX machines to generate them are now in circulation.

Countless machines are still vulnerable to the ping of death. As a denial of service attack, it has few equals.

Web browser risks

The use of the Word Wide Web harbours its own set of problems.

For example, many modern Web browsers are configured to launch the appropriate application when a certain type of file is delivered. Indeed, Internet Explorer can use the components of Microsoft Office almost invisibly, so that when a Word document is downloaded, Word starts up to view the file inside the Internet Explorer window. This means that macro viruses have their chance to execute and seize control, infecting the machine.

Web browsers also make downloading executable programs from remote machines a trivial exercise. This brings with it the risk of infection by viruses that attach themselves to executables. Users should be made aware of the inherent risks of downloading.

Recent developments that affect security include the use of Java and ActiveX 'applets'. These are often automatically downloaded by Web browsers and executed; they are popular as they make the Web a much more interactive experience, as well as making it possible to make pages look more dynamic.

Java

Of the two, Java is the more favoured; it is cross-platform, and applets downloaded from the Internet are run in a controlled environment (sometimes likened to a cage), in which it is, in theory, impossible to perform malicious actions.

Of course, security breaches in this 'virtual machine' have been discovered, and later fixed. Problems exist which cannot be fixed by simple bug-hunting. It is still possible, for example, for an applet to open more and more windows until the computer runs out of resources, or to allocate more and more memory to the same end.

ActiveX

ActiveX, the Microsoft technology for interactive Web development, is entirely different. It has many disadvantages compared with Java: for example, the fact that an ActiveX applet consists of Windows 32 code, and is almost always for Intel processors, which means that it is not cross-platform. Most significantly from the security angle, ActiveX applets are allowed full access to the resources of the local computer. The sole concession to security is the presence of digital signatures, which prove that an applet is authentic and unmodified, and the notion of trusted sources of ActiveX applets.

Web server risks

The risks inherent in Web servers are all too clear. People not only connect to them and retrieve information (in the form of Web pages), but also run programs on them, in the form of CGI (Common Gateway Interface) programs. These programs handle image maps, search engines, forms, etc. Indeed, many Web sites are run purely from CGI scripts, so that there is no static HTML whatsoever. Such CGI scripts generate HTML output which is sent directly to the client’s browser.

CGI scripts, particularly those taking input from forms, are a major risk. If they are improperly implemented, it is often possible for attackers to execute arbitrary commands on the Web server as if they were a highly privileged user such as root.

The phf program

One specific risk to Web servers is worth mentioning. Older versions of the NCSA (National Centre for Supercomputing Applications) HTTP (HyperText Transport Protocol) daemon for UNIX installed a CGI program called phf. This program contains a well-known security hole allowing arbitrary commands to be executed on the server as discussed above. This flaw is extremely well-known, and attacks occur regularly (every couple of months on a sample site). If phf exists on a Web server, it is advisable to delete it.

Confidential information

A more general problem is the fact that Web servers often hold information which should only be given to those authorised to see it, for example, account information on the sites of on-line shops, package tracking information on courier companies' sites, and software license keys on shareware sites. It is crucial that this information remains both unreadable and unchangeable by all those who are not authorised.

Denial of service attacks

A common form of attack against Web servers is the so-called 'denial of service' attack. This aims to reduce the efficiency of the Web server or crash it completely. A well-known example is the 'ping of death' (see above). Another attack involved crashing the Microsoft Internet Information Server (IIS) version 3.0 by submitting a URL of a certain length (as is common with such problems, a patch was issued fairly quickly). The significance of such attacks obviously depends on how critical the Web server is to the subject of the attack.

Denial of service attacks are gaining in popularity, both because of their dramatic effects and the fact that programs which perform such attacks are in widespread circulation. Perhaps the best-known is 'winnuke' and its derivatives, which attempt various attacks designed to bring down Windows 95 and Windows NT systems.

It is impossible to prevent denial of service attacks being carried out. However, it is possible for users to reduce the risk that an attack will reach their machines, or that an attack will succeed if it does reach their machines.

Windows NT risks

Windows NT is growing in popularity as a system for connecting to the Internet. As such, it has both advantages and disadvantages compared with UNIX.

On the plus side, many of the bugs referred to above exist in UNIX operating systems and software, and thus will not be a problem on Windows NT.

On the other hand, much of Windows NT and its associated software is the subject of a steady flow of new bug reports. At the time of writing there are at least three ways to crash a Windows NT server using perfectly legal network data.

On 4 March 1998, a co-ordinated attack by computer hackers crashed Windows NT machines throughout the US, including machines used by the military, the space agency and universities. No data was lost, but the attack illustrated the vulnerability of Windows NT machines.

Microsoft’s Internet Information Server

Microsoft’s Internet Information Server has also been the subject of several hacks.

One version makes the source code for dynamically generated pages trivially available to any Web browser, and another version crashes when presented with a URL of a certain length. All these bugs have been the subject of patches (which Microsoft call ‘hotfixes’), and have later been fixed in the next service pack, but there are doubtless many machines on which some of these simple denial of service attacks will still work.

How to minimise the security risks

Administrators can take measures to protect their networks from the security risks posed by Internet connections.

However, there is a choice between simplicity of use and security. Either users can be connected to the Internet in such a way that they find everything easy to use, can move files easily, browse the Web from their desktop, and talk directly to all the available Internet services, or they can have a secure system. Any form of effective Internet-related security is bound to make some tasks harder to perform.

Separate networks

One simple way to minimise the risks to corporate computer systems is to maintain entirely separate networks. The only machines connected to a network, which in turn is connected to the Internet, are those which are running Internet services. These machines hold only the minimum of data required to perform their tasks, and absolutely nothing confidential.

Once computers which are not running Internet-accessible services (mail, WWW, and FTP, for example) are connected to the system, everything becomes that much more complicated.

Having said that, any permanently active Internet connection will at the very least involve a **router**, and will very probably involve some form of **firewall** (see below).

Routers and firewalls

At the simplest level, a router is a device which manages the flow of **packets** (chunks of information traversing a network). For example, the router can be told that a given list of machines are on one side of the router, usually referred to as the ‘inside’, and all others are on the other side, the ‘outside’. This information allows it to pass packets to the appropriate network (so-called ‘routing’).

Modern routers, however, allow a considerable degree of control at a much lower level than outlined above. For example, it is possible to:

- Allow and deny certain types of packets to certain ports on certain machines.
- Prevent certain types of more general attack, including network address spoofing (see above).

This functionality subsumes much of that earlier ascribed to firewalls. However, modern firewalls incorporate higher-level functionality, for example virus scanning, and are often easier to configure than routers (many come with graphical configuration utilities).

Mailing lists and information sources

Anyone who manages or controls Internet-connected systems should, at the very least, read the CERT (Computer Emergency Response Team) and BugTraq mailing lists. The latest addresses for these lists can be found by searching the World Wide Web for the list names.

There are many more security-related mailing lists from which the discerning administrator can pick and choose, related to almost any security issue, although the quality of the lists varies considerably.

For a more general view of unforeseen problems with computers, Peter Neumann’s RISKS list is recommended.

Thousands of other sources of information on Internet security are available on the Internet itself.

The computer ‘underground’ can also be a useful source of information, as many bugs and the associated exploits circulate for weeks or months before being documented by mainstream security experts.

Conclusions

Any computer connected to the Internet should be considered insecure. Even if the connection is non-permanent (e.g. a dial-up), the machine is still at risk, particularly if it dials for long periods of time.

Remember that most operating systems and software installations enable many more features by default than most sites actually need. Administrators should:

- Always use the minimum operating system and software configuration that will do the job.
- Only make available to outsiders those services which they are allowed to access, and do so through careful router and firewall configuration.
- Try to minimise exposure to denial of service attacks.

The millennium bug

What is the millennium bug?

The millennium bug is a problem that makes some programs fail to recognise the year 2000 as the year after 1999, giving rise to unforeseen or incorrect behaviour. It is also sometimes called the ‘Y2K’ problem, or ‘Year 2000 problem’.

It is **not** a computer virus.

The causes of the bug

In many early computer programs, the year was represented by its last two digits, e.g. 84 for 1984, and the first two digits were assumed to be 19. In some cases, these programs are still in use, or have been incorporated into more recent programs as ‘legacy applications’. For these programs, the year 2000 will become 00, and may be misinterpreted as 1900. Such programs may then miscalculate periods of time spanning the millennium year.

The leap year bug

The year 2000 poses another problem for computer programs: the leap year bug. **The year 2000 is a leap year**, as are all century years divisible by 400. However, some software treats it as a non-leap year and will skip 29 February 2000, so that calculations across this date will be wrong by one day.

The implications of the bug

The millennium bug could cause:

- Failure of computer systems that depend on time-keeping for their operation.
- Incorrect calculation of time periods spanning the 1 January 2000 or the 29 February 2000.

There is particular concern about control systems (e.g. in air traffic control) and electronic financial transactions.

Solving the problem

There are three steps involved in solving the problem:

- Establishing where dates are used in the software.
- Repairing software that is affected by the bug. This involves rewriting the code that controls the date, or modifying the functioning of the program.
- Ensuring that any other systems the software interacts with are not affected by the bug.

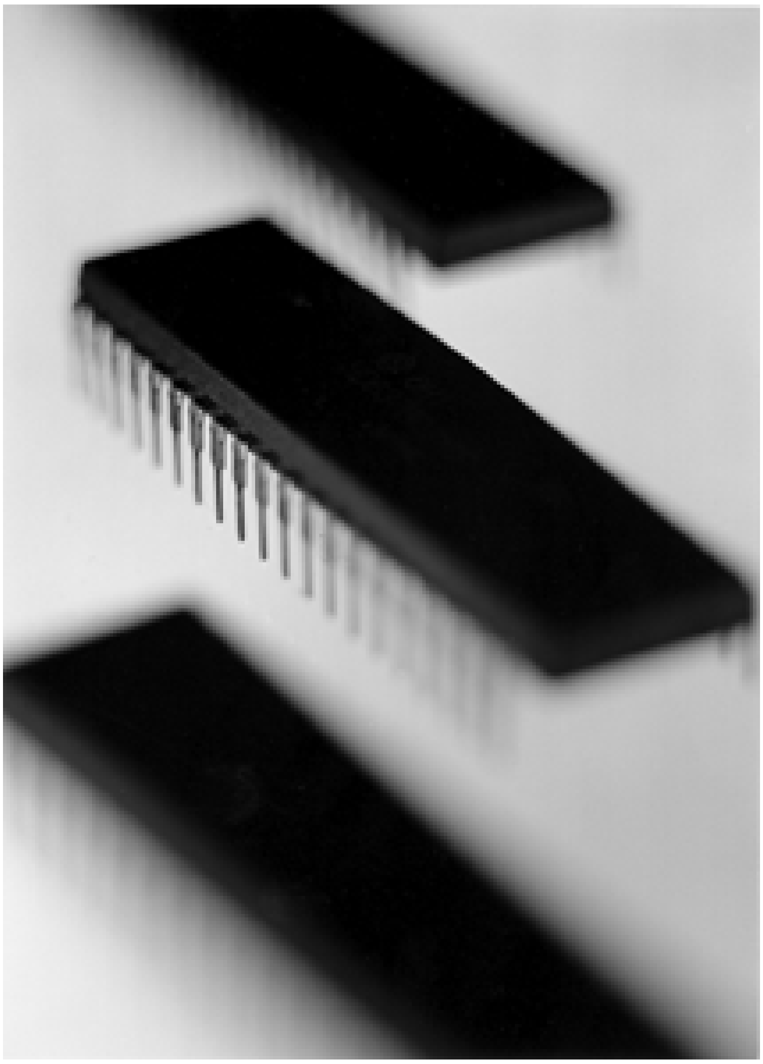
Millennium compliance

Most companies are now ensuring that their systems are 'millennium compliant', i.e. that they will recognise the date change correctly on 1 January 2000. Many also insist that their suppliers should be millennium compliant.

Companies that fail to take action face various sanctions. Auditors may fail to approve company accounts; shareholders may lose confidence; and directors may be liable for losses caused by the bug.

Any organisation that wants to protect its data and its prospects should act now.

Computer viruses 2



Introduction to computer viruses

What is a computer virus?

A computer virus is a special kind of computer program which:

- Spreads across disks and networks by making copies of itself, usually surreptitiously.
- Can produce undesired side-effects in computers in which it is active.

How infection occurs

In order to infect a computer, a virus has to have the chance to execute its code.

Viruses usually ensure that this happens by behaving like a parasite, i.e. by modifying another item so that the virus code is executed when the legitimate item is run or opened.

Good vehicles for viruses include the parts of a disk which contain code executed whenever that disk is booted, and documents which contain macros executed whenever that document is opened with the relevant application.

As long as the virus is active on the computer, it can copy itself to other files or disks that are accessed.

For more information on different types of viruses and their behaviour, see the 'Viruses and their effects' chapter.

How viruses escape detection

The successful spread of a virus depends on how long it can replicate unnoticed, before its presence is made known by the activation of side-effects. Viruses use two main methods of disguise:

- Encrypting (scrambling) their code to avoid recognition.
- Preventing applications from seeing the virus in memory, by interrupt interception or (in the case of macro viruses) by disabling the options to view macros.

For more information, see the 'How viruses conceal themselves' section of the 'Viruses and their effects' chapter.

Virus side-effects

As well as self-replicating code, a virus normally contains a 'payload'. The former is like the propulsion unit of a missile; the latter is like the warhead it delivers. The payload can be programmed to have malicious side-effects.

These effects can range from harmless messages to data corruption or destruction.

For more information, see the 'Virus side-effects' section of the 'Viruses and their effects' chapter.

How viruses spread

Infections spread from machine to machine, and from organisation to organisation, in a number of ways.

Viruses can be transmitted by:

- Booting a PC from an infected medium.
- Executing an infected program.
- Opening an infected file.

Common routes for virus infiltration include:

- Floppy disks or other media that users can exchange.
- Email attachments.
- Pirated software.
- Shareware.

For more information, see the ’How viruses spread’ chapter.

Anti-virus measures

The fight against computer viruses involves five kinds of counter-measure:

Preparation includes making backups of all software (including operating systems) and making a contingency plan.

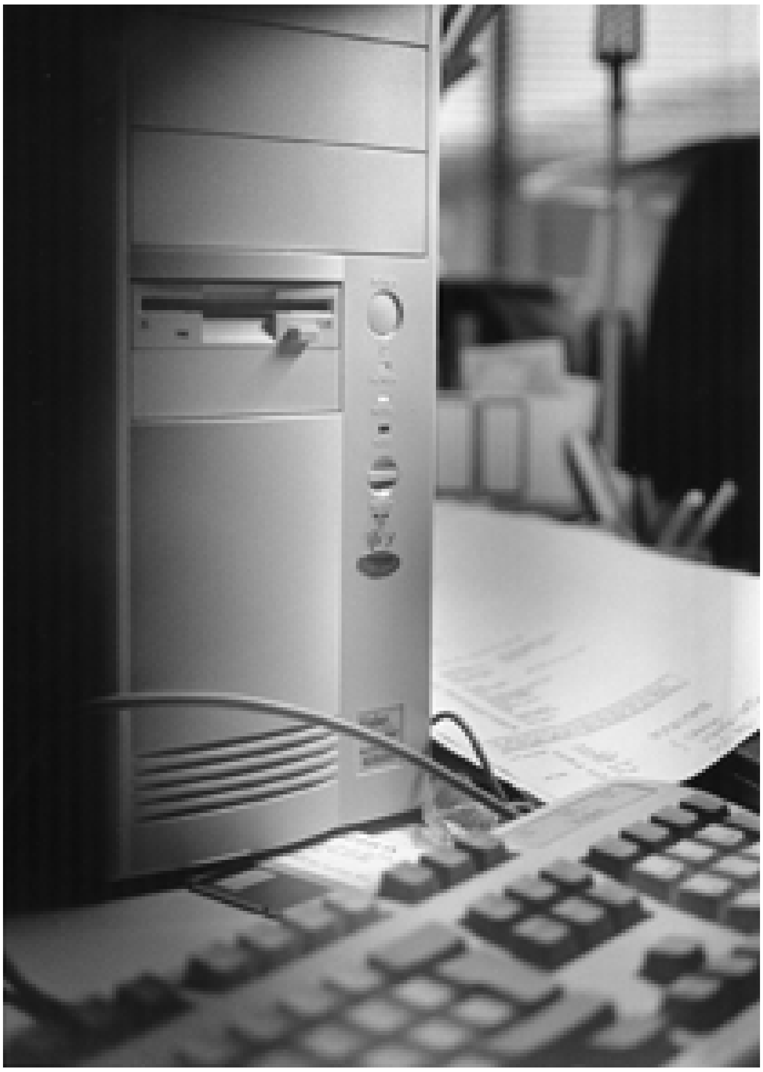
Prevention includes creating user awareness, implementing hygiene rules, using disk authorisation software, or providing isolated ‘quarantine’ PCs.

Detection involves the use of anti-virus software to detect, report and (sometimes) disinfect viruses.

Containment involves identifying and isolating the infected items.

Recovery involves disinfecting or removing infected items, and recovering or replacing corrupted data.

For more information, see the ‘Anti-virus measures‘ chapter.



Viruses and their effects

This chapter discusses:

- The different types of virus.
- How viruses behave after taking control.
- How viruses conceal themselves.
- Virus side-effects.

For information on the other forms of software attack sometimes associated with viruses, see the 'Trojan horses, logic bombs and worms' chapter.

Types of virus

Viruses can be divided into six categories:

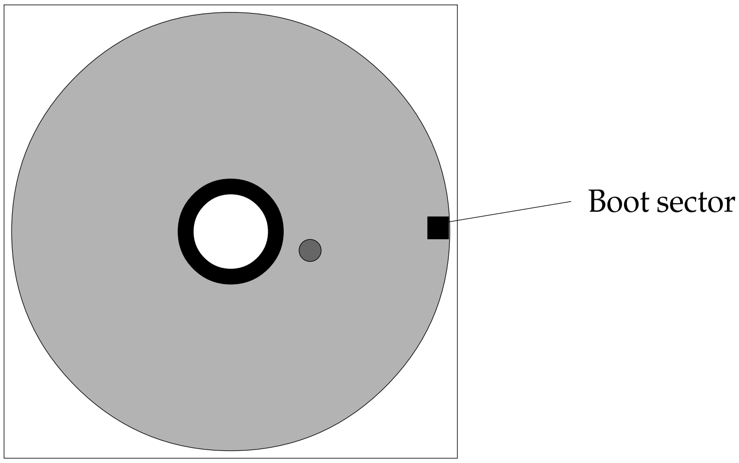
- **Boot sector viruses.**
- **Parasitic viruses.**
- **Multipartite viruses.**
- **Companion viruses.**
- **Link viruses.**
- **Macro viruses.**

Companion and link viruses could be regarded as special cases of parasitic viruses.

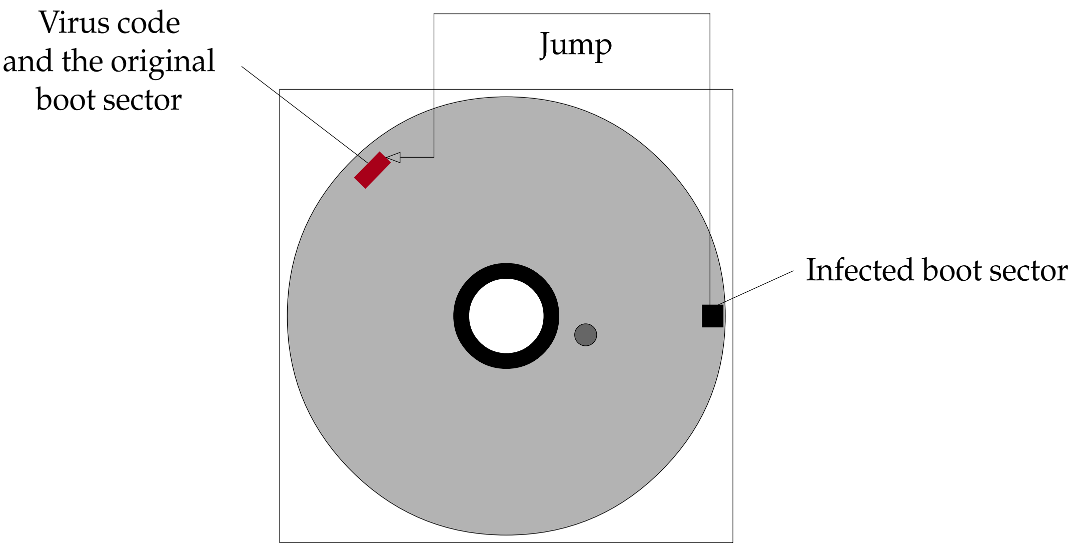
Boot sector viruses

Boot sector viruses modify the contents of either the master boot sector or the DOS boot sector, depending on the virus and type of disk, usually replacing the legitimate contents with their own version.

The original version of the boot sector is normally stored in an unused sector somewhere else on the disk, so that on booting up, the virus version will be executed first. This normally loads the remainder of



Uninfected disk



Disk infected with a boot sector virus

the virus code into memory, and then executes the original version of the boot sector. From then on, the virus generally remains memory-resident until the computer is switched off.

As all boot sector viruses modify the boot sector in some way, this is normally the only item one needs to examine for signs of infection. Some viruses such as *Starship* keep the amount of data modified to an absolute minimum (about 10 bytes).

The place where the rest of the virus code is stored is only of interest when trying to find the original boot sector in order to copy it back and 'disinfect' the disk. Some viruses, such as *Form,* use unused clusters in the disk's File Allocation Table (FAT) and label them as 'bad'. This prevents the operating system from allocating these clusters to files and possibly overwriting the virus code. Others such as *New Zealand* use part of the hard disk not normally used by the operating system (Sector 2, Head 0, Track 0 onwards). Another strategy is to decrease the size of the first partition on the hard disk (*Tequila*).

How boot sector viruses spread

Boot sector viruses are spread through **physical exchange of any media which can be used for booting up** (in most cases floppy disks). As a consequence, they spread comparatively slowly. However, the speed with which they spread can be greatly increased by 'droppers'. These are Trojan horse programs whose only function is to infect the boot sector of the PC and start the infection.

A PC becomes infected with a boot sector virus only if the user boots from an infected disk. It is completely safe to insert an infected disk into the drive and copy data from it (using the COPY command). The PC will not become infected unless it is booted while an infected disk is in drive A:.

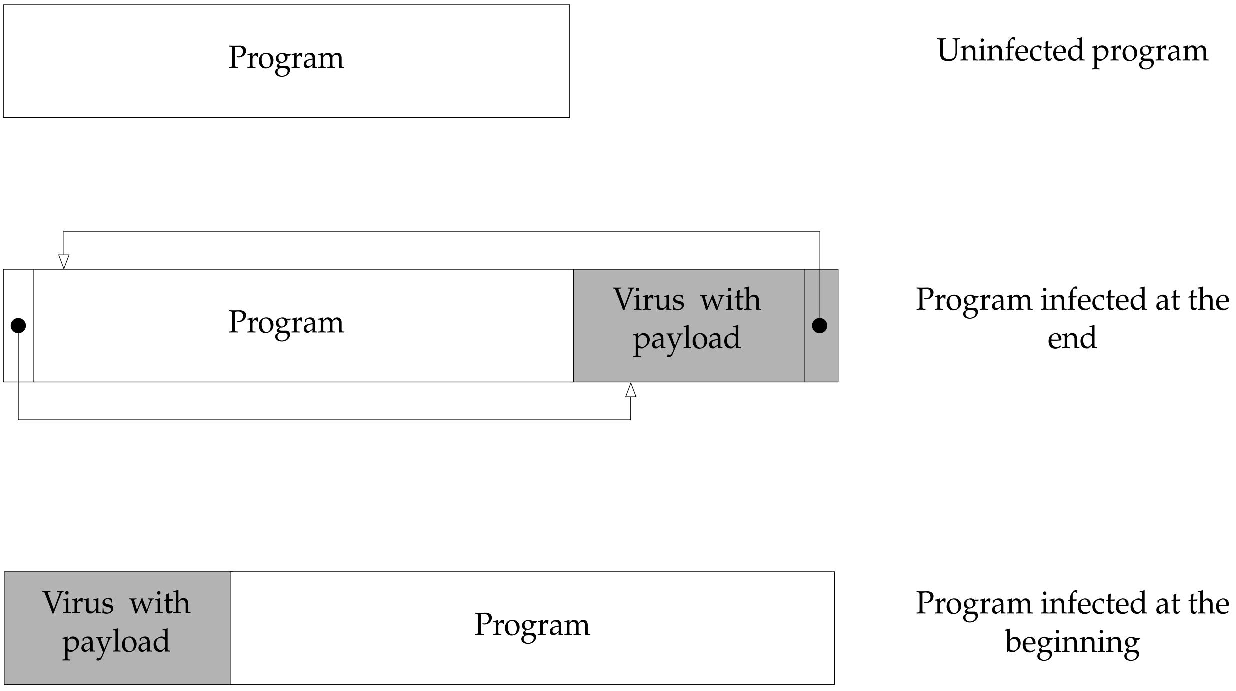
Parasitic viruses

Parasitic viruses attach themselves to COM and/or EXE files and divert the execution flow in such a way that the virus code executes first.

Depending on the virus, the virus code can be inserted at the beginning of the file or at the end, or both, or in the middle of the file.

Wherever it is inserted, the virus code must be executed before the infected host program. Thus, the virus runs at the same privilege level as the original program and once running, can do anything: replicate, install itself into memory, release its side-effects etc.

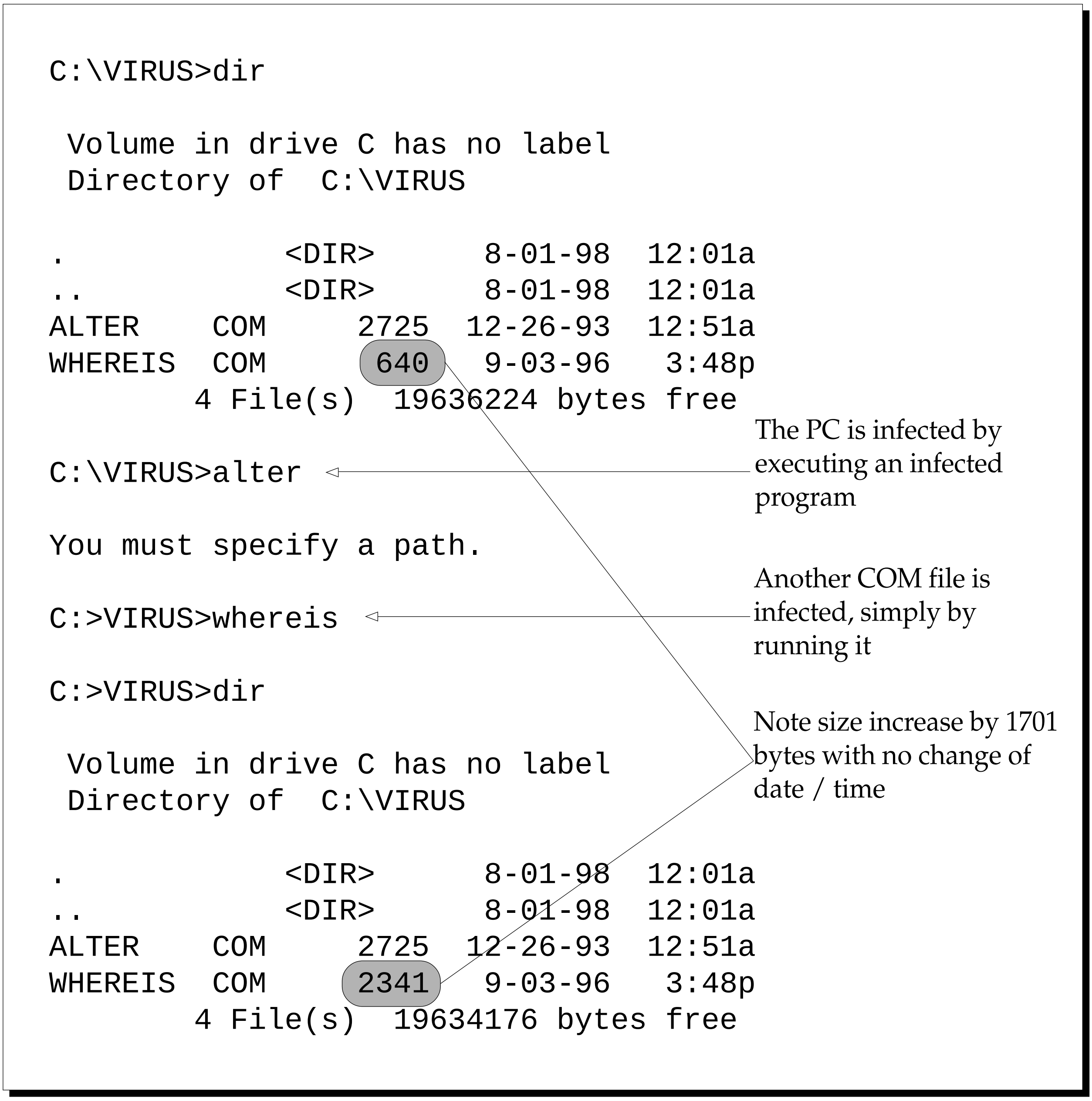
Once the virus code has executed, control passes to the original program which, in most cases, executes normally. The extra execution time is not usually perceptible to the user.



Program infection with a parasitic virus

How parasitic viruses spread

Parasitic viruses spread through **any medium which can be used for storage or transmission of executable code** such as floppy disks, tapes, networks etc. The infection will generally spread if an infected program is executed.



```
C:\VIRUS>dir

 Volume in drive C has no label
 Directory of  C:\VIRUS

.             <DIR>       8-01-98  12:01a
..            <DIR>       8-01-98  12:01a
ALTER    COM      2725  12-26-93  12:51a
WHEREIS  COM       640   9-03-96   3:48p
        4 File(s)  19636224 bytes free

C:\VIRUS>alter

You must specify a path.

C:>VIRUS>whereis

C:>VIRUS>dir

 Volume in drive C has no label
 Directory of  C:\VIRUS

.             <DIR>       8-01-98  12:01a
..            <DIR>       8-01-98  12:01a
ALTER    COM      2725  12-26-93  12:51a
WHEREIS  COM      2341   9-03-96   3:48p
        4 File(s)  19634176 bytes free
```

The PC is infected by executing an infected program

Another COM file is infected, simply by running it

Note size increase by 1701 bytes with no change of date / time

Cascade infecting a program

Most parasitic viruses, like *Cascade*, spread when another (uninfected) program is loaded and executed. Such a virus, being memory-resident, first inspects the program for infection already in place. If it is not infected, the virus will infect it. If it is already infected, further infection is not necessary (although some parasitic viruses like *Jerusalem* do re-infect *ad infinitum*).

'Fast infectors', such as *Nomenklatura*, infect whenever a file is opened. They can infect a large number of files on a system very quickly, especially if a scanner is used while the virus is memory-resident.

Parasitic viruses which are not memory-resident do not install themselves in memory, but spread by finding the first uninfected program on disk and infecting it. An example is the *Vienna* virus.

Multipartite viruses

Multipartite viruses exhibit the characteristics of both boot sector and parasitic viruses. They infect COM and/or EXE files (like parasitic viruses) and boot sectors (like boot sector viruses). Thus their chances of replication are increased.

How multipartite viruses spread

Multipartite viruses are spread through **physical exchange of any media which can be used for booting up** (in most cases floppy disks) as well as through **any medium which can be used for storage or transmission of executable code** such as disks, tapes and networks. The virus will become active if the PC is booted from an infected disk or if an infected program is executed.

Most multipartite viruses such as *Flip* are fully multipartite, which means that a PC infected by booting from an infected disk will infect other disks as well as executables, while a PC infected by executing an infected file will infect other executables as well as disks.

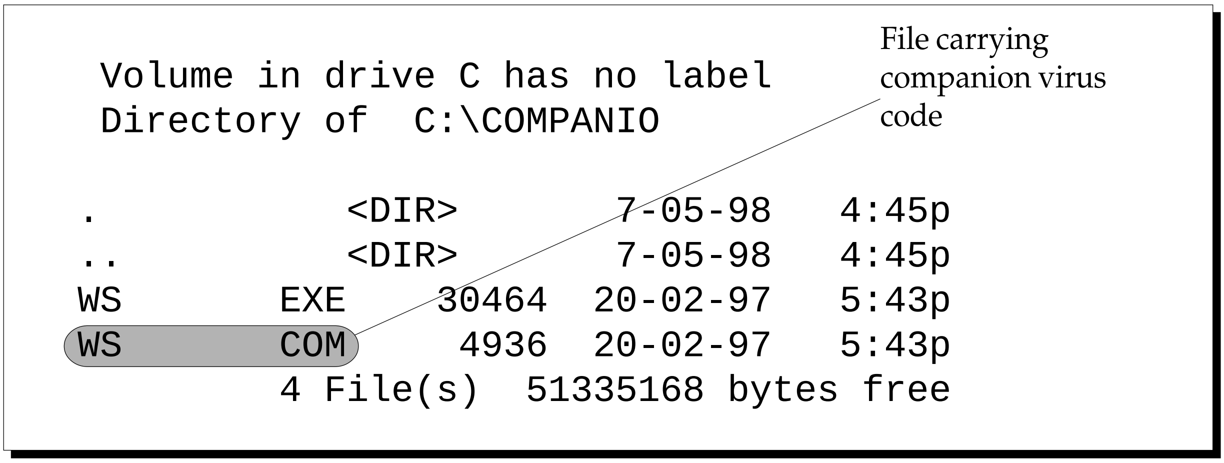
Some multipartite viruses are only partially multipartite; for example, a *Spanish Telecom* infection in a file will infect other EXE and COM files as well as boot sectors, while the same virus in a boot sector will only infect other boot sectors.

The speed of propagation of multipartite viruses is similar to that of parasitic viruses because they can be transmitted over the Internet and thus spread over great distances very quickly.

Companion viruses

Companion viruses exploit the property of the MS-DOS command line interpreter that if two programs with the same name (before the extension) exist in a directory, the operating system will execute a COM file in preference to an EXE file.

A companion virus creates a COM file with the same name as the EXE file it 'infects', storing its own virus code in the COM file. When a user types in the program name, the operating system executes the COM file, which executes the virus code and, in turn, loads and executes the EXE file. The virus makes no change at all to the contents of the 'infected' EXE file.



```
  Volume in drive C has no label
  Directory of  C:\COMPANIO

.          <DIR>        7-05-98   4:45p
..         <DIR>        7-05-98   4:45p
WS       EXE     30464  20-02-97   5:43p
WS       COM      4936  20-02-97   5:43p
         4 File(s)  51335168 bytes free
```

File carrying companion virus code

Companion virus infection

The directory listing below shows an unsophisticated companion virus which has infected WS.EXE by creating WS.COM. Companion viruses normally label the companion COM file with the DOS 'hidden' attribute, which means that they will not be shown in directory listings. This, however, is also the downfall of such viruses in the long run, because the DOS COPY command does not copy hidden files and the virus is thus denying itself the prime means of propagation: copying of executable files by users.

How companion viruses spread

Companion viruses are spread through **any medium which can be used for storage or transmission of executable code** (but see above comment on hidden files). The virus will become active if one of its COM programs is executed.

Companion viruses are unlikely to pose a major threat in the future.

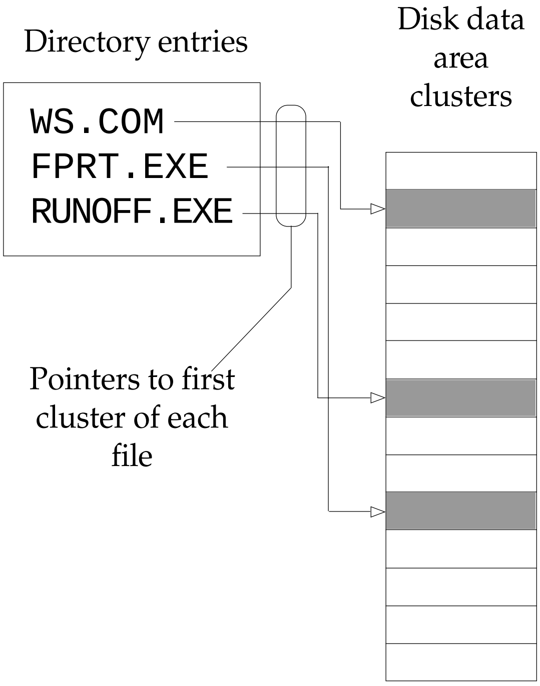
Link viruses

Link viruses work by linking the first cluster pointer of the directory entry of one or more executable files to a single cluster containing the virus code. The original number of the first cluster is saved in the unused part of the directory entry.

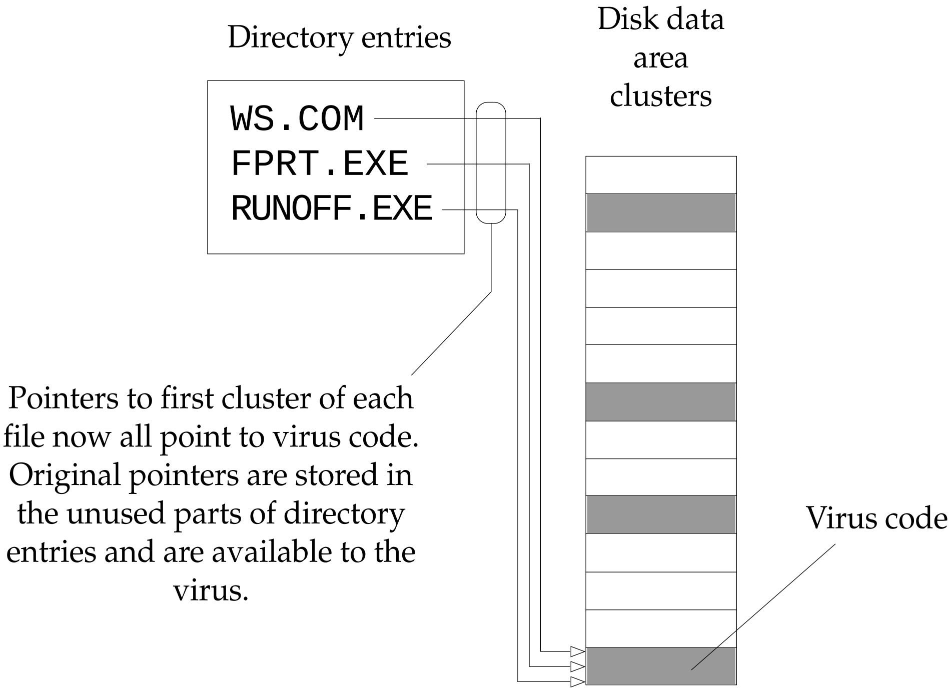
How link viruses spread

Link viruses are spread through **any medium which can be used for storage or transmission of executable code**. A PC will become infected if an infected program is executed.

Most networks cannot be infected by link viruses because their physical directory structure is different from DOS, but can act as carriers of the virus code.



Directory entries in an uninfected system


Directory entries
WS.COM
FPRT.EXE
RUNOFF.EXE
Disk data area clusters
Pointers to first cluster of each file now all point to virus code. Original pointers are stored in the unused parts of directory entries and are available to the virus.
Virus code

Directory entries in a system infected with a link virus

Macro viruses

Macro viruses use macros (instructions in data files that carry out program commands automatically) to become active and to infect other documents. They do this as follows:

1. The user accesses an infected document, which contains a viral macro or macros.
2. Opening the document (or, less usually, closing it, or invoking another command or shortcut) causes the viral macro to be executed automatically.
3. This viral macro usually makes transparent changes to the global macro environment.
4. Malicious macros placed in the global macro environment can launch side-effects and/or copy the virus to documents that are accessed thereafter.

For example, *Winword/Concept* is actuated by opening an infected document. The virus then infects the Word environment by copying the viral macros into the global macro environment. Once the virus is resident in this way, the use of the File/Save As menu option triggers a macro which places a copy of *Winword/Concept* into the document being saved.

How macro viruses spread

Macro viruses can be spread by **any document capable of containing macros**.

An infected document can place malicious macros in the global macro environment, allowing the virus to be copied to documents that are opened thereafter (see step 4 above).

Some macro viruses, e.g. *Winword/Snicker*, can directly infect other documents without installing themselves in the global environment. The places where this kind of virus most often looks for files to infect are the current directory or the Most Recently Used list.

Macro viruses have spread rapidly because:

- Unlike conventional viruses, they can be written relatively easily with little specialist knowledge.
- They can achieve a degree of platform independence: for example, Word macro viruses will typically work on all operating systems for which the Word environment is available, including Windows 95, Windows NT and Macintosh.
- People frequently exchange documents, especially over the Internet, but many are still unaware of the dangers in doing so.

Macro viruses are thus unlikely to disappear in the foreseeable future. Users must become accustomed to the idea that some types of data file should be treated with the same caution as program files.

Virus behaviour after gaining control

Memory-resident viruses

Memory-resident viruses install themselves into memory as Terminate and Stay Resident (TSR) processes when they gain control.

They will normally intercept one or more interrupts and infect other objects when certain conditions are fulfilled, e.g. when the user attempts to execute an application (*Cascade*) or when the user accesses a drive (*Form*).

Switching the PC off will clear the virus from memory. However, warm booting (with *Ctrl-Alt-Del*) may not, because some viruses such as *Joshi* intercept the *Ctrl-Alt-Del* interrupt and survive the warm boot.

Note that macro viruses go ‘memory-resident’ by infecting the global environment.

Non-memory-resident viruses

Non-memory-resident viruses are active only when an infected application is executed. They execute their code completely at that stage and do not remain in memory. Other executables are generally infected only when an infected program is executed (e.g. *Vienna* or *Datacrime*).

This approach is just as infectious as that used by memory-resident viruses. Non-memory-resident viruses are also more difficult to spot, because they do not change the interrupt table or the amount of available memory, and their infectious behaviour can be more unpredictable.

Hybrids

Some viruses use a combination of these two methods.

The *Typo* virus, for example, infects executables on invocation of an infected program, but also leaves a small TSR element in memory after infection. The TSR section contains the payload, while the non-resident portion of the virus contains the replication code. In other hybrid viruses these functions might be allocated differently.

How viruses conceal themselves

Viruses often place obstacles in the path of anyone trying to find or eradicate them.

Two mechanisms are commonly used: encryption and interrupt interception. These and other techniques, including those used by macro viruses, are discussed below.

Encryption

Some viruses use encryption or scrambling of the virus code to make their structure appear different in each infected application.

This is designed to make the extraction of a fixed search pattern more difficult, because the majority of the virus code changes on every infection. However, before the virus code can be executed, it must be decrypted and the decryption routine **must be in plaintext** (unencrypted) form. This usually contains about ten or twenty bytes which are common to every infected executable. An encrypted virus will look identical only when it uses the same encryption key to encrypt its code.

Although encryption algorithms in most viruses are simple and the keys are straightforward, the possibilities for introducing complications are practically endless. For example, a virus can use two stages of encryption, where the key for encrypting the second stage is stored in an encrypted form in the first stage. Such ‘refinements’ make disassembly of the virus more difficult and even viruses encrypted using simple techniques can be complicated to disassemble.

Polymorphic viruses

One commonly-used technique is to make the virus vary the decryption routine between infections. These viruses are known as **polymorphic**. Little code remains the same between infections, making it impossible to extract a fixed pattern and somewhat complicating the search. An algorithmic approach has to be used; the virus scanner is told about a number of virus characteristics such as infective length, bytes which do not change between infections and so on, which are used to recognise virus-infected code.

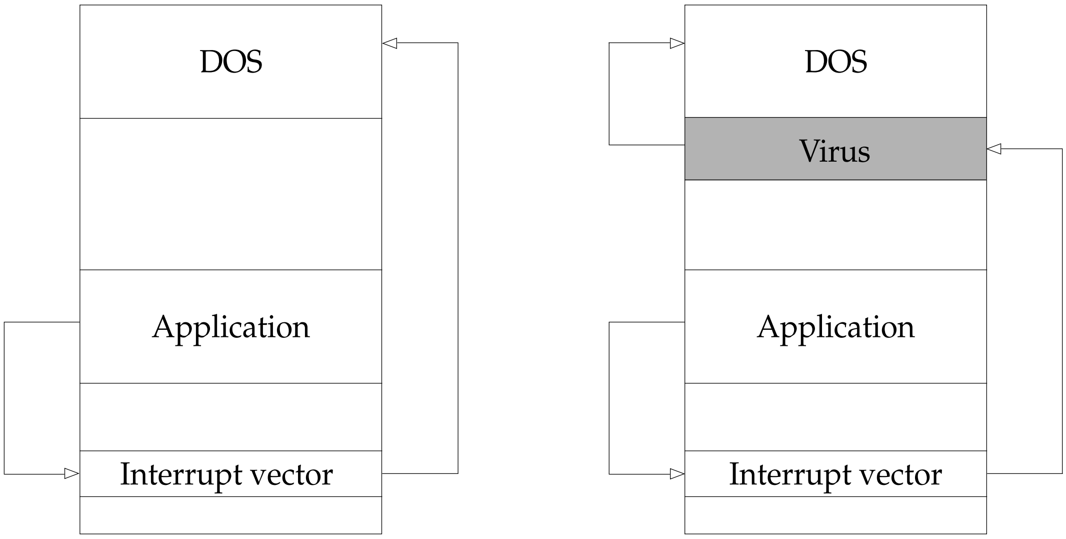
Interrupt interception: stealth viruses

Some viruses use interrupt interception to conceal themselves once they have gained control of the PC.

DOS applications use software interrupts to communicate with the operating system in a portable way. The jump addresses are stored in the interrupt table located at the beginning of memory. This is set up by the operating system to point to the correct addresses depending on the version of DOS.

When an application issues an interrupt, a jump occurs to a predetermined address. If a virus changes one or more of these addresses, any jumps to the operating system can be routed via the virus, which can then decide what to do with a particular request.

For example, if the *CMOS4* virus is active in memory and an application requests the operating system to read from disk the contents of the boot sector (the hiding place of *CMOS4*), the virus will return the contents of the pre-infection boot sector, instead of



Interrupt routing before and after the virus gains control the actual contents. *CMOS4* achieves this by modifying ('hooking itself into') the interrupt table.

Another stealth virus, *4K*, intercepts 18 functions of the DOS interrupt 21H. It can subtract 4,096 bytes from any infected file length displayed by the DIR command, and will even 'disinfect' any infected file if an application tries to read from it, only to re-infect it on closing the file. A DOS based virus scanner or a checksummer will therefore not discover *4K* in infected files if the virus is active in memory.

Joshi hides the contents of an infected boot sector by intercepting ROM BIOS disk services interrupt 13H and returning the contents of the original boot sector if a disk read is attempted. The virus also intercepts the keyboard interrupt 9H, traps *Ctrl-Alt-Del* (warm boot) and survives it.

Important! Correct anti-virus booting, **which includes switching the power off and booting from a clean, write-protected floppy**, is very important (but see also the 'Viruses which make clean booting impossible' section).

Macro viruses which use stealth

Macro viruses may subvert options in the host application in an attempt to evade detection or disinfection.

Since macro viruses can easily be eliminated by deleting macros via the Tools/Macro dialog, some viruses alter or disable this option. For example, *Winword/ShareFun* prevents the use of the Tools/Macros and File/Templates options.

Other macro viruses make it difficult to view macros. *Winword/Nuclear* marks macros execute-only, so that they cannot be edited, and are scrambled inside infected files. *Excel/Laroux*, an Excel macro virus, places macros on a 'hidden' sheet to conceal them.

Viruses which ‘infect’ IO.SYS

A small group of viruses (*3ARA3A*) infect hard disks by creating a second copy of IO.SYS, writing their own code into the original and labelling the original with a *Vol* label. When DOS is booted, it will use the infected version; when the system is checked with a scanner, the scanner will only check the original version of IO.SYS. The floppies are infected in a manner similar to boot sector viruses.

Such viruses have not become widespread as they are DOS version-specific. The infected hard disk’s Volume label is reported as ‘IO.SYS’, which should be enough to raise suspicion.

Viruses which make clean booting impossible

A few viruses such as *CMOS1* exploit the property of one manufacturer’s BIOS which allows the setting of ‘no floppy disks’ in the system configuration data stored in CMOS, thereby forcing the booting from the hard disk. The virus can prevent the clean boot by simulating the floppy boot, while, in effect, booting from the hard disk and thereafter using stealth measures to hide itself.

Note: This technique is BIOS-specific. The setting of ‘no floppy disks’ in CMOS is either detected as a corruption of CMOS or simply ignored by a large majority of BIOSes.

Such stealth measures are only effective on disk and the virus can still be detected in memory.

High-level language viruses

Writing viruses in high-level languages could also contribute to the difficulties of detection.

Most viruses are written in assembly language. They can ‘reach into the machine’ to a much greater extent than is possible when using a high-level language.

Furthermore, the code is smaller and more efficient, making the virus less obvious.

However, writing viruses in a high-level language does have advantages. Budding virus writers do not need to learn assembly language or to know very much about the inner workings of the PC.

Recognition of viruses written in high level language can also be difficult. Their binary image depends not only on the compiler used to create them, but also on the state of various optimisation levels used during compilation. Supposing that there are some 20 C compilers for DOS in existence, and each offers 6 possible optimisations and/or memory models, a single piece of virus source code in a high-level language could quite easily be transformed into 1,280 different binary images.

Furthermore, false positives can easily arise because similar segments of code appear in other legitimate programs compiled with the same compiler.

Burger's *Computer Viruses - A High-Tech Disease* contains viruses written in Compiled Basic and Pascal, and some viruses written in high-level languages have been discovered in the wild, e.g. *Jocker* (Pascal); *Kamikaze* (Turbo-Pascal); *Sentinel* (Turbo-Pascal); *TPworm* (C).

Virus side effects

Virus side-effects, or the virus 'payload', are the part of the virus of most concern to users.

Virus side-effects range from the trivial to the seriously malicious. They can take the form of:

- **Messages.**
  For example, *Winword/DMV* displays a message box confirming that infection has occurred. *Winword/Npad* displays a scrolling message in Word's status bar.

- **Pranks.**
  For example, *Yankee* plays a tune and *Italian* puts a bouncing ball on the screen.
- **Denial of access to services or files.**
  For example, *Parity Boot* hangs the computer and *WM97/NightShade* will password-protect the current document on Friday the 13th.
- **Data corruption.**
  For example, *JackRipper* swaps two words (four bytes) on the disk, and *Winword/Wazzu* moves words in a document randomly and inserts the word ‘wazzu’ at random.
- **Data destruction.**
  For example, *Jersualem* deletes every program run if it is Friday the 13th and *Michelangelo* overwrites parts of the hard disk and floppy disks on 6 March.

The threat of gradual corruption

Many users rely on backups as a safeguard against data loss. However, this can make them peculiarly vulnerable to viruses such as *Winword/Wazzu* which cause gradual and random data corruption. By the time that a user realises that corruption has been taking place, all the backups could already be corrupted.

Altered side-effects

The side-effects are normally the easiest part of the virus to program. They are also the easiest part to **change**.

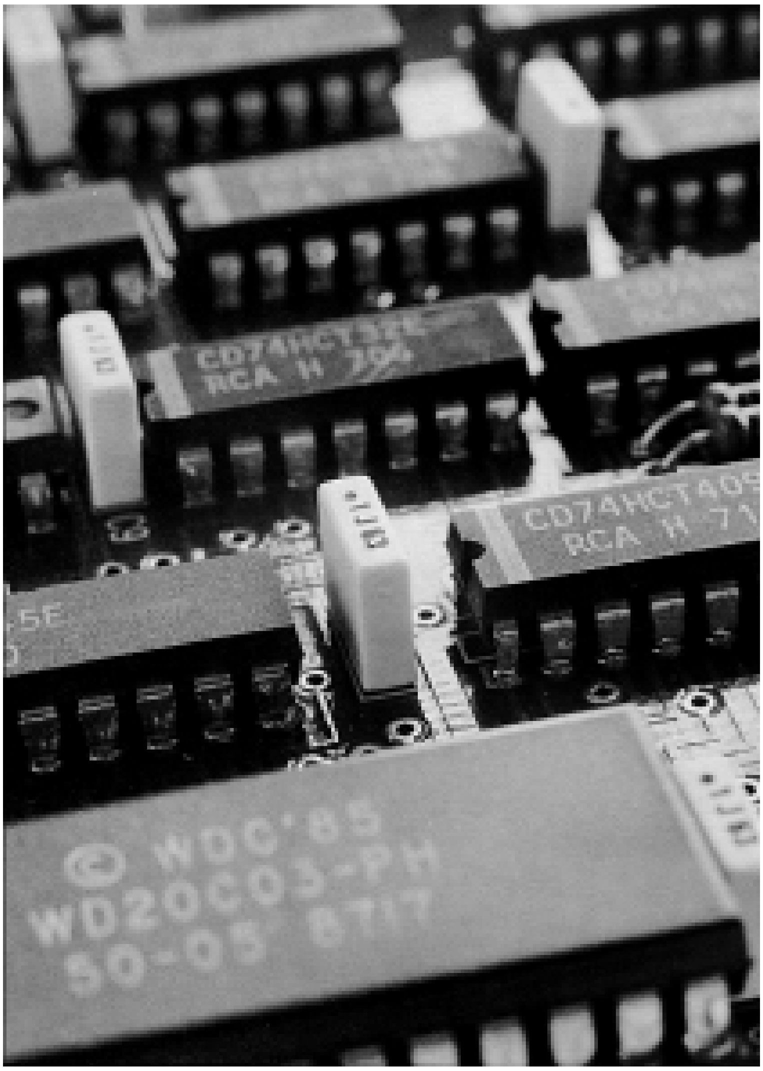
There have been several examples of mutated viruses having had their side-effects completely changed from the original. For example, the original version of *Cascade* simply causes characters to fall from the screen; the later version *Cascade-format* formats sector zero of the hard disk.

Unforeseen side-effects

Some viruses have side-effects that were not anticipated by the virus authors.

This may be because the virus code contains bugs, or because the virus has unpredictable effects under operating systems that have come into use since it was written.

For example, the boot sector virus *Form* is largely innocuous under DOS and 16-bit Windows. However, in the course of infection it saves the partition boot sector it has overwritten, along with some of its own code, in the last two sectors of the partition. Under non-DOS systems, this is fatal: the computer will not start thereafter. Windows NT is particularly vulnerable to this unforeseen effect.


CD74HCT32E
RCA H 706
CD74HCT40
RCA H 71
© WDC'85
WD20C03-PH
50-05 8717

How viruses spread

Computer viruses spread through the interchange of executable code, e.g. in programs or certain types of document.

This interchange is much more frequent and less well regulated on PCs than on minicomputers and mainframes. Thus, the virus threat has been confined mostly to PCs.

When can a PC become infected?

There are three actions that can cause infection of a computer:

- **Booting the PC from an infected medium.**
  Boot sector or multipartite viruses can infect the PC when this is done.
- **Executing an infected program.**
  Parasitic or multipartite viruses can infect the PC when this is done.
- **Loading an infected file into the relevant application.**
  Macro viruses can infect the PC in this way.

For more details about each kind of virus, see the previous chapter, ‘Viruses and their effects’.

Which areas of the PC are at risk?

In order to penetrate a computer, a virus must execute its code. Thus, any file which contains executable code has to be treated as a potential virus carrier. This includes files with interpreted BASIC commands, spreadsheet macros etc.

The following items are at risk:

1. **Master boot sector on hard disks.**
2. **DOS boot sector on hard and floppy disks.**
3. **DOS files IO.SYS and MSDOS.SYS.** CONFIG.SYS is a text file, and cannot contain a virus, but it could easily load and execute a virus written to infect device drivers.
4. **Device drivers, SYS files such as ANSI.SYS, RAMDRIVE.SYS.** Several viruses can infect these.
5. C**OMMAND.COM.** A small number of viruses (e.g. *1575)* target this file specifically while most parasitic viruses infect it like any other COM file.
6. **AUTOEXEC.BAT.** Normally affected by Trojan horses rather than viruses.
7. **Applications: EXE and COM files.** Many viruses attack these. Overlay files (normally OVL, OVR, OV1 etc.) are also subject to attack.
8. **Files with macros.** Microsoft Word, Excel, Access and Office 97 files are currently at risk from macro viruses.

The user should ensure that the code executed during all the above steps is virus-free and uncorrupted. This is harder than it seems because viruses can interfere with a number of these steps and lay obstacles in the path of anybody trying to discover them.

How a virus is spread

A virus becomes active on the PC only when virus code is executed, e.g. when an infected program is run, or when an infected document is opened in an application capable of running the viral macros in that document.

This active state is cleared by switching off the PC or by closing the application. However, the virus will usually remain on the PC and will become active again when the PC or the relevant application is restarted.

Example 1: Infecting PCs and disks

Figure 1 below illustrates how a boot sector virus spreads.

In steps 1 to 4, a PC becomes infected with the virus by booting from an infected floppy disk. The virus becomes active and also infects the hard disk.

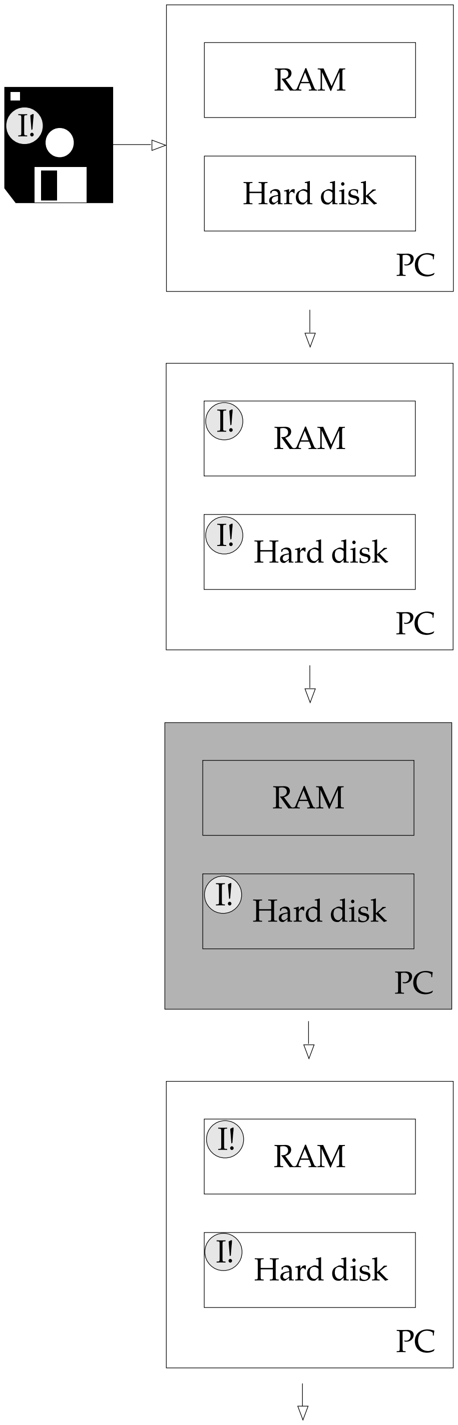
If the power is switched off, the virus disappears from memory, but **not** from the hard disk.

When the power is switched on and the PC booted from the hard disk, the virus once again becomes active.

Steps 5 and 6 demonstrate how the infection spreads onto further floppy disks.

Steps 7 and 8 show that booting from a clean floppy disk (not from the hard disk) can ensure that the virus is not active while anti-virus action is taken.

See the ‘Anti-virus measures’ chapter for more information.



1. In an uninfected PC, both the RAM and the hard disk are free from infection. An infected floppy disk is introduced into the floppy disk drive.

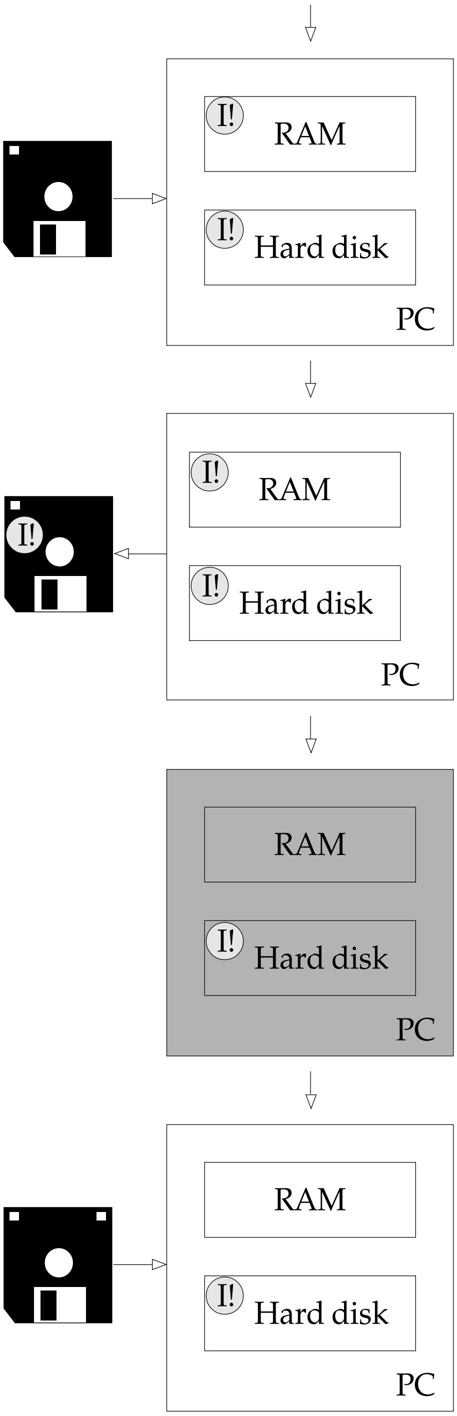
I! shows infected items

2. When an infected program from the floppy disk is run, the hard disk becomes infected and the virus becomes active.

3. If power is now switched off, the hard disk remains infected while the contents of RAM (including the virus) are lost.

4. When the PC is switched back on and booted from the (infected) hard disk, the virus becomes active once again.

Figure 1: Infecting a PC and disks (continued opposite)



5. If an unprotected, clean floppy disk is then used...

6. ...it immediately becomes infected. Any unprotected floppy disk which is used in this PC while the virus is active becomes infected.

7. If power is now switched off, the hard disk once again remains infected, while the contents of the RAM (including the virus) are lost.

8. The virus can be kept inactive by switching the PC back on with a clean write-protected system disk in the floppy disk drive. Despite the fact that the hard disk remains infected, the virus is not active. Anti-virus actions can commence.

Figure 1 (continued): Infecting a PC and disks

Example 2: Infecting applications and documents

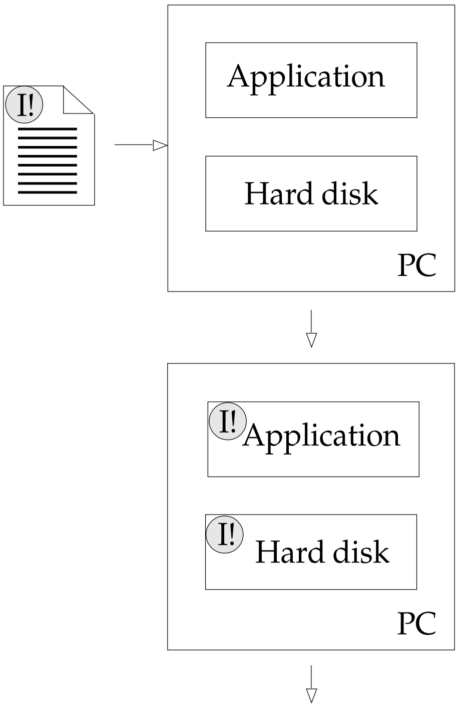
Figure 2 below illustrates how a macro virus spreads.

In steps 1 to 4, an application is infected by opening an infected document.

The application is then closed, but it saves changes in its global environment to the hard disk, thereby infecting it. When the application is restarted, the viral macros again become active.

Steps 5 to 6 demonstrate how the infection spreads onto further documents.

See the 'Anti-virus measures' chapter for information on disinfecting macro viruses.

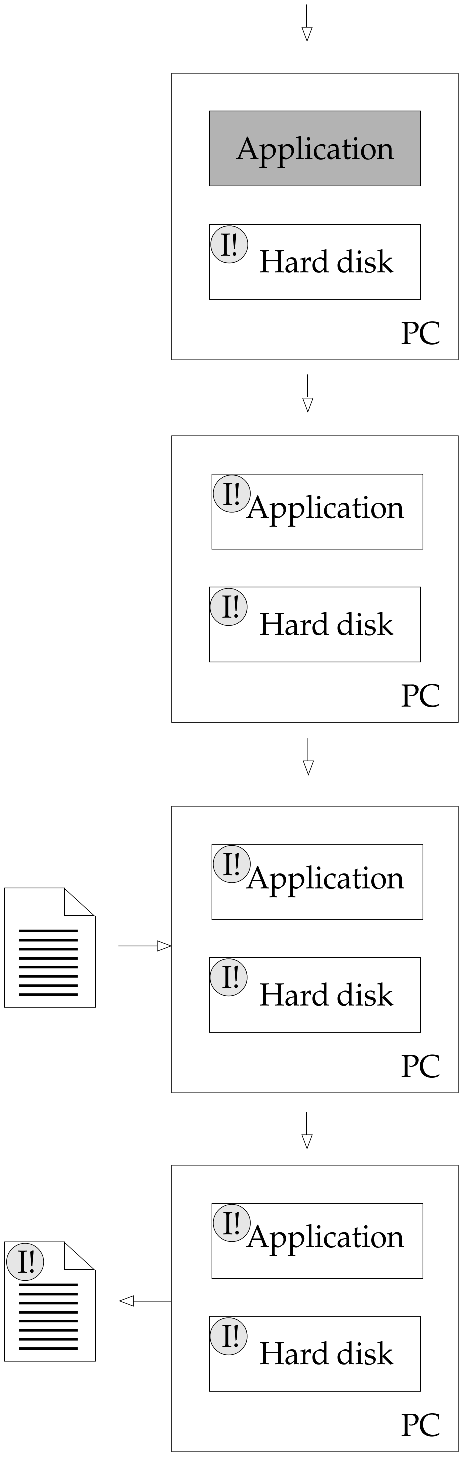


1. In an uninfected PC, both the application and the hard disk are free from infection. An infected document is received on disk or CD, or in an email.

(I!) shows infected items

2. When the infected document is loaded, the application's global environment becomes infected. Word saves global changes to the hard disk on exit, thereby infecting it.

Example 2: Infecting an application and documents (continued opposite)



3. The application is no longer active, but the changes to its global environment, including the viral macros, remain on the hard disk.

4. When the application is re-started, the viral macros become active again.

5. If an uninfected document is loaded...

6. ...it immediately becomes infected.

Example 2 (continued): Infecting an application and documents

Viral infection on networks

The interchange of files on non-networked PCs is mostly done by floppy disks and in consequence is relatively slow and physically controllable. PC networks enable high speed interchange and sharing of data and executables. This automated interchange is comparatively difficult to control.

The danger from a large-scale virus attack in a **non-networked** organisation is usually limited to a few PCs before it is spotted. In such an environment it is relatively easy to contain a virus. However, the likelihood of a large-scale virus attack in a **networked** organisation is much greater and the chances of a successful early containment much smaller.

Virus infiltration routes and methods

Some user actions carry a high risk of spreading viruses. The following list of virus infiltration routes has been assembled by analysing real-life cases.

The Internet

The Internet has played an important role in the spread of viruses, both as an easily accessible repository of virus collections and as a medium for the rapid exchange of infected programs or documents. See ‘Web browsers’ and ‘Email’ for further information.

Web browsers

World Wide Web browsers increase the risk of virus transmission by making it easier for users to exchange and access documents or programs.

Web browsers make downloading executable programs from remote machines a trivial exercise. This brings with it the risk of infection by viruses that attach themselves to executables. Users should be made aware of the inherent risks of downloading.

In addition, many browsers are configured to launch the appropriate application when a certain type of file is delivered. Internet Explorer can use the components of Microsoft Office almost invisibly, so that when a Word document is downloaded, Word starts up to view the file inside the Internet Explorer window. Thus macro viruses have their chance to execute and seize control, infecting the machine.

Email

The present risks

At the time of writing, it is not possible to become infected simply by reading an email (see the 'Virus hoaxes and scares' chapter).

However, computers can be infected by reading attachments (such as Word files) included in an email. If the body of an email consists entirely of an attached file, then 'reading' the email could be considered to include opening and reading its attachment.

The future risks

At the time of writing, there was speculation that the forthcoming Windows 98 operating system, used with Microsoft Outlook, could allow viral code to be actuated simply by reading an email.

Certainly, email software may be developed that automatically detaches documents or executes macros when an email is read, allowing infection by macro viruses.

Whatever happens users should be made aware of the ease with which viruses may be sent and received via email.

Pirated software

It is easy to copy software and in most countries it is illegal to do so. Games are probably the most commonly pirated software and they tend to move between PC users at a far greater frequency than ‘serious’ software. For this reason, they are also most prone to picking up a virus on the way.

Shareware

Shareware is a concept developed in the USA. The software carries the traditional copyright, but anybody is encouraged to copy it and pass it on to others. Whoever uses it is under moral obligation to send a small sum to the author. Unfortunately, shareware distribution is not without problems. Although most authors send ‘the latest version’ once payment has been received, users end up trying (and using) the original version obtained from a friend of a friend of a friend. By the time one receives ‘the latest version’, the computer may be infected many times over with any viruses the original software picked up on the way.

Interestingly, shareware in compressed files (ZIP, ARC etc.) exhibits a certain resistance to carrying viruses if the original ZIP-ed file is passed on.

Disks supplied by computer magazines

Many computer magazines supply disks containing free software. On a number of occasions in the early 1990s such disks were found to contain viruses.

Although greater virus awareness has made it unlikely that such disks will be infected now, this is still a potential route for infection.

Public domain software

Unlike shareware, public domain software is completely free for anybody to use. Unfortunately, it suffers from the same distribution risks as shareware, with the added disadvantage that there is often nobody to supply ‘the latest version’.

Some public domain software is distributed as source code, which can be compiled by the user. This strategy offers a great barrier to virus spread.

If you have to use public domain software, make sure that you obtain it from a reliable source, such as a CD which has been in circulation for a long period.

Home PCs

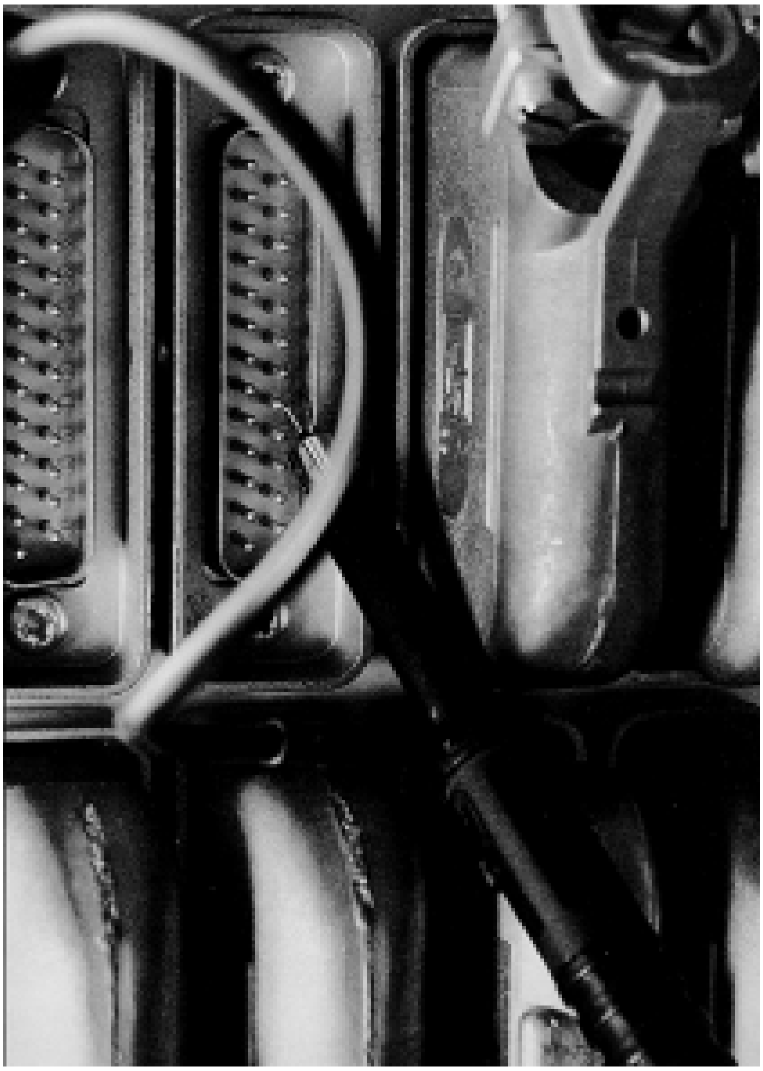
A surprisingly large number of infections in business PCs occur through the use of home computers for company work. Office workers who take documents home on disk, for example, can unwittingly infect their files with any viruses that are on their home PC. When they take the disk back to the office and resume work there, their office PC will be infected.

Shrink-wrapped software

Shrink-wrapped software normally refers to commercial software packages which come in a shrink-wrapped sealed container - usually for legislative purposes rather than anti-virus measures. By breaking the seal, the user implicitly agrees to abide by the manufacturer’s terms and conditions. There is also a good chance that the software has not been tampered with from the time it left the manufacturing plant.

There have however been a few cases of shrink-wrapped software containing viruses. The most celebrated was the release of the *Winword/Concept* virus on Microsoft’s Windows 95 Software Compatibility Test in 1995.

Major software companies operate stringent QA (Quality Assurance) procedures in order to prevent virus propagation into production software. Although there is always a chance that shrink-wrapped software may contain a virus, the probability, in practice, is small.



Anti-virus measures

The fight against computer viruses involves five kinds of counter-measure:

- **Preparation**.
- **Prevention**.
- **Detection**.
- **Containment**.
- **Recovery**.

Preparation

The following subsections outline what should be done **before** a virus attack occurs.

Regular and sound backups

Backups are something which everybody should do anyway, but are especially important in case of an attack by a destructive virus. If data loss occurs, backups make it possible to restore the system efficiently. As part of the backup procedure, the master disks for all software (including the operating system) should be write-protected and stored in a safe place. This will enable speedy restoration of any infected executables.

Important! The backups should be **sound**. There is little point in doing them **unless the data can actually be restored**. Backups should be tested from time to time by performing complete restorations of the system.

Preparing clean boot disks

A clean boot disk, i.e. an uninfected write-protected system floppy disk, is normally an essential part of the virus recovery procedure. A separate clean boot disk will be required for each different operating system version, and it is vital that these are created before the infection occurs.

To create a bootable system disk, enter at a DOS prompt:

```
FORMAT A: /S
```

Copy HIMEM.SYS, EMM386.EXE, FDISK.EXE, SYS.COM, DEBUG.EXE, SMARTDRV.EXE, SCANDISK.EXE (or CHKDSK.EXE for MS-DOS 5 and before), and FORMAT.COM onto the disk. HIMEM.SYS is an Extended Memory (XMS) driver which allows SWEEP to use all the PC's memory thereby improving performance. SMARTDRV.EXE is a disk caching program which improves SWEEP's performance by minimising the amount of disk access required when traversing the directory structure of a disk.

Create a CONFIG.SYS file with the following lines:

```
DEVICE=A:\HIMEM.SYS
DEVICE=A:\EMM386.EXE
DOS=HIGH
FILES=15
BUFFERS=40
```

Create an AUTOEXEC.BAT with the following lines:

```
A:\SMARTDRV.EXE
SET TEMP=C:\
```

Make the disk write-protected (to ensure that it cannot become infected with a virus), and label it with the operating system for which it was created.

If a computer becomes infected, this disk will be used to boot the computer. This will ensure that various items on the computer can be examined through a 'clean' operating system, giving the virus no chance to employ hiding techniques.

DriveSpace, DoubleSpace, Stacker, etc.

Disk compression software increases the effective size of the hard disk by compressing files on-the-fly. However, if the PC is rebooted from a boot disk without the compression manager software, only the host drive contents (i.e. the uncompressed stub) will be visible. Using the command:

```
FORMAT A: /S
```

with DOS 6.0 or later will automatically copy the relevant DriveSpace or DoubleSpace drivers to the boot disk. If a third-party compression driver (i.e. one not supplied with the operating system) is used, additional action may be required to copy the necessary drivers to the floppy disk. If in doubt, consult the software's documentation.

Hard disk overlay managers

In order to get large hard disks to work with older PCs, hard disk management software may be used, without which the contents of the hard disk may not be visible. Consult your system manuals to find out what driver files are required on your boot floppy.

Contingency plan

This plan, which can be put into action in case of a virus attack, is usually part of a point-by-point checklist and should include information on the following:

- The person within the organisation responsible for dealing with the attack and a deputy.
- The consultant(s) outside the organisation who can be called in to help deal with the attack.

- Exact procedure for isolating infected objects.
- Public Relations procedure to prevent unauthorised leaks about the attack spreading outside the organisation.

Prevention

Preventing a virus from penetrating an organisation is equivalent to the military guarding their camps and states guarding their frontiers. Medical parallels can also be drawn, with only sterile surgical instruments being allowed into an operating theatre.

The need to communicate introduces a potential virus entry path into any environment. Application software has to be purchased or updated; new operating systems installed; disks and documents interchanged. The higher the volume of inbound traffic, the more opportunity a virus has to enter the environment.

There are several practical techniques for strengthening the fence: **creating user awareness**, implementing **hygiene rules**, **disabling floppy disk booting**, using **disk authorisation software**, providing a **'dirty' PC**, providing a **quarantine PC**, and using **on-access virus scanning**.

Creating user awareness

Creating user awareness is the most important factor in establishing an effective virus prevention policy. Users must be made aware that execution of unauthorised software (such as demonstration disks and games) can lead to virus penetration of the best guarded environment and consequent losses to the organisation.

Strengthening awareness is a matter of common sense and measures include the use of leaflets, posters, virus demonstrations, presentations, showing educational videos and so on.

Hygiene rules

The observance of 'hygiene' is by far the most effective way of preventing a virus attack. A virus has little chance of infecting a computer if that computer is not networked, has a limited number of users (preferably only one), and is never used with disks from other sources.

The essence of hygiene is the principle that every executable item which is to run on a computer should be treated with suspicion. This includes **demonstration disks**, **shareware**, **public domain**, **Word and Excel documents received from outside the organisation**, and **almost everything from the Internet** (unless it originates from a reputable company). Any disk which does not look mass-produced should be treated with added suspicion. This is simply due to the fact that if something is mass-produced, there is an increased likelihood that somebody will have noticed an infection already and alerted other users.

Disabling floppy disk booting

On most modern PCs, booting from floppy disks can be disabled, so that the PC always boots from the hard disk. This simple measure can drastically reduce the danger from infections.

Disabling floppy disk booting is usually done by using the setup software; unfortunately, the way of doing this depends on the PC.

Disk authorisation software

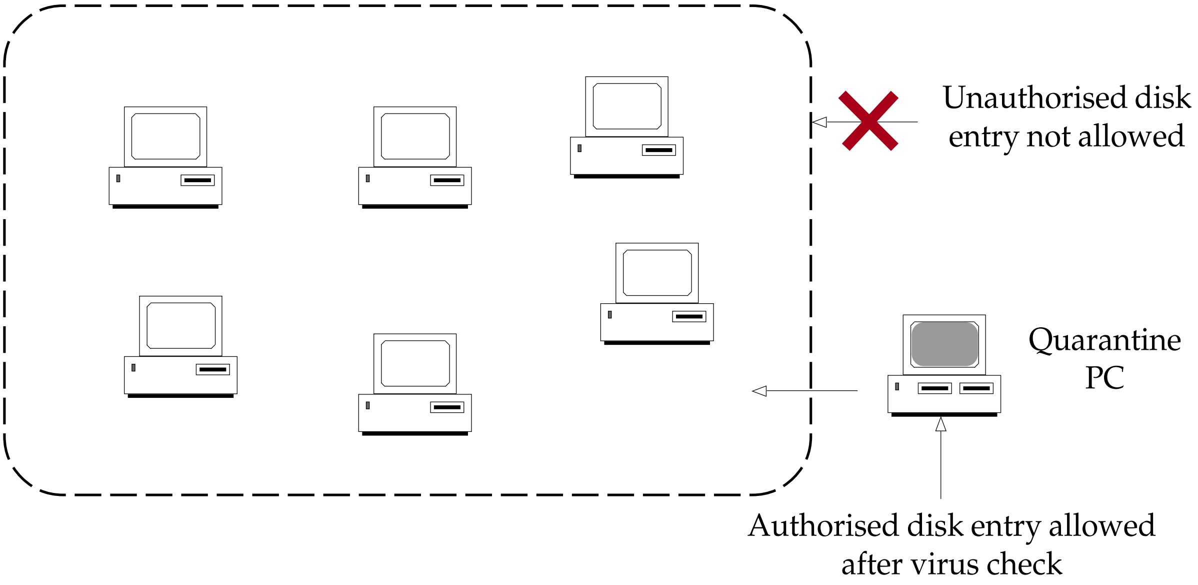
Disk authorisation software can be used very effectively to enforce anti-virus procedures, thereby decreasing the likelihood of virus penetration. Any disks from **outside** the organisation are unreadable on the PCs **inside** the organisation. Once the disks have been checked for viruses, they can be converted into the company format, thus becoming readable.

Products such as Sophos' D-FENCE are also effective in preventing PCs outside the organisation from reading authorised company disks.

The use of disk authorisation software is a very effective way of enforcing the company anti-virus policies, while not requiring the administrative overheads of a full access-control package.

Dirty PC

A dirty PC is a physically isolated machine, not connected to networks, which can be used for playing games and doing anything which would be dangerous to do on a machine used for day-to-day work. Employees should be encouraged to use it to try out any software coming from outside, including demo disks and games. No company work should **ever** be done on that machine, and no disks used on the dirty PC should ever be used in any other computer. Anti-virus software should be run as often as possible to check the hard disk of this machine for virus presence. This concept is a useful tool against viruses, if carefully controlled.


Unauthorised disk entry not allowed
Quarantine PC
Authorised disk entry allowed after virus check

Quarantine PC used for checking all incoming disks

Quarantine PC

A quarantine PC is a stand-alone machine not connected to networks. Apart from having permanently installed anti-virus software, this PC is kept completely clean and is used to check incoming disks for viruses as well as for trying out new software. It is similar in function to the barrier guard in military barracks.

Only disks and programs which have been cleared are allowed through.

The quarantine function is often linked with the gateway function when using disk authorisation software.

On-access virus scanning

On-access virus scanning can prevent infection by denying access to infected items.

Such scanning is performed by programs which stay in memory and check each item as the user attempts to access it.

The process is transparent and requires no user intervention. Indeed, in most organisations, users do not choose whether to use the software or not; it is enabled by default.

For more details of virus scanning, see the ‘Detection’ section below.

Detection

Effective virus detection (with anti-virus software) is the main technique for preventing a virus entering an organisation. It is also an essential tool for detecting a virus should the virus penetrate the initial virus-prevention obstacles placed in its way.

Three types of anti-virus software are commonly used:

- Scanners.
- Checksummers.
- Heuristic software.

Scanners

Scanners are the most widely used type of anti-virus software. They rely on the knowledge of what particular viruses look like in order to discover them. This implies that if a virus was not known to the scanner manufacturer at the time of manufacture, the product is not going to discover it.

Scanners need to be updated regularly in order to be effective: monthly or quarterly updates are common.

The most commonly used types of scanners are on-demand scanners, on-access scanners and server based scanners. The former two are used on clients and stand-alone workstations, while the latter is used on the server itself.

```
0EF2:0501 FA            CLI
0EF2:0502 8BEC          MOV  BP,SP
0EF2:0504 E80000        CALL 0507
0EF2:0507 5B            POP  BX
0EF2:0508 81EB3101      SUB  BX,0131
0EF2:050C 2E            CS:
0EF2:050D F6872A0101    TEST BYTE PTR [BX+012A],01
0EF2:0512 740F          JZ   0523
0EF2:0514 8DB74D01      LEA  SI,[BX+014D]
0EF2:0518 BC8206        MOV  SP,0682
0EF2:051B 3134          XOR  [SI],SI
0EF2:051D 3124          XOR  [SI],SP
0EF2:051F 46            INC  SI
0EF2:0520 4C            DEC  SP
0EF2:0521 75F8          JNZ  051B
```

Cascade decryption routine - scanning software can use shaded bytes for recognition

On-demand scanners

On-demand scanners can check one or more logical drives, or even individual executables and documents, for viruses. The main disadvantage of this approach is that a user has to initiate or schedule the scans.

On-access scanners

These are programs which stay in memory and perform virus checking transparently, without the need for user intervention.

On-access scanning software comes in the form of a TSR (Terminate and Stay Resident process) under DOS, a VxD (Virtual Device Driver) under Windows 3.x and Windows 95 and an FSFD (File System Filter Driver) under Windows NT.

Server based scanners

Server based scanners are run on the file server. Scanners generally provide virus detection on the server itself, while the Sophos InterCheck technology enables you to check the clients using the server based scanner. The main advantages of using the scanner on the server are easy centralised updating, fewer restrictions on memory, and lower processing power needed to detect polymorphic viruses.

	File 1:		**File 2:**
	010101000		010101000
	101000101		101000101
	001001010		001001010
	001010010		001010010
	000000100		000000100
	100**1**01000		100**0**01000
	001010100		001010100
	001010100		001010100
Checksum:	07FC67623	**Checksum:**	567FB1837

1 bit change results in a completely different checksum

Checksummers

Checksummers are normally used for virus detection on clients. Checksums of all executable objects on the client are taken when there are no viruses present. The same process is then performed at regular intervals and any changes investigated. Since executable objects do not normally change (unless attacked by a virus, during software upgrades etc.), an unexplained change needs to be investigated.

The main advantage of checksummers is that a checksummer does not need to know anything about a particular virus in order to recognise it and thus does not need updating. The main disadvantage is that a reported change needs intelligent interpretation to distinguish between a virus attack and a legitimate change; hence the number of false alarms can be high.

Heuristic software

Heuristic software attempts to recognise viruses, known or unknown, on the basis of general rules about what viruses look like.

Because it does not depend on identification of particular viruses, it is usually thought of as software which does not need updates. Unfortunately, in practice that is not true. The virus-writing community is constantly engaged in writing viruses which are not recognised by the popular heuristic software. As soon as those viruses appear in the wild, the heuristic software author has to update his program with new recognition rules, which causes the virus-writing community to produce a modified virus etc. Heuristic software needs to be updated less frequently than scanners, but the need to update is nevertheless there.

The main disadvantage of heuristic scanners is an unacceptably high number of false alarms.

Containment

If a virus is detected inside a company, infected objects have to be identified and isolated. A contingency plan prepared in advance will be extremely valuable at the moment of virus discovery. A point-by-point checklist makes it more difficult to forget an important item in the general panic which sometimes follows a virus attack.

Network access

Depending on where in the network the virus has been discovered, the type of the network and the type of the virus, one may decide to disconnect the PCs physically from the network.

Disk interchange

Any unauthorised disk interchange between PCs should be temporarily suspended. Masking tape placed over disk drives is a good physical indicator that the drives should not be used.

Write-protect tabs

All floppy disks which are not intended to be infected should be protected by opening the write-protect shutter to ensure that nothing can be written to the disk.

Write-protection on disks is a hardware function and cannot be overridden in software.

Recovery

Recovery from a virus attack involves:

1. Identifying what is infected.
2. Eliminating the virus.
3. Recovering from any virus side-effects.

Identifying what is infected

The procedure for eliminating the virus depends on which areas of the PC are infected. Use virus detection software to determine the virus type and the area(s) of the computer infected. There are four areas on a PC that are open to viral attack:

1. Master boot sector.
2. DOS boot sector.
3. Executable files (COM, EXE, etc.).
4. Documents capable of containing macros.

It is also vital to locate all infected items. Just one infected item that escapes the virus elimination procedure can restart the infection.

Eliminating the virus

The following information is intended as a guide only. The virus detection software should be consulted to confirm the procedure for the specific virus.

Do not forget to preserve a virus sample in a safe place for detailed analysis by one of the organisations involved in virus research. This is especially important if the virus is not recognised by current scanners, implying that it is a new or modified version.

Infected boot sectors on the hard disk

Anti-virus software, including Sophos Anti-Virus, may be able to disinfect hard disk boot sectors.

If disinfection is not possible or fails, it may still be possible to eliminate the virus by overwriting the infected boot sector with a clean one.

Boot the PC with a clean boot disk to ensure that the virus is not active in memory (see the ‘Preparing clean boot disks’ subsection of the ‘Preparation’ section above). Check that you can see the contents of the hard disk after the clean boot, e.g. by entering

```
DIR C:
```

If the directory listing is okay, the **master boot sector** can be overwritten with the command

```
FDISK /MBR
```

and the **DOS boot sector** can be overwritten with the command

```
SYS C:
```

If using the SYS command to overwrite a DOS boot sector virus, it is essential that the clean boot disk was for the same version of DOS as the infected PC.

Also, if the infected PC is not a DOS machine (e.g. it is running Windows NT), the DOS SYS command should not be used because it is operating system and version specific.

Important! If the contents of the hard disk are not visible after a clean boot, i.e. the directory listing is not okay, contact anti-virus experts for advice. Some boot sector viruses require additional action for full recovery. For example, *OneHalf* encrypts the boot sector so that it is only readable when the virus is in memory.

Infected boot sectors on floppy disk

If the anti-virus software does not or cannot disinfect the boot sector virus from an infected floppy disk, reformat the disk as follows.

Boot the PC with a clean boot disk, copy the valuable data from the infected disk to a clean destination (it is safe to copy files if the PC has been booted from a clean boot disk), and reformat the disk using

```
FORMAT A:
```

if the disk is in drive `A:`.

Infected executable files

It is generally inadvisable to attempt to disinfect infected executables. This is because it is not possible to ensure that the executable has been properly restored after disinfection; it may be unstable which may put valuable data at risk.

Boot the PC with a clean boot disk to ensure that the virus is not active in memory. Then locate all the infected executables with virus detection software, delete them, and restore clean versions from the original installation disks, from a clean PC, or from sound backups.

Infected documents

Anti-virus software, including Sophos Anti-Virus, may be able to disinfect infected documents. Although it is not necessary to boot from a clean system disk before attempting automatic disinfection, it is important to ensure that the application that created the document is not open when disinfection is attempted.

In some cases it is also possible to edit the macros manually from the infected document using the relevant application. However, some macro viruses now operate a form of stealth to prevent users from doing this. For example, *Winword/ShareFun* prevents the use of the Tools/Macro and File/Templates menu options.

Recovering from virus side-effects

Recovery from virus side-effects depends on the virus. In the case of innocuous viruses such as *Cascade*, recovery from side-effects is not necessary, while in the case of a virus such as *Michelangelo*, recovery will involve the restoration of a complete hard disk.

Some viruses, such as *Winword/Wazzu*, gradually make minor changes to users' data. This sort of corruption (e.g. the removal of the word 'not' from a sentence in a Word file) can be very hard to detect and highly undesirable.

The most important thing when recovering from virus side-effects is the existence of **sound backups**. Original executables should be kept on write-protected disks, so that any infected programs can easily be replaced by the original clean versions.

Sometimes it is possible to recover data from disks damaged by a virus. This is a rather specialist task performed by commercial data recovery agencies and can be very expensive.

Other points

There are a few other points worth bearing in mind during recovery from a virus attack:

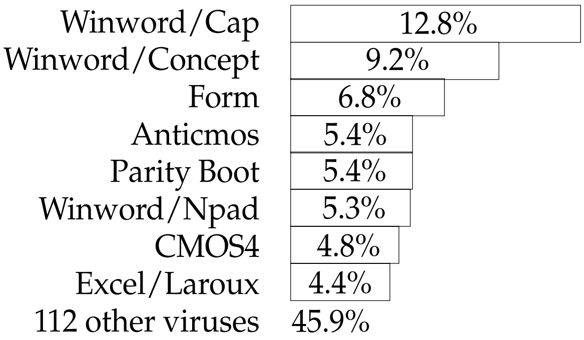
- Discover and close loopholes which allowed the virus to enter the organisation.
- Inform any possible recipients of the infected objects outside the organisation that they may be affected by the virus.
- Thoroughly check all potentially infected objects (e.g. disks, executables, documents). Even one infected object can restart the infection.
- Consider the implications to the organisation of the bad publicity.

151 FS

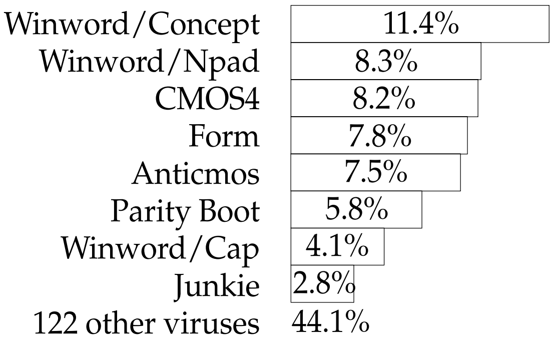
Common viruses

Virus statistics

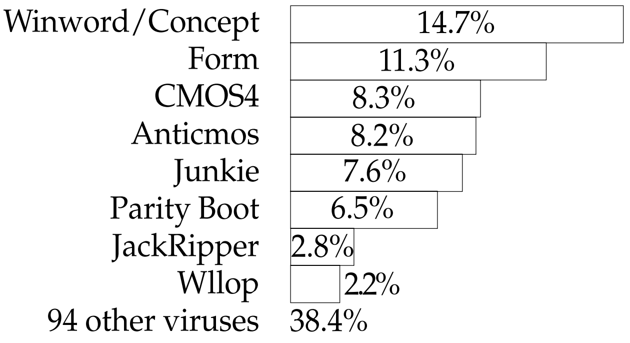
The graphs below show viruses reported to Sophos in each six month period.



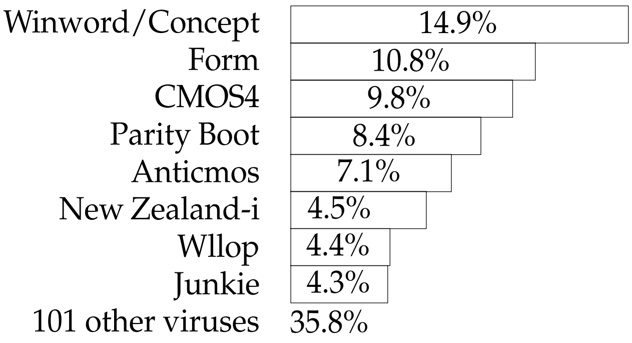
Virus reports from 1st July 1997 to 31st December 1997 (938 reports)



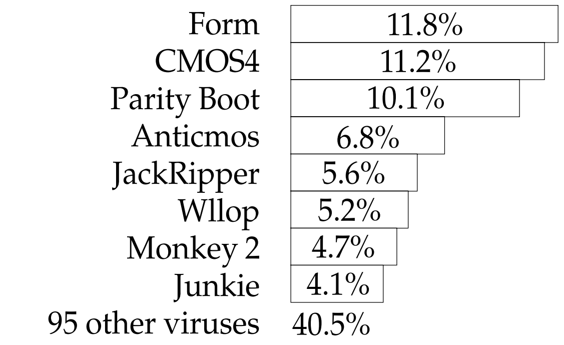
Virus reports from 1st January 1997 to 30th June 1997 (883 reports)



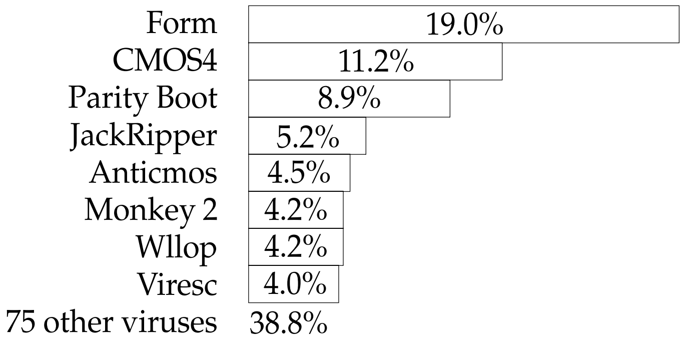
Virus reports from 1st July 1996 to 31st December 1996 (647 reports)



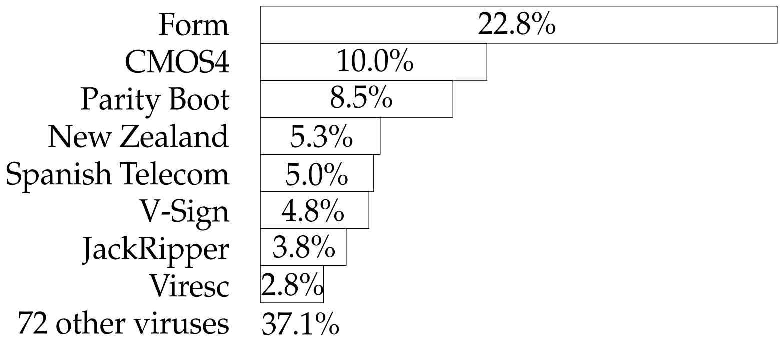
Virus reports from 1st January 1996 to 30th June 1996 (1033 reports)



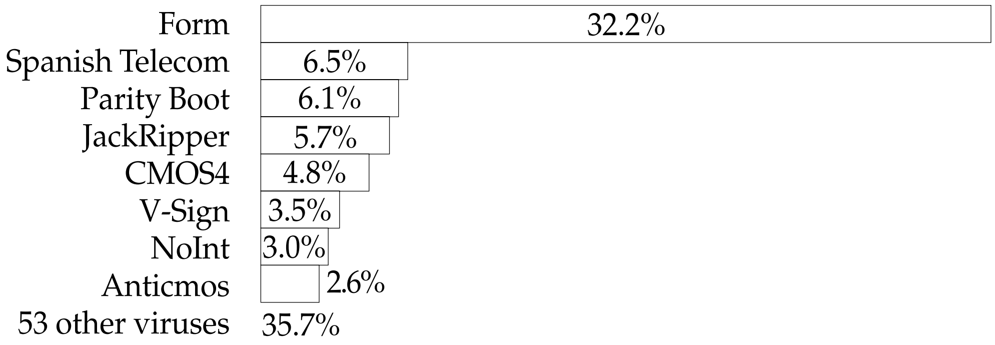
Virus reports from 1st July 1995 to 31st December 1995 (783 reports)



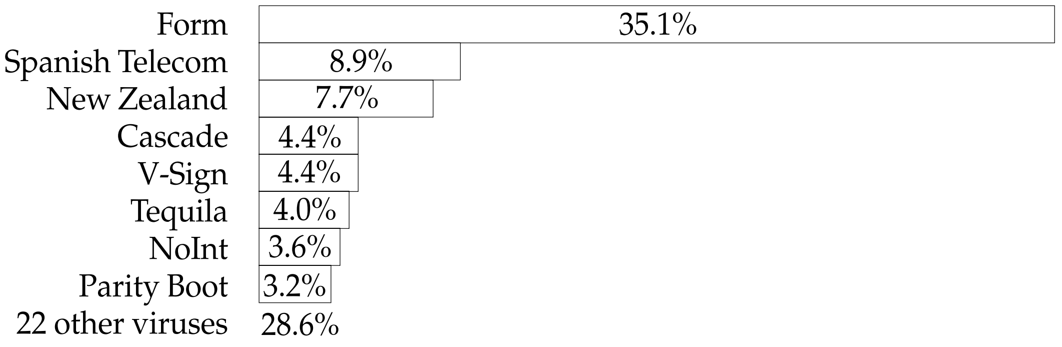
Virus reports from 1st January 1995 to 30th June 1995 (573 reports)



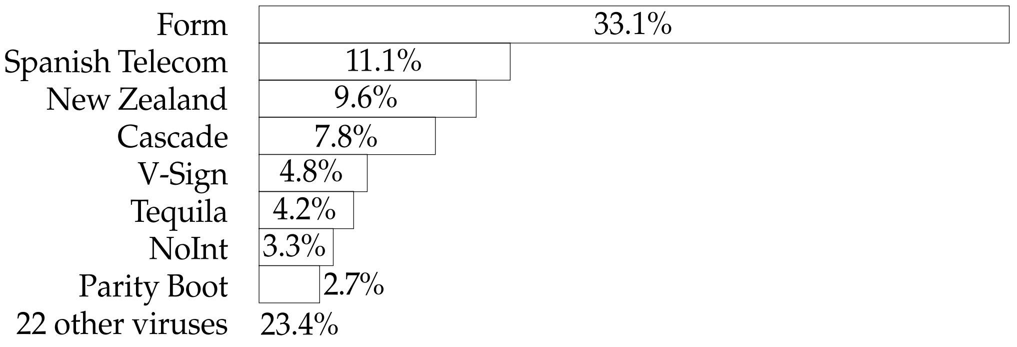
Virus reports from 1st July 1994 to 31st December 1994 (248 reports)



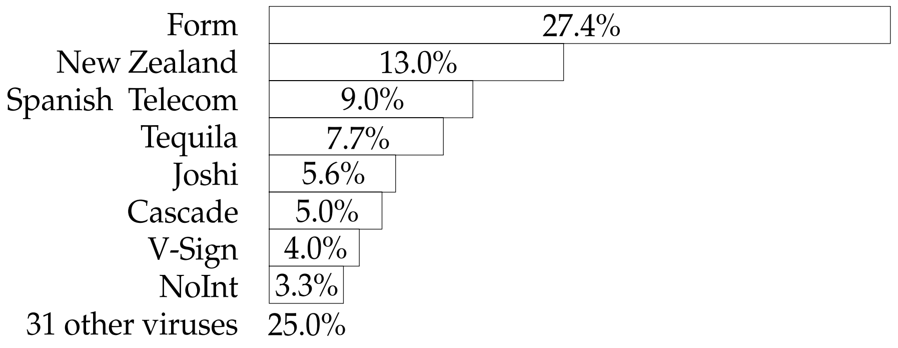
Virus reports from 1st January 1994 to 30th June 1994 (230 reports)



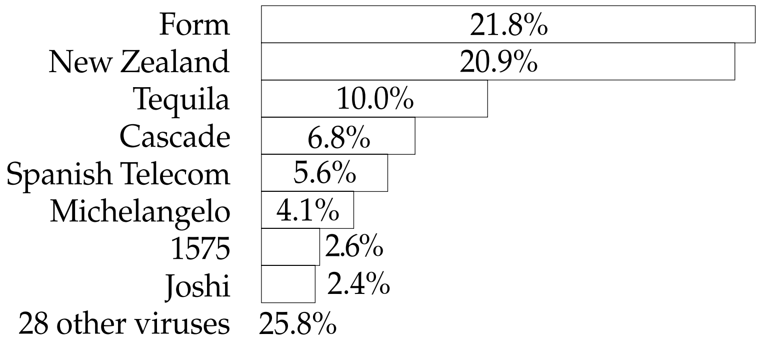
Virus reports from 1st July 1993 to 31st December 1993 (248 reports)



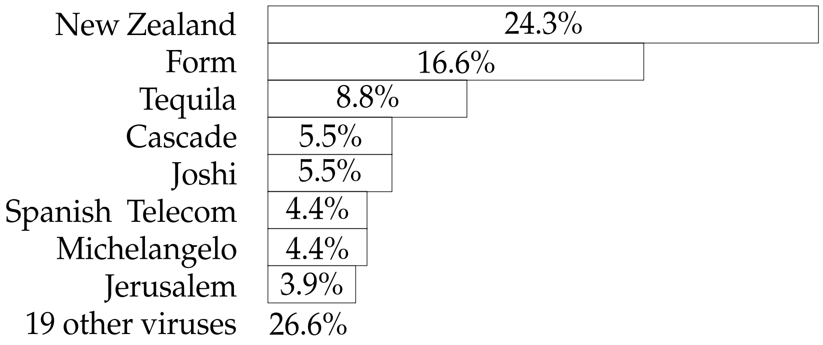
Virus reports from 1st January 1993 to 30th June 1993 (332 reports)



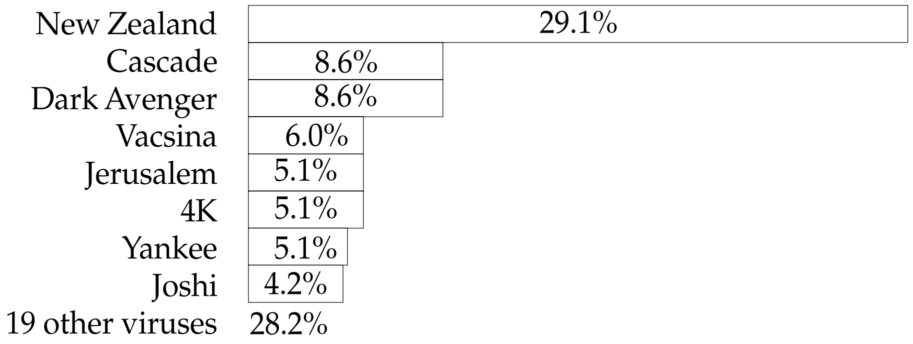
Virus reports from 1st July 1992 to 31st December 1992 (299 reports)



Virus reports from 1st January 1992 to 30th June 1992 (340 reports)



Virus reports from 1st July 1991 to 31st December 1991 (181 reports)



Virus reports from 1st January 1991 to 30th June 1991 (117 reports)

Viruses in the wild

The number of viruses known to the research community continues to increase rapidly. The first viruses appeared in 1986-87, by February 1992 there were about 1,500 known viruses, and by June 1998 this number had risen to around 15,000.

Only about 300 viruses are currently causing real problems in the wild, and reports to Sophos indicate that currently just eight (and their close mutations) are responsible for about 55% of all infections.

Almost all the reported cases involved a few PCs, but a number of large-scale attacks (100+ PCs) were also reported. These usually involved file servers and were in the majority of cases attributable to poor use of network security features.

Perhaps the most significant feature of the past three years has been the rapid rise of the macro virus. Macro viruses did not appear in the wild until the second half of 1995, but within two years accounted for 46% of all virus infections reported to Sophos.

Many users may still be unaware of the dangers of exchanging Word or Excel documents. The spread of macro viruses has also been aided by the growth in popularity of the Internet and use of email.

The virus report statistics also show that some kinds of viruses are more successful than others in the wild. Macro viruses accounted for only 6% of all known viruses, but for 46% of all virus infections. Similarly, boot sector viruses accounted for 6% of known viruses, but for 45% of all reported infections. However, parasitic viruses, which represent the vast majority of known viruses (79%) accounted for only 3% of reports. Clearly macro and boot sector viruses spread more successfully than parasitic viruses.

Analyses of common viruses

The following are analyses of common viruses.

1575

Aliases: Green caterpillar.

Type: COM and EXE file infector.

Resident: Yes.

Stealth: No.

Trigger: When a file which has been infected for over two months is run, COMMAND.COM is already infected (or does not exist in the root directory), and there is another copy of the virus already resident in memory.

Payload: A green caterpillar moves from the top left-hand part of the screen turning the text yellow. If uninterrupted this lasts for three minutes. Any input which causes the screen to scroll up results in the caterpillar jumping back.

Comments: The virus specifically looks for C:\COMMAND.COM and infects it. It traps the DOS find-file functions and consequently infects files when the DIR command is used. Infected files are usually corrupted by the virus and will often not run properly.

4K

Aliases: 4096, Frodo, IDF, Israeli Defence Forces.

Type: COM and EXE file infector.

Resident: Yes.

Stealth: Yes.

Trigger: 22nd September.

Payload: May display text related to J R R Tolkien's *Lord of the Rings*, and usually causes the system to hang when an infected file is executed.

Comments: This virus searches for COM and EXE files to infect based on a checksum of the ASCII code of the characters in the extension. Some other file extensions, such as MEM, PIF, QLB, OLD, DWG, LOG and TBL, have the same checksum, so these files can be corrupted too.

Anticmos

Aliases: CMOS Killer, Lennart.

Type: Master boot sector infector.

Resident: Yes.

Stealth: No.

Trigger: Random.

Payload: Erases all CMOS configuration information on the computer. This includes information on the type of floppy and hard disk drives installed, so the machine is rendered unbootable.

Comments: Anticmos is a typical boot sector virus. It infects the master boot sector of hard disks and the boot sector of floppy disks. The original floppy boot sector or master boot sector is not preserved on infection.

There is a variant, Anticmos Lixi (alias: Anticmos B), which contains the text string 'I am Li Xibin!'

Bupt

Aliases: Bupt9146, BuptBoot, WelcomB.

Type: Master boot sector infector.

Resident: Yes.

Stealth: No.

Trigger: None.

Payload: None.

Comments: Bupt is, as far as the user is concerned, just another master boot sector virus. It infects the hard drive in the standard manner (when a computer is inadvertently booted from an infected floppy disk), and infects floppy disks used in the computer from that point on.

Bupt will not work on XTs.

Cascade

Aliases: 1701, Fall, Hailstorm, Russian,

Type: COM file infector.

Resident: Yes.

Stealth: No.

Trigger: Original version: between 1st October and 31st December 1988. Formatting version: between 1st October and 31st December in any year except 1993.

Payload: Original version: causes characters to ‘fall’ from the screen. Formatting version: formats sector zero of the hard disk.

Comments: The ‘falling characters’ payload of the original Cascade virus is perhaps one of the most well known virus side-effects.

CMOS4

Aliases: AntiEXE, D3.

Type: Master boot sector infector.

Resident: Yes.

Stealth: Yes. Its stealth consists of hiding the real boot sectors.

Trigger: A 3 in 256 chance for each data read.

Payload: The virus examines the data being read to see if it is the start of an EXE file. If it is, and certain conditions regarding the size and nature of that program are met, the virus corrupts the data. A corrupted .EXE file would not execute and would not copy cleanly. However, to our knowledge, a program meeting the conditions has never been found, so the payload is for practical purposes harmless.

Comments: CMOS4 is a boot sector virus with simple stealth. It infects the master boot sector of hard disks and the boot sector of floppy disks.

Dark Avenger

Aliases: Eddie.

Type: COM and EXE file infector.

Resident: Yes.

Stealth: Yes.

Trigger: No.

Payload: Data sectors on the hard disk are overwritten with garbage.

Comments: The author of the virus claimed that it was the first virus in Bulgaria. The virus contains the text ‘Eddie lives...somewhere in time!’ and ‘Diana P.’ It only infects a program if it is at least 1775 bytes long, which may be an attempt to avoid detection.

Excel/Laroux

Aliases: None known.

Type: Excel macro virus.

Resident: Yes, within Excel environment.

Stealth: Partial - it hides in hidden module sheet.

Trigger: None.

Payload: None.

Comments: The first macro virus to infect Microsoft Excel files (.XLS workbooks), Laroux appeared in mid 1996.

It is a simple virus, similar to the first Word macro viruses, and contains two macros, auto_open and check_files. The auto_open macro is run when the infected document is opened, and merely instructs Excel to call the check_files macro every time a new worksheet is activated.

When this happens, the virus creates a file in the XLSTART directory called PERSONAL.XLS and copies the viral macros into it. This file is automatically opened every time Excel is run, much like Word's NORMAL.DOT. From then on, it infects every workbook used.

When PERSONAL.XLS is infected, the virus will be loaded every time Excel is started.

Excel/Sofa

Aliases: None known.

Type: Excel macro virus.

Resident: Yes, within Excel environment.

Stealth: Yes.

Trigger: None.

Payload: None.

Comments: Excel/Sofa is a virus that infects Microsoft Excel workbooks. Unlike the earlier Excel/Laroux, Sofa hides its presence in an infected environment.

On installation, the virus copies itself to the file BOOK.XLT, and changes Excel's configuration so that BOOK.XLT is loaded at start-up. If BOOK.XLT does not exist, the virus displays the message

> Microsoft Excel has detected a corrupted
> add-in file.
> Click 'OK' to repair this file.

After infection, it then displays

> File successfully repaired!

Sofa is not a very reliable virus, and mostly fails to work, because it makes assumptions about Excel’s configuration that are generally not true.

When the virus first infects the system, it changes Excel’s name in the window’s title bar from ‘Microsoft Excel’ to ‘Microsofa Excel’, hence the name.

The active viral code is stored in a hidden module sheet, and a copy of the viral source code is also stored in a hidden worksheet. The copy of the source is used to create new infections rather than directly copying the module sheet.

Form

Aliases: None known.

Type: Partition boot sector infector.

Resident: Yes.

Stealth: No.

Trigger: 18th of every month.

Payload: Produces a click every time a key is pressed.

Comments: Form originated from Switzerland and has been one of the most common viruses for a number of years. Unlike most boot viruses, it infects the boot sector of the active partition instead of the master boot sector.

On infection, the virus overwrites the partition boot sector with its own code, saving the original partition and a sector of its own code in the last two sectors of the partition. Form does not allocate or otherwise protect these sectors in any way, but since the FAT file system allocates sectors from the beginning of the disk, they will not be overwritten unless the disk fills up completely.

Unfortunately, Form assumes that the active partition is a DOS FAT partition. This is fatal under non-DOS operating systems - if the sectors at the end of the partition are written to, the computer will not start. Windows NT appears particularly vulnerable to this problem, and a Form infection on NT can cause major problems.

If you find Form on an NT machine, **do not switch it off**. The system will most likely not restart, and it will be difficult to restore. Contact qualified anti-virus technical support immediately.

Under DOS (and 16-bit Windows) however, Form exists inconspicuously, and this is largely why it is so widespread.

There are several variants with minor differences.

Hare

Aliases: Krsna, HDEuthanasia.

Type: Multipartite polymorphic virus.

Resident: Yes.

Stealth: Yes.

Trigger: 22nd August & 22nd September.

Payload: Prints the message ‘"HDEuthanasia" by Demon Emperor: Hare Krsna, hare, hare...’ and attempts to overwrite all hard drives on the system with garbage.

Comments: Hare is 7-8Kb in size on disk and 9Kb in memory, making it one of the largest non-Windows viruses that has been in the wild. It is also one of the most complicated viruses a user is likely to encounter.

Helloween

Aliases: None known.

Type: COM and EXE file infector.

Resident: Yes.

Stealth: No.

Trigger: 1st November.

Payload: Displays the message in Czech: ‘Nesedte porad u pocitace a zkuste jednou delat neco rozumneho! ******************* !! Poslouchejte HELLOWEEN#-#nejlepsi metalovou skupinu !!’ This translates as: ‘Don’t just sit there, do something reasonable! Listen to HELLOWEEN, the best heavy metal band!!’. The virus then waits for a keypress, and reboots the PC.

Comments: This virus targets other executable files as well as COM and EXE. It has been reported to infect BIN, OVR and certain SYS files. A primitive attempt to avoid detection is made by specifically not infecting COMMAND.COM and files with names that suggest they might be anti-virus programs. The message for the payload is encrypted within the virus code to make it more difficult to find.

JackRipper

Aliases: Jack the Ripper, Ripper.

Type: Master boot sector infector.

Resident: Yes.

Stealth: Yes.

Trigger: A 1 in 1024 chance for every sector write.

Payload: Swaps two words (four bytes) around. The virus takes care to avoid damaging its own sectors in this way.

Comments: The most notable thing about JackRipper is its payload, which is both subtle and destructive. This type of damage, sometimes referred to as 'data diddling', is extremely dangerous. If the virus goes undetected, it is impossible to determine which data on the PC has been modified by the virus, or indeed how much has been modified!

Its other functionality is fairly standard for a boot sector virus, although it does keep most of its boot sector code encrypted in a primitive attempt to fool scanners.

Jerusalem

Aliases: Friday the 13th, Israeli, PLO.

Type: COM and EXE file infector.

Resident: Yes.

Stealth: No.

Trigger: Non-destructive: after the system has been infected for 30 minutes. Destructive: Friday 13th, as long as the year is not 1987.

Payload: Non-destructive: row 5 column 5 to row 16 column 16 on the screen are scrolled up two lines, creating a 'black window', then the system slows down due to a time-wasting loop installed on each timer interrupt. Destructive: every program run will be deleted.

Comments: One of the first known viruses. This virus has gained a lot of notoriety in the press due to its trigger date and its destructive payload. The infection technique is quite primitive. It is unable to recognize an infected EXE file and will continue infecting the same file over and over again. The size of infected EXE files are increased by 1808 bytes on each infection. COM files are infected only once and the virus avoids the COMMAND.COM file.

Joshi

Aliases: None known.

Type: Master boot sector infector.

Resident: Yes.

Stealth: Yes.

Trigger: 5th January.

Payload: Displays the message ‘Type "Happy Birthday Joshi"’ on a cyan background. The message will remain on the screen until the user either enters ‘Happy Birthday Joshi’ or switches the machine off and boots from a clean disk.

Comments: The virus originated in India. It will stay in memory after Ctrl-Alt-Del. When a floppy disk is infected the virus mimics the text of the original boot sector so a quick scan by a disk utility will not reveal the virus. It also uses 6Kb of an infected PC’s memory.

Junkie

Aliases: DrWhite.

Type: Multipartite (infects COM files, MBR, boot sector of floppy disks).

Resident: Yes.

Stealth: No.

Trigger: None.

Payload: None.

Comments: Junkie is a reasonably simple multipartite virus that infects COM files, MBRs and floppy boot sectors. The virus only becomes resident when started from a boot sector; the COM form acts only as a dropper for the MBR form, doing nothing else.

On hard disk, the virus' two extra sectors of code are stored on the first track; on floppy, they are located at the end of the disk, and (being unprotected) may be overwritten by data. The code is encrypted in a very basic way.

Junkie infects COM files either when they are opened or when they are executed. If a virus scanner does not find Junkie in memory, it could infect all COM files on the system as it scans. However, since Junkie is quite old, this is no longer a real problem.

There are a number of variants. Junkie 1308, unlike its more common ancestor, is dangerous; it may format the hard disk.

Michelangelo

Aliases: None known.

Type: Master boot sector infector.

Resident: Yes.

Stealth: Yes.

Trigger: 6th March.

Payload: On hard disks it overwrites the first 17 sectors on every track of the hard disk, heads 0 to 4. On 360K floppy disks it destroys sectors 1 to 9, heads 0 and 1. On other floppy disks it destroys the first 17 sectors of each track.

Comments: Michelangelo is a very well-known virus. The name was given to it by a researcher who noticed that its trigger date was the same as Michelangelo's birthday, 6th March, although there is no evidence to suggest that this was the virus writer's reason for choosing that date.

The virus was based on the code of the New Zealand virus.

Monkey

Aliases: None known.

Type: Master boot sector infector.

Resident: Yes.

Stealth: Yes.

Trigger: None.

Payload: None.

Comments: Monkey is a simple boot sector virus. If an infected floppy disk is used to boot the machine, Monkey infects the Master Boot Record (MBR) on the first hard disk.

Monkey is notable because it scrambles the partition table and its own code sectors on the hard disk, making the partitions on it inaccessible from a clean floppy boot. Overwriting the MBR code gets rid of the virus itself, but does not undo the scrambling of the partition table, so data is still inaccessible.

To recover from Monkey, use an anti-virus product which is capable of disinfecting it, such as Sophos Anti-Virus.

There is a very similar variant called Monkey 2.

New Zealand

Aliases: Marijuana, Stoned.

Type: Master boot sector infector.

Resident: Yes.

Stealth: Yes.

Trigger: Once in 8 times, if booted from a floppy disk.

Payload: Displays the message 'Your PC is now Stoned!'

Comments: This was the first master boot sector virus. It was written by a student in Wellington, New Zealand. He claimed that the virus was never meant to go into the wild, but his only copy of it was stolen and propagated.

As well as displaying the message 'Your PC is now Stoned!' it contains the text 'LEGALISE MARIJUANA!' It can cause damage to floppy disks larger than 360K.

NoInt

Aliases: Bloomington, Stoned III, Stoned-b2.

Type: Master boot sector infector.

Resident: Yes.

Stealth: Yes.

Trigger: None.

Payload: None.

Comments: NoInt is another variant of the New Zealand virus. It has no remarkable properties and no specific payload.

Parity Boot

Aliases: Parity, Parity Check.

Type: Partition boot sector infector.

Resident: Yes.

Stealth: Yes.

Trigger: Random.

Payload: Displays the words 'PARITY CHECK' and hangs the computer. This mimics a genuine memory error, which leads many victims to believe that their RAM is at fault.

Comments: Parity Boot is a reasonably common and simple boot sector virus. It is spread on the boot sectors of floppy disks, and infects hard disks when an infected floppy is used to boot the machine. After this, most floppy disks that are used by the machine will be infected.

There are a number of variants, all of which differ little.

Spanish Telecom

Aliases: Campana, Spanish Trojan, Telefonica.

Type: Multipartite (infects COM and EXE files, MBR, boot sector of floppy disks) polymorphic.

Resident: Yes.

Stealth: Yes.

Trigger: 400th infected boot cycle.

Payload: Overwrites all data on the first two fixed disks and displays the message 'Campana Anti-TELEFONICA (Barcelona)'.

Comments: This was one of the first multipartite viruses. The file infector part contains the boot sector code and will infect the boot sector as well as files. This does not happen in reverse however. The parasitic code is 3,700 bytes. Its first 85 bytes contain 'armoured' code to detect debugging software and several randomised instructions designed to make a search for the virus byte pattern difficult.

It contains the text 'Virus Anti - C.T.N.E. (c)1990 Grupo Holokausto. Kampanya Anti - Telefonica. Menos tarifas y mas servicios. Programmed in Barcelona (Spain). 23-8-90. -666-'. The message translates to 'Lower telephone tariffs, more service'.

Tequila

Aliases: None known.

Type: Multipartite (infects EXE files, MBR, boot sector of floppy disks).

Resident: Yes.

Stealth: Yes.

Trigger: Dependent on the date and number of files infected.

Payload: Displays a crude Mandelbrot (fractal) set on screen. The virus then prompts the user to execute an Int 21H function which displays a text message giving the name T. Tequila and a Swiss P.O. Box number. The messages continues with the text: 'Loving thoughts to L.I.N.D.A. BEER and TEQUILA forever !'

Comments: The virus contains encrypted text 'Welcome to T.TEQUILA's latest production', 'Contact T.TEQUILA/P.o.Box 543/6312 St'hausen/ Switzerland'.

It hides itself in the boot sector by preserving the original messages and partition information.

Vacsina

Aliases: TP04, TP04VIR.

Type: COM and EXE file infector.

Resident: Yes.

Stealth: No.

Trigger: As each file is infected.

Payload: The PC beeps.

Comments: Only files more than 1,212 bytes long (the length of the virus) will be infected. One of a family of viruses written in Bulgaria by an author known as TP.

Viresc

Aliases: 2KB, French Boot, Jumper.

Type: Master boot sector infector.

Resident: Yes.

Stealth: No.

Trigger: Random.

Payload: Locks the computer on boot-up, and continuously displays a single character (ASCII code 238) on the screen.

Comments: Viresc infects in the standard manner - that is to say, when the user boots from an infected floppy disk, the virus makes the transition to the hard disk, and infects floppy disks used in the machine from that point on.

Its most notable feature is the rather peculiar way it handles floppy infection - it appears to specifically target the FORMAT command in an obscure manner, but the technique works well enough for the virus to prosper in the real world.

V-Sign

Aliases: None known.

Type: Master boot sector infector.

Resident: Yes.

Stealth: No.

Trigger: After infecting 64 floppy disks.

Payload: Displays a large V on the screen.

Comments: V-Sign is another simple boot sector virus, spread on the boot sectors of floppy disks, and infecting the master boot sector of a hard disk when the system is booted from an infected floppy.

The code of many viruses contains bugs, and V-Sign has a bad one - the virus fails to detect that a system is already infected, due to a programming error, and proceeds to infect the system again. This wipes out the saved copy of the original Master Boot Record (MBR) that the virus saves, and thus makes the system unbootable. Since the original MBR code has been lost, it cannot be put back; disinfection must consist of placing a standard MBR back, instead.

Win95/Boza

Aliases: None known.

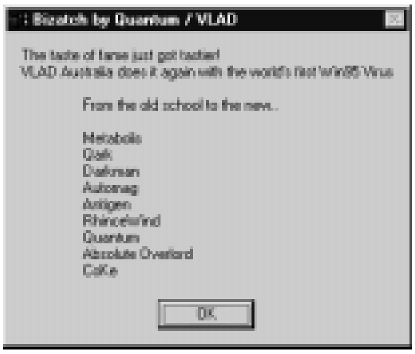
Type: Direct-action PE infector.

Resident: No.

Stealth: No.

Trigger: 31st of any month.

Payload: Displays the following dialog box:



Bizatch by Quantum / VLAD

The taste of fame just got tastier!
VLAD Australia does it again with the world's first Win95 Virus

From the old school to the new..

Metabolis
Qark
Darkman
Automag
Antigen
RhinceWind
Quantum
Absolute Overlord
CoKe

Comments: This virus achieved some notoriety by being the first virus to infect PE files (PE is the EXE format used by 32-bit Windows programs under Windows 95 and Windows NT). It is a very simple virus, similar in its operation to the most basic DOS file viruses. It does not go resident, but is a direct action infector; when an infected program is run, it immediately searches out up to three uninfected executables and infects them.

Winword/Cap

Aliases: None known.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: Yes. Empty macros are used to prevent Word showing menu items. For example, the ToolsMacro (or ExtrasMakro under German Word) is empty, which prevents the use of the ToolsMacro to see whether or not there are macros present. The virus also removes the menu item itself so that it does not even appear in the list of available choices. Winword/Cap also turns off the prompt to save NORMAL.DOT.

Trigger: None.

Payload: None.

Comments: The Winword/Cap virus installs the following macros: FileTemplates, ToolsMacro, FileSaveAs, FileClose, AutoClose, FileSave, FileOpen, AutoOpen, AutoExec and CAP. In addition, the virus will find the current local language version of the macros and will install these as well as the English ones. For example, if the virus infects a German version of Word, it will also install macros named DateiÖffnen, DateiSpeichern, DateiSpeichernUnter, DateiSchließenOderAllesSchließen.

With the exception of the CAP macro itself, all the macros are very short stubs which either call subroutines within CAP or do nothing at all.

Winword/Concept

Aliases: Prank Macro.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: No.

Trigger: None.

Payload: There is a macro called PayLoad, which contains the text ‘That’s enough to prove my point’, but this is never triggered.

Comments: Winword/Concept was the first virus written for the Microsoft Word environment found ‘in the wild’. Today, it is one of the most commonly encountered viruses.

The virus takes control thanks to its AutoOpen macro; macros of this name are automatically run by Word if they are present in a document. It also contains a FileSaveAs macro which is invoked when Word is saving a document; this contains the infection code.

There are already a number of Winword/Concept variants, thanks to the ease of modifying an existing Word macro virus to do new things.

Winword/DMV

Aliases: None known.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: None.

Trigger: Whenever a document is infected.

Payload: Displays a message box containing the text ‘Macro virus has been spread. Now execute some other code (good, bad, or indifferent).’

Comments: Winword/DMV came to the attention of anti-virus researchers some time after Winword/Concept, although it was written in December 1994, making it the first known Word macro virus written.

According to the comments in the code, it was written ‘to reveal a significant security risk in software that supports macro languages with auto-loading capabilities’.

Unlike Winword/Concept and Winword/Nuclear, Winword/DMV is self-contained in a single macro which is automatically invoked by Word when an infected document is closed. Since the viral macro is triggered both when you close the document explicitly yourself and when you close it implicitly by exiting Word, there is no obvious escape from the virus once you have opened an infected file.

By way of justifying itself as a ‘research’ virus, Winword/DMV pops up dialog boxes each step of the way as it executes. On closing an infected file, the virus infects NORMAL.DOT (the default global macro template); on closing an uninfected file once NORMAL.DOT is infected, the virus infects that file. This infection strategy is simple and effective.

The last dialog box that appears simply notes that the virus has replicated, and remarks that a ‘good, bad, or indifferent’ payload could be executed at that point.

DMV stands for Document Macro Virus.

Winword/Imposter

Aliases: None known.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: No.

Trigger: None.

Payload: None.

Comments: This is a very simple Word macro virus, similar to Winword/Concept.

It is chiefly remarkable in that it masquerades as an earlier Word virus, Winword/DMV; when it infects, it pops up a dialog box containing the one word 'DMV'. Its name is derived from this behaviour.

Winword/MDMA

Aliases: MDMA-DMV.

Type: MS Word macro virus.

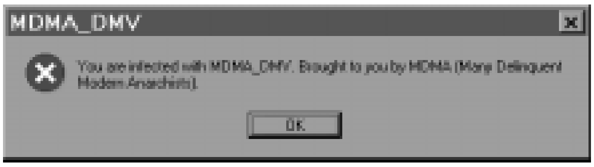
Resident: Yes, within Word environment.

Stealth: No.

Trigger: 1st of every month.

Payload: On Windows 95 systems, Winword/MDMA deletes all CPL and HLP files (Control Panel and Help files respectively) from the C:\WINDOWS directory (if Windows is not installed in this location, this will clearly have no effect), turns off login script processing, and turns on the StickyKeys and HighContrast options from the Accessibility settings.

The following dialog box is also displayed:



You are infected with MDMA_DMV. Brought to you by MDMA (Many Delinquent Modern Anarchists).

Comments: Winword/MDMA goes to great lengths to determine the operating system the current Word session is running on. It attempts to distinguish between Windows 3.1, Windows 95, Windows NT, and Macintosh. There are several errors in this routine, the final one of which results in the virus always believing it is running on a Windows 95 system. The only relevant payload, therefore, is that for Windows 95 systems.

MDMA first appeared in the wild in June 1996.

Winword/Npad

Aliases: None known.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: No.

Trigger: A counter is placed in the WIN.INI file. Whenever a file is opened within Word, the counter is incremented. When it reaches 23, the payload is executed and the counter reset to 0.

Payload: Displays a scrolling message in Word's status bar reading 'D0EUNPAD94, v.2.21, (c) Maret 1996, Bandung, Indonesia'.

Comments: Winword/Npad is a normal macro virus. It infects in the standard way, and is protected by the standard Word macro 'encryption' system.

Winword/Nuclear

Aliases: None known.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: No.

Trigger: There are three different triggers for three different payloads. Only one works, as described below.

Payload: The first time Word is started after infection at a time between 5pm and 5:59pm, a macro is run which attempts to infect the system with a DOS virus, Ph33r. Fortunately, the macro is corrupt and the attempt fails.

There is also a macro, called PayLoad, which on the 5th of April attempts to destroy important system files (IO.SYS, MSDOS.SYS and COMMAND.COM).

This does not work fully either - only the IO.SYS file is destroyed.

The final trigger works. Each time a document is printed, there is a 1 in 12 chance that the virus will add the words:

> And finally I would like to say:
> STOP ALL FRENCH NUCLEAR TESTING IN THE PACIFIC

to the end of the document as it prints.

Comments: Winword/Nuclear is a destructive Word macro virus which is in the wild. It infects and spreads in the normal way - via an AutoOpen macro on infected documents which is run when the document is opened, infecting the NORMAL.DOT template.

It is fortunate that Nuclear's author did not have the equipment or the foolhardiness to test his creation - it could have been much worse.

Winword/ShareFun

Aliases: ShareTheFun, ShareF.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: Yes. Contains basic stealth routines which prevent the use of the Tools/Macro or File/Templates menu options to look for the virus in infected files.

Trigger: Once in every four times that an infected document is opened.

Payload: On a machine where Microsoft Mail is running, the virus attempts to use Microsoft Mail to send itself to three randomly selected recipients from the user's list of correspondents. If the virus succeeds in propagating itself by email, recipients will see a message with the subject line 'You have GOT to read this!' Within the message is an infected attachment named DOC1.DOC.

On a machine where Microsoft Mail is not running, the virus triggers a different payload: one in every four times an infected document is loaded, the virus simply bails out of Windows.

Comments: Winword/ShareFun is a macro virus which tries to accelerate its spread using email. Despite its warhead, ShareFun does not work like the infamous Good Times hoax. Merely looking at a message transmitted by the virus is not enough to get infected. To spread, the mail attachment must be loaded into Word.

It is otherwise unremarkable, and straightforward to detect and disinfect.

Winword/Talon

Aliases: None known.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: Yes. To try to hide its presence, it subverts the Tools/Macro menu item so that, when selected, a dialogue reports: 'This Option Is Not Available, Please Insert The MS-Office CD And Install The Help Files To Continue'.

Trigger: 18th June.

Payload: The virus repeatedly displays the messages 'Your System Is Infected With The Macro Virus Talon #1' and 'This Macro Virus Was Brought To You By: TALON 1997'.

Comments: Winword/Talon was posted to the Internet newsgroup alt.comp.virus.

Winword/Wazzu

Aliases: None known.

Type: MS Word macro virus.

Resident: Yes, within Word environment.

Stealth: No.

Trigger: Every time a document is infected.

Payload: Between one and three words in the document are randomly moved to another location. Additionally, the word ‘wazzu’ may be inserted into the document at a random location.

The virus is particularly damaging because there can be no automated way of undoing the effects of the payload. The word ‘wazzu’ can be searched for and removed, but only careful manual checking can find and fix the moved words.

Comments: Winword/Wazzu is an insidiously damaging Word macro virus that is very common in the wild. Technically it is a very simple virus, spreading in the manner common to most Word macro viruses.

Wllop

Aliases: Sampo, Turbo.

Type: Master boot sector infector.

Resident: Yes.

Stealth: Yes.

Trigger: 30th November.

Payload: Displays a message referring to the University of East Manilla.

Comments: In many ways, Wllop is a typical master boot sector virus - it infects the hard disk when the computer is booted from an infected floppy disk. However, it does contain several non-standard features.

One unusual feature of Wllop is its ability to ‘fake’ the presence of Spanish Telecom - when a write-protected floppy disk is scanned with the virus active in memory, the calling program is returned an image of a Spanish Telecom-infected boot sector.

Also notable is Wllop’s behaviour when it infects a system that is already infected with any of four boot sector viruses of which it has specific knowledge. It is able to recognise and bypass these viruses, obtaining certain system information directly from the code of the other viruses!

The only other interesting point about Wllop is that it is able to survive a so-called ‘warm reboot’ (pressing Ctrl-Alt-Del).

WM97/NightShade

Aliases: None known.

Type: MS Word 97 macro virus.

Resident: Yes, within Word environment.

Stealth: No.

Trigger: There is a one in seven chance of a non-malicious payload being run every time the virus executes. A malicious payload is run if the date is Friday 13th.

Payload: Non-malicious payload: uses Word Assistant to proclaim ‘Attention: Word97.NightShade by Pyro [VBB]’. Malicious payload: password-protects the current document using the password ‘NightShade’.

Comments: This is a native Office 97 virus which spreads via the AutoClose macro, in a similar way to Winword/DMV, which is automatically invoked whenever a document is closed.

Yankee

Aliases: TPxx, Vacsina.

Type: COM and EXE file infector.

Resident: Yes.

Stealth: No.

Trigger: 5pm, or when Ctrl-Alt-Del is pressed, or a certain time after the virus has gone resident.

Payload: Will play the tune 'Yankee Doodle Dandy'.

Comments: A family of such viruses exist. The Yankee variant appears to be a more advanced version of Vacsina, which plays a tune in place of a beep.

One variant contains code to recognize and deactivate the Italian (Ping-Pong) virus.

50
40

Trojan horses, logic bombs and worms

Apart from viruses, there are three forms of software attack on computer systems:

- Trojan horses.
- Logic bombs.
- Worms.

Trojan horses

A Trojan horse is a program which performs functions other than those stated in its specifications. These functions can be (and often are) malicious.

An example of a Trojan horse is the program *ARC513* which pretends to be an improved version of the legitimate compression utility *ARC*. In reality, it deletes the files specified for compression.

Trojan horses are often used as a means of infecting an unsuspecting user with a virus. If a legitimate program becomes infected with a virus, it becomes a Trojan horse and if a user executes it in the belief that they are executing a *bona fide* copy, their computer will become infected.

Extortion via a Trojan horse

In 1989, twenty thousand floppy disks marked 'AIDS Information Version 2.00' were mailed out in London. The package posed as a legitimate program giving information on the biological AIDS virus and assessing the user's risk group.

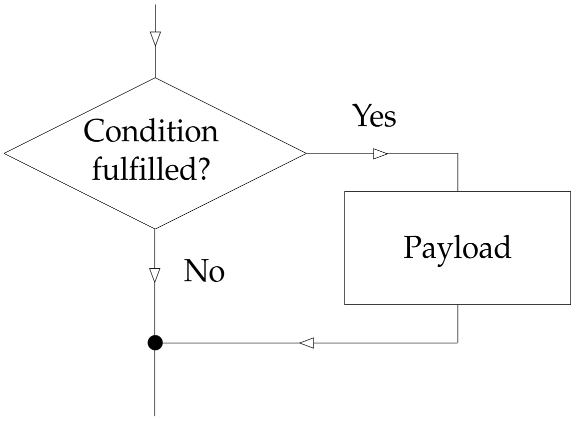
On the reverse of the instruction leaflet, in very small print, was a 'License Agreement' which requested payment for using the software. Once the package was installed, the program printed an 'invoice' giving the address to which payment should be sent.

The installation procedure also made modifications to the AUTOEXEC.BAT file, with the effect that after it had been executed around 90 times, the side-effects were activated. Most of the names of the files on the hard disk were encrypted and marked 'Hidden'. The only non-hidden file contained the blackmail message offering a repair program on receipt of a licence fee.

The perpetrator of this Trojan horse was extradited from the USA to England and put on trial for this attempted extortion.

Logic bombs

A logic bomb is a simple test which releases a damage routine when triggered by some condition such as time, or the presence or absence of data such as a name.



Logic bomb program flow

A hypothetical example would be a maliciously modified copy of a spreadsheet which zeroed a particular cell every Tuesday between 10 and 11 a.m., but otherwise did not reveal its presence. The results would be very confusing and difficult to trace.

Several examples of (malicious) logic bombs have been documented. In one case, a programmer writing a payroll package ‘ensured’ his continuing employment by introducing instructions that would delete files if his name was removed from the payroll. He was fired, and the logic bomb triggered its destruction routine. Only after being offered reinstatement did he agree to point out the logic bomb in the code. In another, similar case, the programmer was not reinstated but prosecuted.

Logic bombs in viruses

Logic bombs are often found in viruses, where the payload is triggered when a certain condition is met. For example, the *Form* virus executes its payload only on the 18th of each month, the *Winword/Npad* virus displays a message at every twenty-fourth file access in Word, and the *Italian* virus puts a bouncing ball on the screen only if a disk access is made during a 1-second interval in every 30 minutes.

The delay before the logic bomb triggers allows the virus to spread extensively before its side-effects reveal its presence.

Worms

Worms are rogue programs similar to viruses, but do not need a carrier in order to replicate. They replicate in their entirety, creating exact copies of themselves.

Worms are normally found on computer networks and multi-user computers, and use inter-computer or inter-user communications as the transmission medium.

Christmas tree worm on VM/CMS

Probably the best-known mainframe worm was the *Christmas tree worm* which paralysed the IBM worldwide network on 11th December 1987.

It was written in REXX and spread on VM/CMS installations. The program was a combination of a Trojan horse and a chain letter. It invited the user to type 'CHRISTMAS', drew a Christmas tree on user terminals and sent itself to all the user's correspondents in the user files NAMES and NETLOG.

UNIX Internet worm

A number of worm attacks have occurred on UNIX systems.

The most widely reported was the Internet worm which struck the US DARPA Internet computer network on 2nd November 1988. The worm was released by a Cornell University student. It replicated by exploiting a number of bugs in the UNIX operating system running on VAX and Sun Microsystems hardware, including a bug in *sendmail* (an electronic mail program) and in *fingerd* (a program for obtaining details of logged in users).

Stanford University, Massachusetts Institute of Technology, the University of Maryland and Berkeley University were infected within 5 hours of the worm being released. The NASA Research Institute at Ames and the Lawrence Livermore National Laboratory were also infected. The UK was unaffected.

The worm consisted of some 4,000 lines of C code and once it was decompiled, the specialists distributed bug fixes to *sendmail* and *fingerd*, which prevented further spreading. From the decompilation, it appears that the worm was not malicious. It did, however, cause overloading of infected systems.

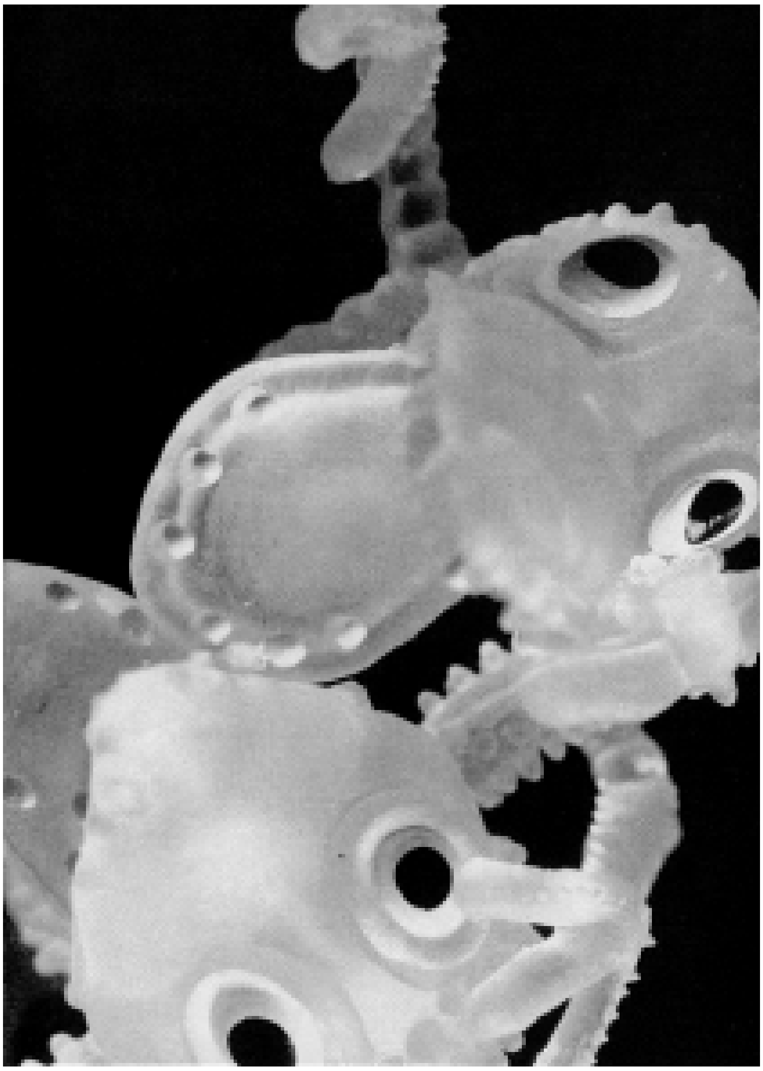
SPAN worm on VAX/VMS

On 16th October 1989, VAX/VMS computers on the SPAN network were attacked by a worm. The worm propagated via DECnet protocols and if it discovered that it was running with system privileges, changed the system announcement message to 'WORMS AGAINST NUCLEAR KILLERS'. An abbreviated form of this message was then presented in graphics, followed by the text 'You talk of times of peace for all, and then prepare for war.'

The worm also changed the DECnet account password to a random string and mailed information on the password to the user GEMPAK on SPAN node 6.59. If the worm had system privileges, it disabled mail to the SYSTEM account and modified the system login command procedure to *appear* to delete all files (it did not actually do it). The worm then proceeded to access other systems by picking node numbers at random and used the PHONE command to get a list of active users on the remote system. After accessing the RIGHTSLIST file, it attempted to access the remote system using the list of users found, to which it added a list of 81 standard users coded into the worm. It penetrated accounts where passwords were the same as the name of the account or were null.

The worm then looked for an account which had access to SYSUAF.DAT. If such an account was found, the worm copied itself to that account and started executing.

Within a very short time, the Computer Emergency Response Team (CERT) in the USA
(Tel +1 412 268 7090) issued a warning and a corrective response.



Virus hoaxes and scares

There have been many erroneous or alarmist virus reports in recent years. These fall into three categories:

- Hoaxes.
- Scares.
- Misunderstandings.

Hoaxes

These are false virus reports.

Such reports play on users' fears and/or lack of knowledge (e.g. by describing alarming but impossible side-effects). They can have nuisance value and lead to loss of working time.

By far the commonest form is the hoax email virus.

Hoax email viruses

Typically, PC users receive a warning about an email that, if read, can delete data or format the hard drive. The user is urged to forward the warning to other users, thus propagating the hoax.

Such reports are untrue. It is impossible for a virus to be transmitted in the normal text portion of an email. Viruses can be carried in attached files, but these must be detached and executed or opened before the virus can spread (see 'Email' in the 'Virus infiltration routes and methods' section of the 'How viruses spread' chapter).

Particularly widespread and tenacious examples of hoax email viruses include *Good Times* and *Join the Crew*.

Scares

Scares are reports that may have their origin in fact or informed speculation, but are distorted or greatly exaggerated.

Warnings about 'Java viruses' on the Internet are a recent example. No Java virus currently exists and the consensus is that viruses in the true sense of the word are impossible in Java. The reports are based purely on speculation about whether viruses are possible or about the threat from non-viral forms of software attack (see also 'Misunderstandings').

Another scare involved a Trojan horse passing itself off as a popular shareware program. In May 1995, PKZip warned that a fake version of their compression program was being circulated. This included a program, PKINST.EXE, which, if run, attempted to format the C: drive and delete all the files on this drive. Bugs in the code made this Trojan completely harmless, but the warning was sufficient to produce widespread hysteria.

Misunderstandings

Real computing problems are sometimes mistakenly attributed to computer viruses.

The media's tendency to call any software problem a virus often underlies such misunderstandings.

The 'millennium bug', for example, has often been attributed to a virus. Although a potentially serious problem, this is nothing to do with a virus. See 'The millennium bug' chapter for further details.

Another example was a reported ‘virus’ that advanced a PC’s clock by 100 years. Although there are viruses that can change the system date, the cause in this case was a bug in the computer’s BIOS that advanced the clock erroneously under very specific circumstances.

3

Anti-virus rules for employees

To help avoid infection

- Never accept disks or programs for use on your PC without first having them virus checked.
- Do not open Word or Excel documents of unknown origin, or other documents which can contain macro viruses, with the application that created them. Open the document with a viewer program which does not execute any macros that might be contained in the document.
- Never use game programs, demo disks, or other software of potentially doubtful origins on your ‘work’ PC. If you want to run such software, ask to use a ‘dirty PC’ provided for this purpose.
- Never copy software to give to other people. This can help spread a virus, and is usually illegal.
- If you lend disks to anyone else, have them checked before you use them on your computer again.
- Never leave floppy disks in a PC longer than necessary. Before you switch a PC on, check that the floppy drive is empty.
- If you ever boot from a floppy disk, only an approved write-protected ‘boot disk’ should be used.
- If in doubt, ask your PC support desk for advice.

The importance of preventing virus infection

A virus attack could threaten the company's prosperity and your job. Your organisation may use anti-virus software, but remember :

PREVENTION IS BETTER THAN CURE!

If a virus infection is suspected

If you suspect that there is a virus on your PC, for example because of strange behaviour or a warning from an anti-virus program, do as follows:

- Inform the PC support desk immediately.
- Do not let anyone use your PC. Put a sign on it to this effect.
- Gather together all disks which may have been used on the PC, so that they can be checked. Make sure no-one uses them.
- Alert your colleagues: the problem may not just be on your machine.

Even if you think you may have caused the problem, do not try to hide it. This only makes things worse.

If computers go wrong, you should not automatically assume that a virus is responsible. Amongst the strange occurrences which have been reported as a virus are: a printer running out of paper, floppy disk drives not being fully closed, screen monitors not connected or switched off, and so on.

You should be aware of the possibility of a virus attack, but provided you follow the anti-virus rules, the likelihood of it is comparatively small.

Sophos Data Security products 3

SOPHOS
www.sophos.com

Introduction to Sophos Data Security

Software for disk authorisation, encryption, and the secure erasure of files.

Sophos Data Security products

Sophos software is available for:

- Disk authorisation.
- Encryption.
- Secure erasure.
- Virus detection.

For an introduction to the virus detection software, see the 'Introduction to Sophos Anti-Virus' chapter.

Disk authorisation

Sophos produces a disk authorisation system called D-FENCE. There are two versions, D-FENCE and D-FENCE 4.

D-FENCE acts as a security guard, controlling the software that enters or exits from an organisation and ensuring that it is checked for viruses.

Disks pass through a gateway PC where they are converted from the standard DOS format to the company format, or vice versa. Unconverted disks are not readable on company PCs (and company disks are not readable on non-company PCs, if this option is chosen). During conversion the disks can be scanned with anti-virus software.

D-FENCE 4 includes all these features, together with high-security encryption of hard and floppy disks. It is aimed at protecting information on portables and laptops. Separate versions of D-FENCE 4 are available for commercial and UK Government use.

Encryption

Sophos offers several products for the encryption of data, i.e. for scrambling it so that it can not be read by an unauthorised person.

E-DES and **PUBLIC** are stand-alone file encryption applications which provide secure file storage and transmission for a wide variety of operating systems, including DOS, OpenVMS and UNIX. E-DES encrypts files using DES or SPA (Sophos Proprietary Algorithm). PUBLIC encrypts files using RSA or MDH in combination with DES or SPA.

Encryption toolkits are available for various algorithms including DES, RSA, SPA and MDH. The Severn Bridge algorithm is also available for UK Government customers. The toolkits allow the encryption and decryption of data to internationally recognised standards on a wide range of platforms (DOS, UNIX, NetWare, OpenVMS, Windows NT, OS/2 and MVS). They provide a simple API which can be used with most programming languages.

All the toolkits are compatible, so that data encrypted on one platform can easily be decrypted on another. They are delivered as object modules and are incorporated into applications at link time.

Secure erasure

Sophos has two products for erasing data irretrievably, **SHRED** and **PURGE**. SHRED erases files irretrievably by multiple overwrites. Similarly, PURGE erases complete disks.

Training and education

Sophos holds practical workshops on security and virus detection at its training centre in Abingdon, England, or in-house for individual organisations. Sophos also publish technical reports on security and virus issues, which are available free of charge.

Further information

Details of all the above products and services can be found in the chapters that follow.

D-FENCE

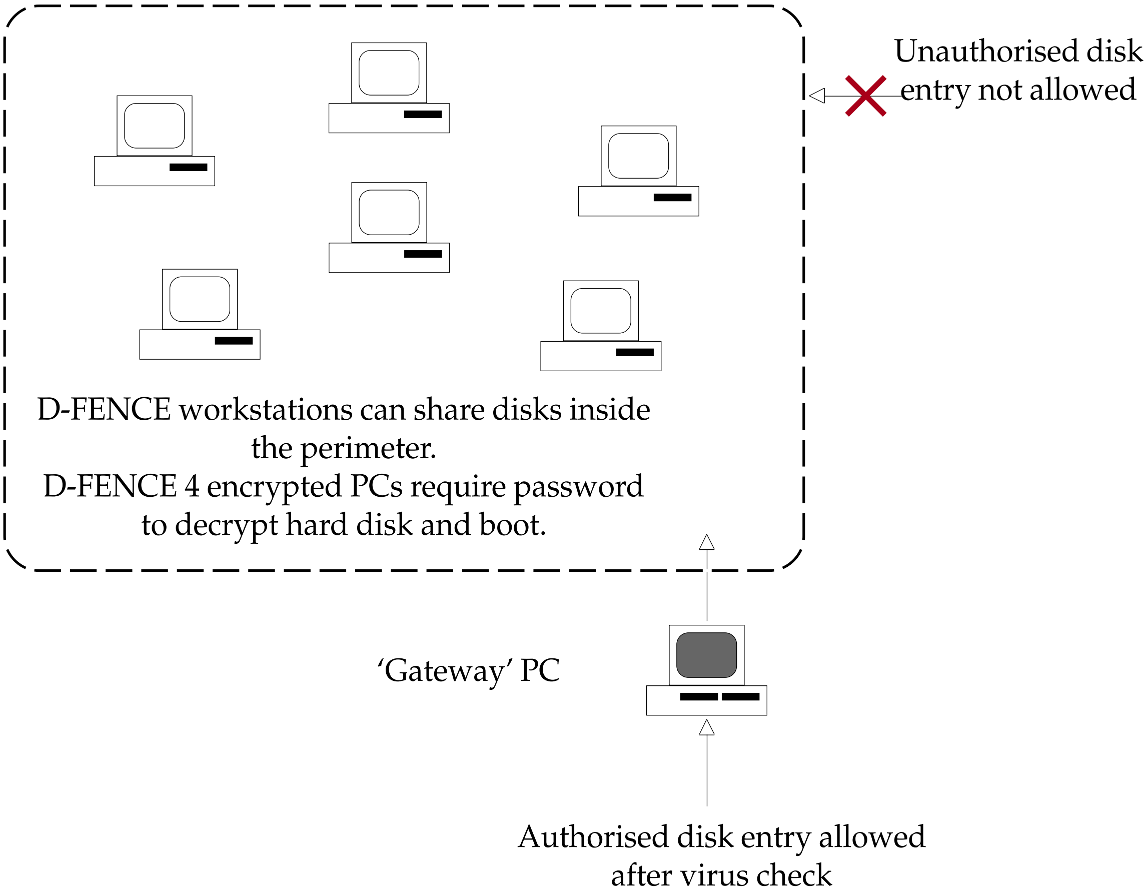
Disk authorisation system

Prevents the use of unauthorised
floppy disks on IBM-PCs
and compatibles.

Description

D-FENCE versions 3 and 4 act as a security guard which prevents the use of unauthorised floppy disks inside a designated group. Thus the software that enters an organisation can be strictly controlled, and the data within an organisation can be securely protected.

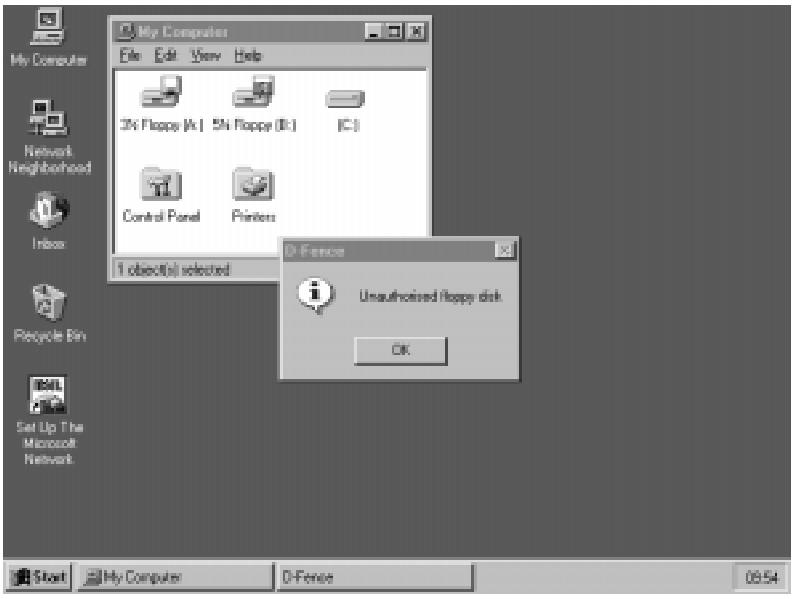
D-FENCE version 4 also offers the option to encrypt all data on hard and floppy disks. If a disk is completely encrypted, it is impossible to inspect the data on it without the correct passwords. This is useful, for example, to ensure the confidentiality of the data on the hard disk should the PC become lost or be stolen. For commercial users of D-FENCE version 4, encryption is performed using SPA (Sophos Proprietary Algorithm) together with a boot-up


Unauthorised disk entry not allowed
D-FENCE workstations can share disks inside the perimeter.
D-FENCE 4 encrypted PCs require password to decrypt hard disk and boot.
'Gateway' PC
Authorised disk entry allowed after virus check

D-FENCE protecting a group of workstations password designated by the system administrator. For UK Government (HMG) customers, encryption uses the Severn Bridge algorithm with CLEARVIEW passwords for boot-up. This scheme preserves the basis of groups: everyone in a group has their hard disk encrypted with the same key, whilst all users have different boot-up passwords. This enables the system administrator to provide new passwords without having to decrypt and re-encrypt the data, and to remove the decryption without access to individual users' passwords.

D-FENCE modes of operation

D-FENCE version 3 has two modes of operation: **Import Control mode** and **Import/Export Control mode**. D-FENCE version 4 has three additional modes: **Encrypt hard disks and use DOS floppy disks**, **Encrypt hard disks and use Import/Export Control floppy disks**, and **Encrypt hard and floppy disks**.



D-Fence

Unauthorised floppy disk

Unauthorised disk message

In **Import Control mode**:

- The importation of disks is controlled. Any incoming disk must first pass through a ‘gateway’ PC, which converts the disk from standard DOS format to D-FENCE format. Unconverted disks cannot be read on PCs within the D-FENCE group.
- The exportation of disks is **not** controlled. Converted disks can be read on non D-FENCE PCs, although if altered they have to be reauthorised on the gateway PC before they can be used on a D-FENCE PC again.

In **Import/Export Control mode**:

- The importation of disks is controlled as in Import Control mode.
- The exportation of disks is also controlled and converted disks have to be deauthorised on a gateway PC before being readable on normal PCs.

In **Encrypt hard disks and use DOS floppy disks mode** (D-FENCE 4 only):

- The PC requires a password (D-FENCE 4 SPA) or 2 passwords (D-FENCE 4 HMG) to decrypt the hard disk and boot. This is useful, e.g. to ensure the confidentiality of the data on the hard disk should the PC be lost or stolen.
- Ordinary DOS floppy disks can be read from and written to.

In **Encrypt hard disks and use Import/Export Control floppy disks mode** (D-FENCE 4 only):

- The PC requires a password (D-FENCE 4 SPA) or 2 passwords (D-FENCE 4 HMG) to boot.
- Floppy disks have to be imported and exported as per Import/Export Control mode.

In **Encrypt hard and floppy disks mode** (D-FENCE 4 only):

- The PC requires a password (D-FENCE 4 SPA) or 2 passwords (D-FENCE 4 HMG) to boot.
- Floppy disks have to be imported and exported as per Import/Export Control mode.
- The data on floppy disks is encrypted, providing an extra level of security.

D-FENCE installation

The memory-resident D-FENCE code has to be installed on all the PCs within a D-FENCE group, and on a typical PC this process is quite simple. It takes about a minute for non-encrypting modes, and depends on the size and speed of the hard disk for D-FENCE 4 disk encrypting modes. Once installed on a PC, D-FENCE does not interfere with its normal operation and is completely transparent to the user.

Gateway PCs (where used) contain all of the D-FENCE installation software, but do not run the memory-resident part of D-FENCE. This means that they can read and write normal DOS disks. The gateway PC can be configured to check all DOS disks for viruses with Sophos Anti-Virus automatically, before converting them to D-FENCE format. The gateway PC can also create bootable D-FENCE system disks, which can be used to clean boot D-FENCE PCs.

Features

D-FENCE version 3 and version 4:

- Are simple to install, use, customise and maintain.
- Have low memory and processing overheads.
- Are robust in use.

- Allow the erased areas of floppy disks to be purged when they are being imported or exported at the gateway PC. This prevents data being smuggled into and out of organisations via erased files.
- Prevent programs, such as disk backup programs, bypassing D-FENCE by direct hardware access.
- Allow floppy disks to be virus-checked from within D-FENCE.

Applications

- Prevent the use of unauthorised floppy disks.
- Help to prevent accidental data leaks.
- Enable the formation of closed PC user groups inside one organisation.
- Enforce security and (optionally) virus checking of incoming and/or outgoing floppy disks.

Technical details

Product names: D-FENCE Versions 3.14 (May 1998) and 4.11 (October 1997).

Function: D-FENCE 3; prevents the use of unauthorised floppy disks.

D-FENCE 4; encryption and authentication of floppy and hard disks.

Modes of operation: Import Control mode (Versions 3 and 4); Encryption of the partition table on fixed disks, and checksum of root directory and FAT stored on authorised floppy disks.

Import/Export Control mode (Versions 3 and 4); Encryption of the partition table on fixed disks, and encryption of FAT and root directory on authorised floppy disks.

Encrypt hard disks and use DOS floppy disks mode (Version 4).

Encrypt hard disks and use Import/Export Control floppy disks mode (Version 4).

Encrypt hard disk and floppy disks mode (Version 4).

Memory overhead: 3 to 5 Kb.

Execution overhead: D-FENCE 3; negligible.

D-FENCE 4 approximately 0.1% to 10%.

Compatibility: D-FENCE is compatible with Windows 3.x, Windows 95, Windows NT, OS/2 and DOS networks such as Novell NetWare and Digital Equipment's PATHWORKS.

3CX
C·R CONSTRUCTION.

DES Toolkit

Relocatable object module

A relocatable object module which can be linked with applications programs, to provide encryption and decryption of data according to the Data Encryption Standard (DES) on a wide variety of platforms.

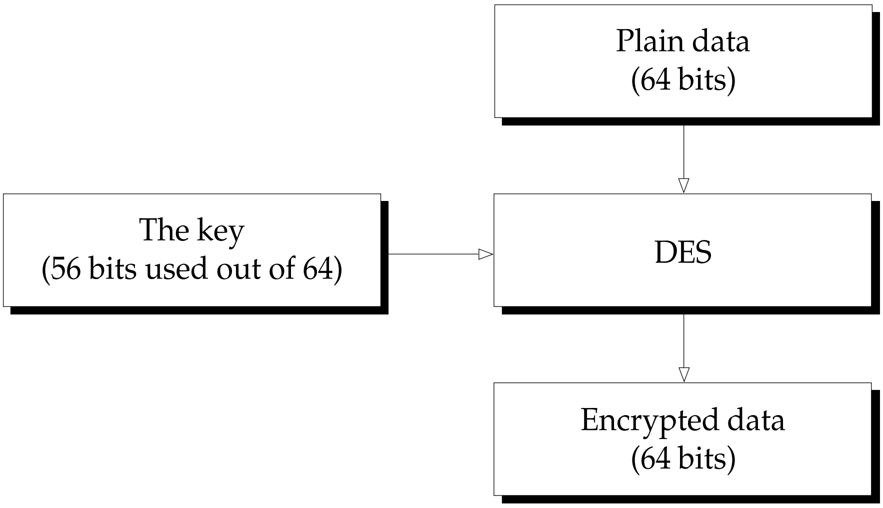
Description

Simple encryption methods can be cracked easily using a computer and are inadequate for confidential information. The **US Data Encryption Standard (DES)** is a sophisticated algorithm which encrypts and decrypts 64-bit blocks of data on the basis of a 56-bit key. DES has been in wide use since 1977 and has proved to be reliably secure.

The DES Toolkit consists of a single relocatable object module, which can be linked into application programs. It contains eight functions providing all the tools needed for encryption and decryption of data according to the US Data Encryption Standard.

Applications written in virtually any language such as **Fortran**, **Pascal**, **C** or **Cobol** can use the encryption facilities provided by the Toolkit. The DES Toolkit is simply included in the application at link time.

Encryption according to DES transforms a 64-bit plaintext block into a 64-bit ciphertext block. For the purposes of the Toolkit, these are blocks of 8 consecutive 8-bit bytes.



Encryption using DES

DES uses 56 bits of a 64-bit key for encryption. The key is a block of 8 consecutive 8-bit bytes, of which 8 bits are ignored by DES.

The DES Toolkit is proven, tested, self-contained, requires no hardware, and is available immediately.

Functions

init Before encryption or decryption can be performed, and whenever a different key is to be used, the initialisation function **init** must be called.

isweak A known and documented property of DES is that certain keys can give rise to poor encryption. Out of the approximately 70,000,000,000,000,000 unique 56-bit keys, 4 are weak and 12 are semi-weak. The function **isweak** allows the calling program to test a key for weakness before it is used.

encrypt and **decrypt** Once **init** has been called, all subsequent encryption or decryption can be performed using the functions **encrypt** and **decrypt**. The re-initialisation is only required before using a different key.

cbcenc and **cbcdec** When several blocks of data are encrypted in succession using the same key, normal DES encryption will produce an identical cipher block for each identical plaintext block. Repetitions in the plaintext can thus cause repetitions in the ciphertext, leading to a potential weakness in the defence against cryptanalytical attack. A recommended solution is to use Cipher Block Chaining (CBC), which makes each cipher block dependent not only on the current plaintext block and the key, but also on all previous plaintext blocks. The functions **cbcenc** and **cbcdec** automatically provide CBC for encryption and decryption.

wipe The function **wipe** is provided to erase any sensitive information internal to the Toolkit before program termination.

inbits To facilitate compatibility with other DES implementations, the function **inbits** can be used to reorder the bits in a key or data block.

Example

A simple example (in C) using some of the functions provided in the Toolkit follows.

The example shows the basic use of the functions **init**, **encrypt** and **decrypt**. The output from the program in the example is:

```
11 22 33 44 55 66 77 88
85 e1 c7 f0 42 bd ea d2
11 22 33 44 55 66 77 88
```

```
main()
{
      static unsigned char key[]={ 0x01, 0x23,
             0x45, 0x67, 0x89, 0xab, 0xcd, 0xef };
      static unsigned char data[]={ 0x11, 0x22,
             0x33, 0x44, 0x55, 0x66, 0x77, 0x88 };

      init(key); /* initialise DES */
      print(data);
      encrypt(data); /* encrypt 8 bytes */
      print(data);
      decrypt(data); /* decrypt 8 bytes */
      print(data);
}


void print(unsigned char s[])
{
      int i;

      for(i=0;i<8;i++) printf("%02x ",s[i]&0xff);
      printf("\n");
}
```

Applications

- Encryption of individual fields in database software.
- Professional security for financial application programs.

- Custom security software.
- Incorporation of encryption into existing programs.

Technical details

Software name: DESTOOL.OBJ Version 1.05 (March 1995).

Function: Encryption of data according to the US Data Encryption Standard (DES).

Encryption method: DES in Electronic Code Book mode (ECB) or Cipher Block Chaining (CBC) mode.

Key checking: Weak and semi-weak DES key detection.

MS-DOS: Typical speed (standard version); 18.8 Kb/sec on 100MHz 486 PC.

Typical speed (fast version); 166 Kb/sec on 100MHz 486 PC.

Compatibility with various compilers supported. Size; 10K.

OpenVMS: Psect $CODE with attributes CON, LCL, SHR, EXE, NOWRT. Alignment; byte. Psect $DATA with attributes CON, LCL, NOSHR, NOEXE, WRT. Alignment; longword. Arguments passed by reference. Size; 20 Blocks.

IBM MVS: Adheres to MVS calling standard. Arguments passed by reference. Size; 15K.

IBM AS/400: Adheres to AS/400 calling standard. Arguments passed by reference. Size; 37K.

Sun SPARC: Adheres to SunOS calling standard. Arguments passed by reference. Size; 12.6K.

DESTEST

DES testing package

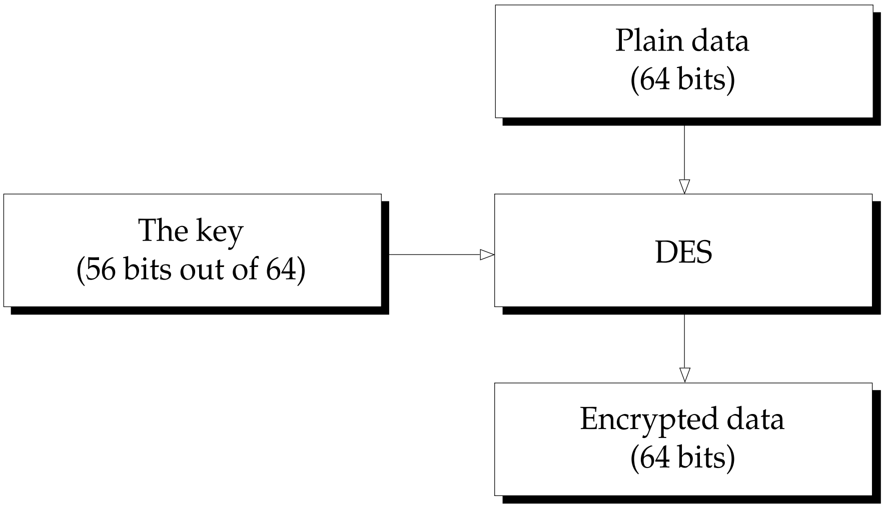
A package for verifying hardware and
software implementations of the
Data Encryption Standard (DES).

Description

DESTEST is a flexible tool which allows the testing of hardware DES encryption devices as well as software implementations of DES for compliance with the ANSI X3.106 standard. It is command line driven, so that it can be incorporated into batch files or command files for automatic execution.

Encryption performed according to DES transforms a 64-bit plaintext block into a 64-bit ciphertext block. For the purposes of DESTEST, these are blocks of 8 consecutive 8-bit bytes.

DESTEST supports nine modes of operation, as specified in ANSI standard X3.106: Electronic Codebook (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), Cipher Feedback (a) 5-bit (CFB5), Cipher Feedback (a) 6-bit (CFB6), Cipher Feedback (a) 7-bit (CFB7), Cipher Feedback (a) 8-bit (CFB8), Cipher Feedback (a) 64-bit (CFB64) and Output Feedback (OFB) (a).



Encryption using DES

Input, output and keys can be represented in a choice of formats. The DES standards do not specify a mapping between their numbering of bits in a 64-bit block and the order of bits in a string of 8 bytes, and so DESTEST supports a choice of bit-numbering conventions.

It is also possible to specify repeated encryption of each 64-bit block, which is useful for comprehensive testing of new DES implementations.

Modes of operation

ECB **ELECTRONIC CODE BOOK:** In Electronic Codebook mode, each 64-bit block of data is encrypted and produces 64 bits of encrypted data. Identical blocks of data produce identical blocks of encrypted data. ECB is the default mode of operation of DESTEST.

CBC **CIPHER BLOCK CHAINING:** CBC is commonly used to eliminate undesirable repetitions in ciphertext, by making each ciphertext block dependent not only on the current plaintext block and the key, but also on all previous plaintext blocks. DESTEST provides command line qualifiers which invoke CBC mode for encryption and decryption.

CFB **CIPHER FEEDBACK:** In cipher feedback mode, data is divided into units of **k** bits each and encryption and decryption produce encrypted data **k** bits long. An initialisation vector of up to 64 bits is used. **k** must be in the range of 1 to 64. ANSI standard X3.106 also defines 5-bit CFB(a) operation, 6-bit CFB(a) operation, 7-bit CFB(a) operation, 8-bit CFB(a) operation and 64-bit CFB(a) operation. DESTEST provides command line qualifiers which invoke CFB and CFB(a) modes for encryption and decryption.

OFB **OUTPUT FEEDBACK:** In output feedback mode, data is divided into units of **k** bits each and encryption and decryption produce encrypted data **k**

bits long. An initialisation vector of up to 64 bits is used. **k** must be in the range of 1 to 64. DESTEST provides command line qualifiers which invoke OFB mode for encryption and decryption.

Input/Output formats

DESTEST allows the input file, output file and the key to be specified in one of three formats: hexadecimal, ASCII and binary. Formats can be mixed, so that, for example, the input file can be in hexadecimal, the output file in binary and the key in ASCII.

Hexadecimal format Hexadecimal format uses digits 0-9 and letters A-F (or a-f) to represent 4-bit combinations 0000 to 1111. Two hexadecimal digits are used to represent a byte.

ASCII format ASCII format uses digits 0-9 to represent the 4-bit combinations 0000 to 1001, as well as the ASCII characters : ; < = > and ? to represent 1010 to 1111. Two characters are used to represent one byte.

Binary format Binary format takes each character as an 8-bit value, reading until the true end-of-file.

Applications

- Testing of DES hardware.
- Testing of DES software.
- High speed file encryption.
- Testing of DES speed on particular machines.
- Developing or optimising DES implementations.

Technical details

Product name: DESTEST Version 1.01 (March 1995).

Function: Testing of DES software and hardware implementations.

Mode of operation: Command line driven.

Modes of encryption: Electronic Codebook (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), Cipher Feedback (a) 5-bit (CFB5), Cipher Feedback (a) 6-bit (CFB6), Cipher Feedback (a) 7-bit (CFB7), Cipher Feedback (a) 8-bit (CFB8), Cipher Feedback (a) 64-bit (CFB64), Output Feedback (a).

Input/output file format: Hexadecimal, ASCII or binary.

E-DES for DOS

File encryption package for DOS

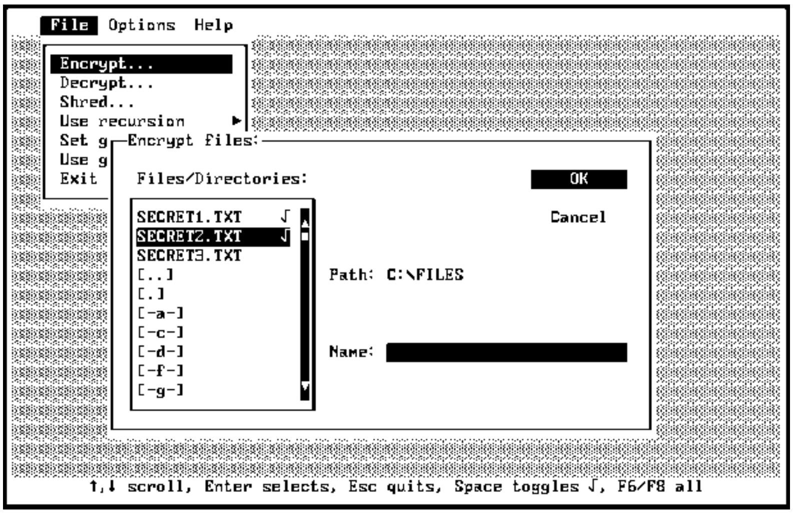
An encryption package for secure storage
of confidential information in DOS.

Description

E-DES for DOS is an encryption package for secure storage of confidential data and programs. An encrypted file contains its data in a secure 'scrambled' form. Encrypted files can be examined, but their contents are unintelligible and unusable until they are decrypted. Any type of file can be encrypted, such as a word-processed document, a spreadsheet file or a program file.

Encryption is performed on the basis of a user-defined key which can be in the form of a word, phrase or number. The file can only be decrypted if the correct key is supplied, so the security of the encrypted file is dependent upon the confidentiality of the key.

E-DES also allows for the secure erasure (shredding) of unwanted but sensitive files. Unlike ordinary file deletion, E-DES not only removes the directory entry for the specified files, but also overwrites those parts of the floppy or hard disk on which the contents of the file were stored. The contents of the shredded file are erased positively and irretrievably, and neither standard system utilities nor special retrieval programs can reconstruct the file.



```
SECRET1.TXT  √
SECRET2.TXT  √
SECRET3.TXT
[..]
[.]
[-a-]
[-c-]
[-d-]
[-f-]
[-g-]

Path: C:\FILES

Name:

↑,↓ scroll, Enter selects, Esc quits, Space toggles √, F6/F8 all
```

E-DES for DOS file specification

Features

E-DES for DOS:

- Is fully compatible with E-DES for Windows. Files encrypted with one can be decrypted with the other, and vice versa.
- Can use one of two encryption algorithms: a high speed software implementation of the US Data Encryption Standard (DES) or the Sophos Proprietary Algorithm (SPA).
- Can save encrypted files in ASCII format for transmission by email.
- Is an interactive program with comprehensive on-line help.
- Offers protection against errors by the user. On encryption, the key is checked for typing mistakes. On decryption, special checksum information constructed by E-DES allows automatic detection of incorrect keys. Files cannot be decrypted using the wrong key.
- Offers the shredding (secure erasure) of files to one of three security levels.
- Always cleans up on termination, by shredding unwanted data.
- By using a temporary file for all operations, E-DES allows encryption and decryption to be aborted at any moment without danger of leaving a partially processed file. Even in the case of a power failure during encryption, the original file will not be corrupted.
- Offers data compression as an option. This means that encrypted files are usually much smaller than the original.
- Accepts command line qualifiers and can work in batch mode. This allows complicated procedures to be carried out on a regular and reliable basis. This feature is of particular value to large corporations,

which often need to standardise their security procedures in a rigorous way.

Applications

- Safe storage of personal data on computer media.
- Protection of financial, technical and management information.
- Security of files and programs on multi-user systems.
- Compliance with Data Protection Act 1984.
- Emergency protection for communicated data.
- Secure erasure of unwanted but sensitive data.

Technical details

Product name: E-DES for DOS Version 4.03 (April 1998).

Function: File encryption package.

Mode of operation: Interactive and/or command line driven, with extensive on-line help.

Encryption method: DES or SPA encryption of data using Cipher Block Chaining (CBC) to eliminate repetitions in the encrypted text.

Compression method: Ross Data Compression algorithm.

Key specification: Natural language or hexadecimal. Keys can be up to 80 characters. 16-digit hexadecimal keys are automatically recognised and used directly; all others are compressed into a 56-bit key for DES, or a 64-bit key for SPA. No distinction is made between upper and lower case characters. All non-alphabetic and non-digit characters are ignored.

Shredding levels: ‘Quick’ (1 overwrite), ‘Government’ (3 overwrites + verify), and ‘Military’ (7 overwrites + verify).

Speed of DES encryption, compressed binary output: 101 Kb/sec for 133MHz Pentium Digital Celebris GI5133ST.

Typical SPA encrypted file size decrease with compression enabled: Binary format; 35% of original 143K text file.

ASCII format; 60% of original 143K text file.

Speed of SPA encryption, compressed binary output: 356 Kb/sec for 133MHz Pentium Digital Celebris GI5133ST.

Input file format: Any file type. File size restricted only by the space available on disk.

Encrypted file format: Binary or printable ASCII. Checksums on individual records and overall file for error detection.

ABBA

E-DES for Windows

File encryption package for Windows

An encryption package for secure storage
of confidential information in Windows.

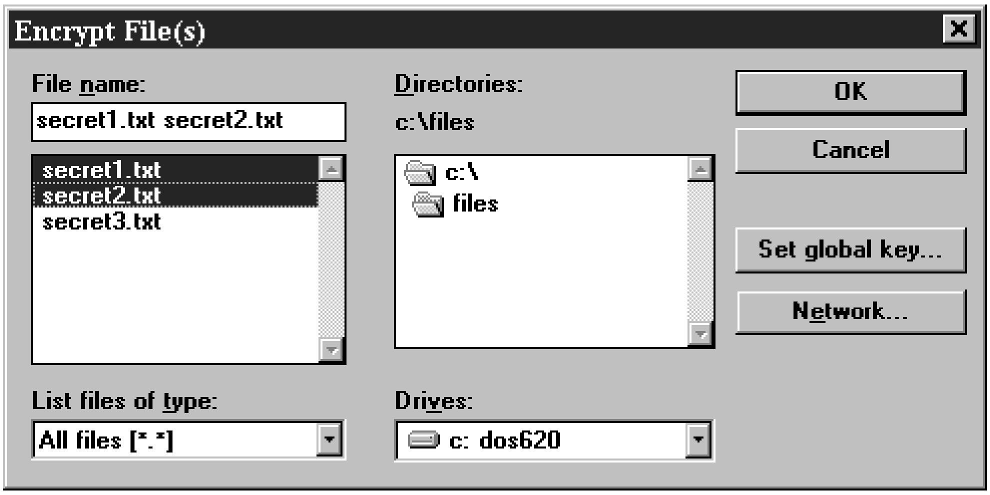
Description

E-DES for Windows is an encryption package for secure storage of confidential data and programs.

An encrypted file contains its data in a secure ‘scrambled’ form. Encrypted files can be examined, but their contents are unintelligible and unusable until they are decrypted. Any type of file can be encrypted, such as a word-processed document, a spreadsheet file or a program file.

Encryption is performed on the basis of a user-defined key which can be in the form of a word, phrase or number. The file can only be decrypted if the correct key is supplied, so the security of the encrypted file is dependent upon the confidentiality of the key.

E-DES also allows for the secure erasure (shredding) of unwanted but sensitive files. Unlike ordinary file deletion, E-DES not only removes the directory entry for the specified files, but also overwrites those parts of the floppy or hard disk on which the contents of the file were stored. The contents of the shredded file are erased positively and irretrievably, and neither standard system utilities nor special retrieval programs can reconstruct it.



Encrypt File(s)

File name: secret1.txt secret2.txt

secret1.txt
secret2.txt
secret3.txt

Directories: c:\files

c:\
files

List files of type: All files [*.*]

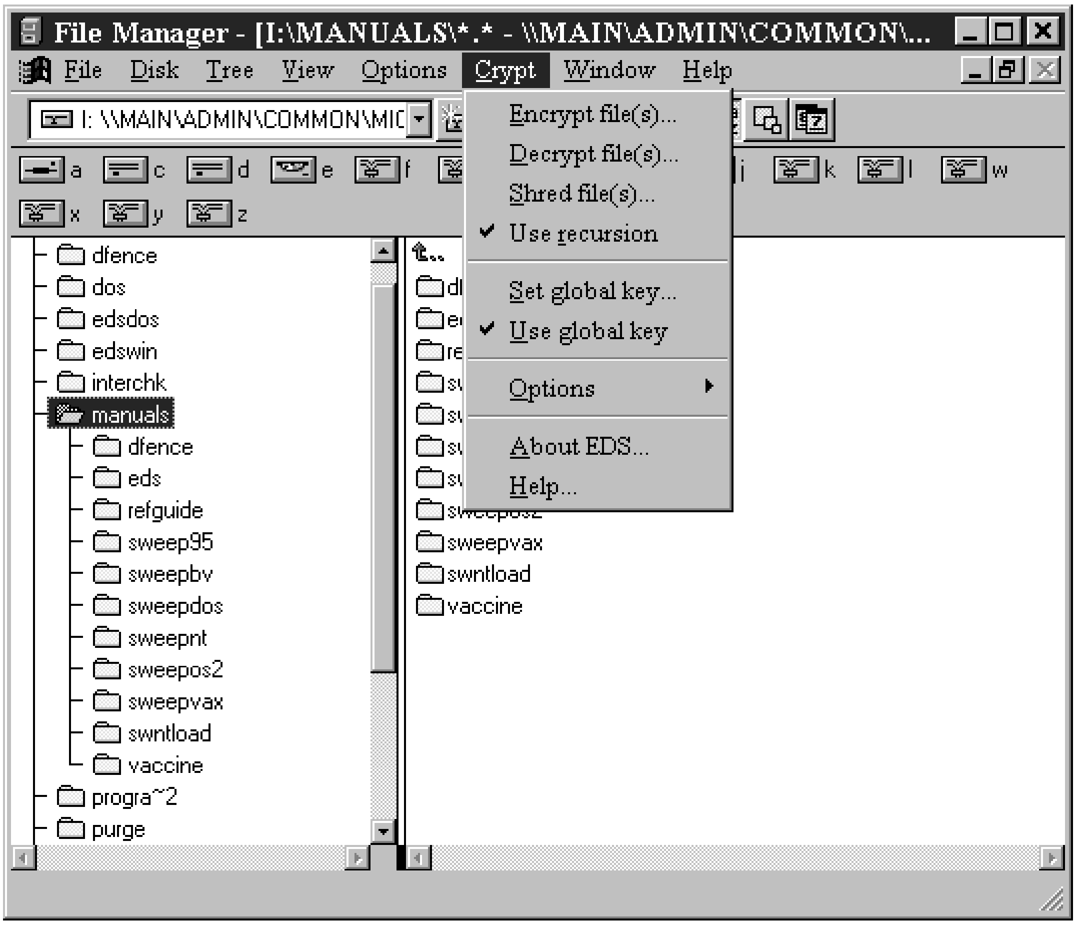
Drives: c: dos620

E-DES for Windows file specification

Features

E-DES for Windows:

- Is fully compatible with E-DES for DOS. Files encrypted with one can be decrypted with the other, and vice versa.
- Can use one of two encryption algorithms: a high speed software implementation of the US Data Encryption Standard (DES) or the Sophos Proprietary Algorithm (SPA).
- Can save encrypted files in ASCII format for transmission by email.



E-DES for Windows used from File Manager

- Is an interactive program with comprehensive on-line help.
- Offers protection against errors by the user. On encryption, the key is checked for typing mistakes. On decryption, special checksum information constructed by E-DES allows automatic detection of incorrect keys. Files cannot be decrypted using the wrong key.
- Offers the shredding (secure erasure) of files to one of three security levels.
- Always cleans up on termination, by shredding unwanted data.
- By using a temporary file for all operations, E-DES allows encryption and decryption to be aborted at any moment without danger of leaving a partially processed file. Even in the case of a power failure during encryption, the original file will not be corrupted.
- Offers data compression as an option. This means that encrypted files are usually much smaller than the original.
- Can also add a custom menu to the Windows File Manager allowing the encryption, decryption and shredding of files/directories selected from the File Manager.

Applications

- Safe storage of personal data on computer media.
- Protection of financial, technical and management information.
- Security of files and programs on multi-user systems.
- Compliance with Data Protection Act 1984.
- Emergency protection for communicated data.
- Secure erasure of unwanted but sensitive data.

Technical details

Product name: E-DES for Windows Version 4.03 (April 1998).

Function: File encryption package.

Mode of operation: Interactive and/or from the Windows File Manager, with extensive on-line help.

Encryption method: DES or SPA encryption of data using Cipher Block Chaining (CBC) to eliminate repetitions in the encrypted text.

Compression method: Ross Data Compression algorithm.

Key specification: Natural language or hexadecimal. Keys can be up to 80 characters. 16-digit hexadecimal keys are automatically recognised and used directly; all others are compressed into a 56-bit key for DES, or a 64-bit key for SPA. No distinction is made between upper and lower case characters. All non-alphabetic and non-digit characters are ignored.

Shredding levels: 'Quick' (1 overwrite), 'Government' (3 overwrites + verify), and 'Military' (7 overwrites + verify).

Speed of DES encryption, compressed binary output: 237 Kb/sec for 133MHz Pentium Digital Celebris GI5133ST.

Speed of SPA encryption, compressed binary output: 285 Kb/sec for 133MHz Pentium Digital Celebris GI5133ST.

Input file format: Any file type. File size restricted only by the space available on disk.

Encrypted file format: Binary or printable ASCII. Checksums on individual records and overall file for error detection.

Typical SPA encrypted file size decrease with compression enabled: Binary format; 35% of original 143K text file.

ASCII format; 60% of original 143K text file.

PUBLIC

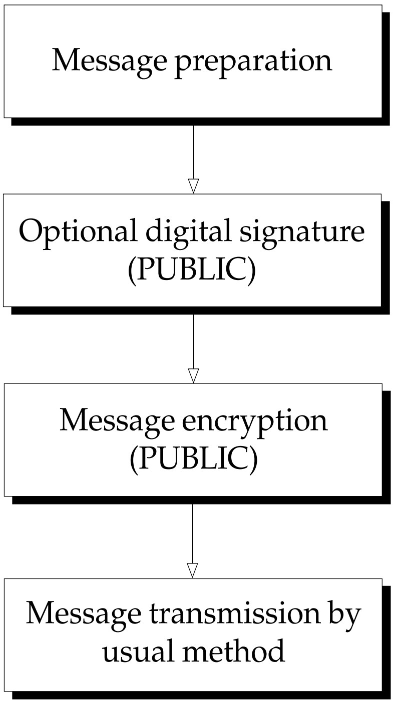
Communications security system

A system for security of information in transit.
Its features include public key encryption,
digital signature of files and comprehensive
key management utilities.

Description

PUBLIC is a package for security of communications in any group of two or more users who transfer information to each other. PUBLIC protects the information sent, using digital signatures and/or encryption. A typical application would be a bank or group of banks, using a commercially available electronic mail service for file transfers between their computers worldwide, and needing to secure the transfer.

PUBLIC is used by the sender of information to encrypt it before sending, and by the recipient to decrypt it after reception. It is used in conjunction with the existing method of data transfer, e.g. X.25 packet-switch system, telephone line, the postal system or a courier. It can be used when sending any kind of information: binary data files, text messages, spreadsheet files etc.

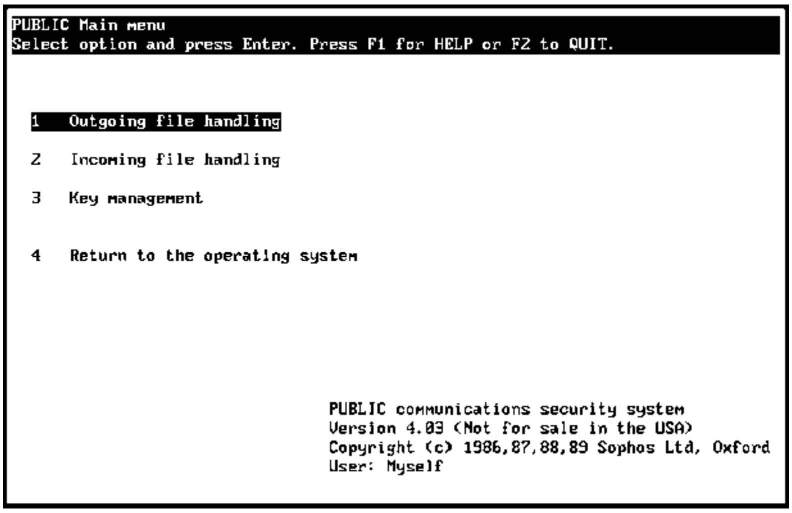

Message preparation
Optional digital signature (PUBLIC)
Message encryption (PUBLIC)
Message transmission by usual method

Message flow using PUBLIC

PUBLIC is based on the use of **different** keys for encryption and decryption. Each user has a pair of unique keys: a **private key** and a **public key**. The public key can be made freely available to all other users and is used for **encryption** of files. The private key, on the other hand, is used for **decryption** and is kept secret. It is protected cryptographically with a password chosen by the authorised user at the time the key was generated. All the user must do is remember their password; PUBLIC will ask for it whenever it has to perform any operation using the private key. Without the password, the private key is unusable.

This means that **anybody can encrypt information to send to a particular user, while only that user can decrypt it** (using their private key).

Every new user first chooses a password, which they must remember and keep secret, and then generates a unique key-pair. The uniqueness of his key is obtained by taking an extremely large random number from which the private and public parts of the key are then derived. The process of generating the key-pair is highly complex, but PUBLIC handles

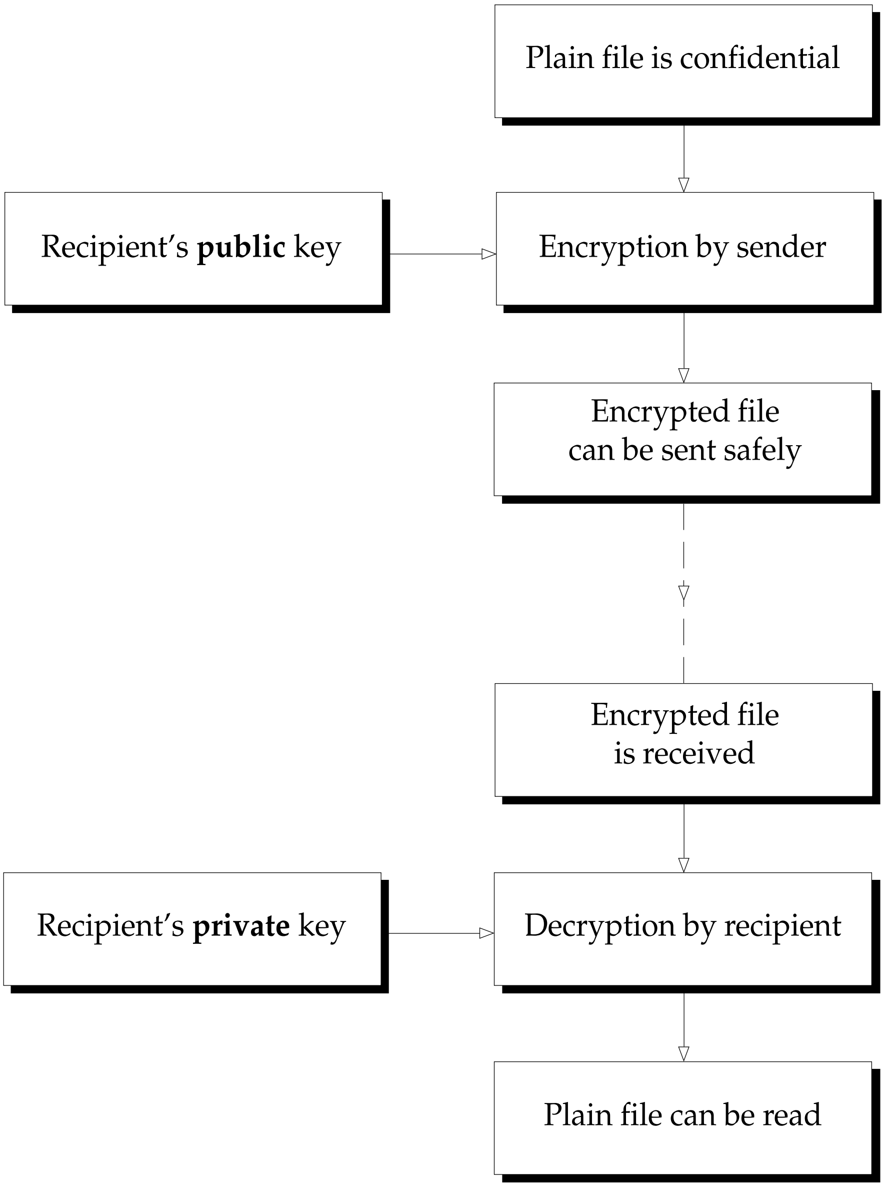


```
PUBLIC Main menu
Select option and press Enter. Press F1 for HELP or F2 to QUIT.


 1   Outgoing file handling

 2   Incoming file handling

 3   Key management


 4   Return to the operating system



                                PUBLIC communications security system
                                Version 4.03 (Not for sale in the USA)
                                Copyright (c) 1986,87,88,89 Sophos Ltd, Oxford
                                User: Myself
```

PUBLIC initial menu

this complexity internally: from the user's point of view, it is simple. **Key generation is only done once**, when the system is installed.

Once the key is generated, its public part is written out to a file which is distributed (usually by the network supervisor) to all other users. They will then be able to send encrypted information to the new user and to verify their digital signature.


Plain file is confidential
Recipient's **public** key
Encryption by sender
Encrypted file can be sent safely
Encrypted file is received
Recipient's **private** key
Decryption by recipient
Plain file can be read

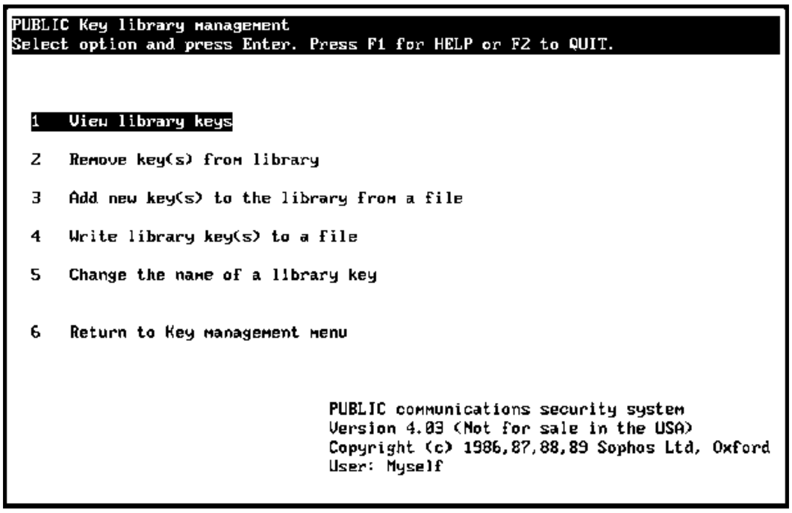
Public key encryption

The list of other users' public keys is held as a computer file and used automatically by PUBLIC. The list can be updated at any time if new users join the network, or if an existing user should decide for security purposes to generate a new pair of keys. All that the user must remember is their password; there is no need to remember their own key or anyone else's keys.

Features

For encryption of information, PUBLIC uses the **RSA** algorithm to encrypt and transmit randomly generated DES keys. A standard option in PUBLIC allows the **MDH** algorithm to be used instead of RSA, for example when greater security or very fast key generation are required. The information itself is then encrypted using DES. This hybrid approach gives all the benefits of public key encryption, while maintaining the higher data processing rate achievable with DES. The 512-bit public key algorithms give extremely high security.

PUBLIC provides all the functions needed for key library management. Those relating to the user's own



```
PUBLIC Key library management
Select option and press Enter. Press F1 for HELP or F2 to QUIT.


  1   View library keys

  2   Remove key(s) from library

  3   Add new key(s) to the library from a file

  4   Write library key(s) to a file

  5   Change the name of a library key


  6   Return to Key management menu


                                  PUBLIC communications security system
                                  Version 4.03 (Not for sale in the USA)
                                  Copyright (c) 1986,87,88,89 Sophos Ltd, Oxford
                                  User: Myself
```

PUBLIC library management menu

key are available by default, while using PUBLIC in 'supervisor' mode extends the menu structure to include the full range of key library management functions.

PUBLIC has comprehensive security features to prevent a variety of methods of cryptanalytical attack. A new, completely random DES key is generated for each encryption operation. The user's private key is stored encrypted, and the key library is protected cryptographically against tampering.

Using command line qualifiers, PUBLIC can be configured exactly as needed for integration with other packages and into batch files.

In the PUBLIC+ version, the package can provide additional security and convenience with a **hardware electronic token** which plugs into a simple adaptor connected to the PC or terminal. The tokens can be carried on a normal key-ring and offer facilities such as expiry dates, secure sharing of private keys and so on. PUBLIC+ is available on request only.

Customised non-DES versions of PUBLIC are also available for special applications.

Example

Original file:

```
To Great National Bank
Buy 20 kg of gold at 11:00 a.m.
Montague
```

File with digital signature (internal representation):

```
To Great National Bank
Buy 20 kg of gold at 11:00 a.m.
Montague
:Above message signed by Montague
:On 10/03/95 00:14:41
:Digital signature
IGRQGGKIGGIGGTHUSLSRLMTTORHPPUVRVPKJ
```

```
UHLJMGHLKNOIKSRRTQRRNSJMSPMOJKJGPMLK
OKHMKHKOPOVJRNNLISHHOUNRURRQJKRVJQPG
NQVIIOJTUOUTTJLKGQGMTTNNTGITTUVM
```

Encrypted, signed file

```
:Recipient Great National Bank
:Sender Montague
:Date/time encrypted 10/03/95 00:16:34
)hYT4v(&*>>?Kgyy67<pha!_nQj^&8tyg7G}
g‘+CpD[{q_tOvxBNW#_|Zs9&^j:L?+^Uh7kG
9=jEDK”<c5AJ[#hgfdFRdf%dfg”sg45lkj_(J
```

Applications

PUBLIC’s digital signature and encryption functions make it suitable for securing communications in new or existing networks.

A great benefit of PUBLIC to the system designer or integrator is its **compatibility with any network**. It is simple to install PUBLIC in all or part of an existing network without having to change hardware configurations.

PUBLIC can be used to secure wide area networks, X.400 messaging systems, or any electronic mail facility, by:

- Banks and insurance companies.
- Systems houses.
- Embassies and government departments.
- Police forces and military units.

Technical details

Product name: PUBLIC Version 4.04 (March 1995).

PUBLIC+ Version 4.04 using hardware tokens (March 1995).

Function: Public key system for digital signature and encryption in communications.

Mode of operation: Interactive, menu-driven, also drivable entirely from command line.

Encryption method: 512-bit RSA or MDH encryption of a DES key, followed by DES encryption of data using Cipher Block Chaining (CBC). SPA available instead of DES on application.

RSA or MDH key-pair generation: Random numbers produced by timing user keystrokes, or from Crypto-Jet if fitted, or from token if PUBLIC+. Extensive primality checks.

DES key generation: Random value produced for each encryption by timing user keystrokes, or from Crypto-Jet if fitted, or from token if PUBLIC+. Weak and semi-weak keys are rejected.

Digital signature: Using RSA or MDH keys in combination with one-way DES function of data.

Speed measurements, RSA+DES, no Crypto-Jet, including all disk access: Data encryption; 0.1 sec + 72 Kb/sec for 100MHz 486 Compaq DESKPRO XE 4100.

Data decryption; 1.6 sec + 72 Kb/sec for 100MHz 486 Compaq DESKPRO XE 4100.

Digital signature; 1.6 sec + 72 Kb/sec for 100MHz 486 Compaq DESKPRO XE 4100.

Digital signature check; 0.1 sec + 72 Kb/sec for 100MHz 486 Compaq DESKPRO XE 4100.

Key generation; 20 sec typical.

Input file format: Any file type. File size restricted only by the space available on disk.

Encrypted file format: Binary or printable ASCII. Checksums on individual records and overall file.

Encrypted file size increase: Binary format; 1.03 times larger. ASCII format; 1.41 times larger (typical figures).

Key library file storage: Printable ASCII protected with own digital signature, 330 bytes/entry.

Private key file storage: DES-encrypted using password chosen by user. PUBLIC+ uses combination of token and password. Encrypted private key is stored as printable ASCII.

Hardware tokens (PUBLIC+ only): PC interface; printer port piggy-back.

Minicomputer interface; RS-232 piggy-back.

Features; password expiry date, password reuse prevention, secure private key sharing, token expiry date, tamper protection.

chicco

PURGE

Positive disk erasure utility

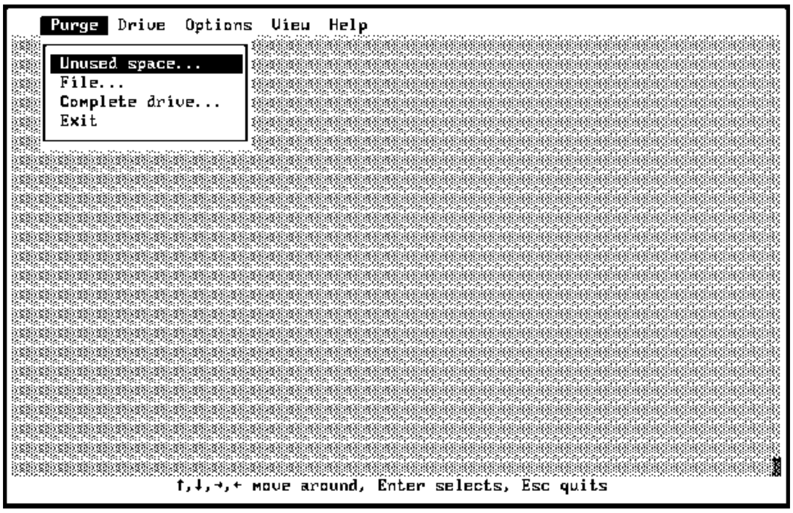
A utility for irrecoverable erasure of information from disks for PCs running DOS.

Description

Sensitive information left on unused disk sectors presents a common computer security risk. The DOS command DEL only removes the directory entry for the specified file, leaving the contents of the file intact. This information can be recovered, and even when normal undeleting utilities fail to work, it is usually still possible to examine the information on the disk.

When a **file** is selected for purging, it is erased positively and removed from the disk. Its name is also removed from the directory.

If purging of the **unused space** is selected, all temporary storage areas and any files which may have been deleted in the past will be erased positively and irretrievably. The configuration of the disk will be preserved, as will programs or data files that are still required. When erasing unused space, PURGE covers not only the un-allocated disk sectors but also all the unused areas which have been allocated to specific files. The ‘dead space’ between the end of a file and the end of storage allocated to that file can



↑,↓,→,← move around, Enter selects, Esc quits

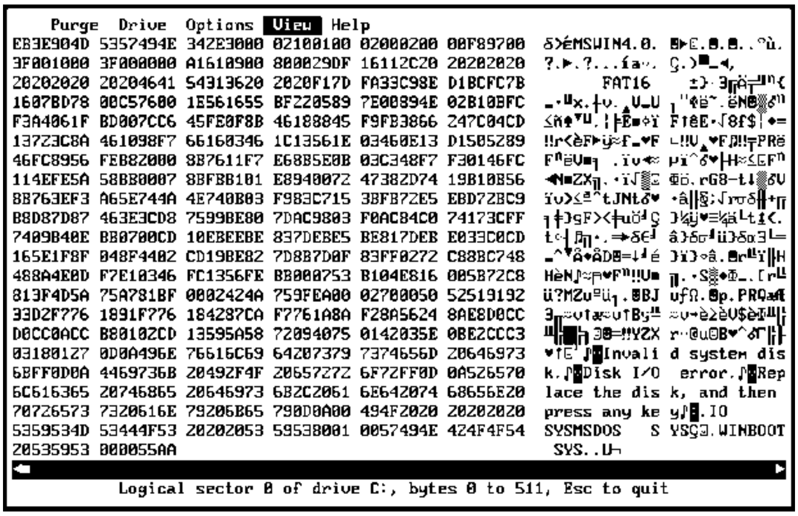
PURGE main screen

otherwise provide a particularly high-persistence trap for confidential data, because these areas are never overwritten during normal operation.

If purging of the **entire disk** is selected, the entire contents of a disk are erased positively and irretrievably. Neither standard system utilities nor special retrieval programs can reconstruct a disk which has been correctly erased with PURGE.

Features

- The secure erasure of files, unused space and complete floppy or hard disk drives.
- Three levels of security; ‘Quick’ (1 overwrite), ‘Government’ (3 overwrites followed by a verify), and ‘Military’ (7 overwrites followed by a verify).
- Any logical or physical sector on the disk can be examined.
- Erasure at physical or logical sector levels.
- Safety features to prevent accidental erasure of the wrong drive, and warning the user if any part of the purging procedure has been unsuccessful.



```
EB3E904D 5357494E 342E3000 02100100 02000200 00F89700  δ>éMSWIN4.0. ▣►ε.▣.▣..⌐ù.
3F001000 3F000000 A1610900 800029DF 16112C20 20202020  ?.►.?...ía▿. Ç.)▀_◄,
20202020 20204641 54313620 2020F17D FA33C98E D1BCFC7B        FAT16      ±}·3╔Ä╤╝ⁿ{
1607BD78 00C57600 1E561655 BF220589 7E00894E 02B10BFC  ▬•╜x.┼v.▲V▬U ┐"♠ë~.ëN☻▒δⁿ
F3A4061F BD007CC6 45FE0F8B 46188845 F9FB3866 247C04CD  ≤ñ♠▼╜.|╞E■☼ïF↑êE·√8f$|♦═
13723C8A 461098F7 66160346 1C13561E 03460E13 D1505289  ‼r<èF►ÿ≈f▬♥F∟‼V▲♥F♫‼╤PRë
46FC8956 FEB82000 8B7611F7 E68B5E0B 03C348F7 F30146FC  Fⁿë V■╕ .ïv◄≈µï^♂♥├H≈≤☺Fⁿ
114EFE5A 58BB0007 8BFBB101 E8940072 47382D74 19B10B56  ◄N■ZX╗.•ï√▒☺Φö.rG8-t↓▒♂V
8B763EF3 A65E744A 4E740B03 F983C715 3BFB72E5 EBD72BC9  ïv>≤ª^tJNt♂♥∙â╟§;√rσδ╫+╔
B8D87D87 463E3CD8 7599BE80 7DAC9803 F0AC84C0 74173CFF  ╕╪}çF><╪uÖ╛Ç}¼ÿ♥≡¼ä└t↨<.
7409B40E BB0700CD 10EBEEBE 837DEBE5 BE817DEB E033C0CD  t○┤♫╗•.═►δε╛â}δσ╛ü}δα3└═
165E1F8F 048F4402 CD19BE82 7D8B7D0F 83FF0272 C88BC748  ▬^▼Å♦ÅD☻═↓╛é}ï}☼â .r╚ï╟H
488A4E0D F7E10346 FC1356FE BB000753 B104E816 005B72C8  HèN♪≈ß♥Fⁿ‼V■╗.•S▒♦Φ▬.[r╚
813F4D5A 75A781BF 0002424A 759FEA00 02700050 52519192  ü?MZuºü┐.☻BJuƒΩ.☻p.PRQæÆ
33D2F776 1891F776 184287CA F7761A8A F28A5624 8AE8D0CC  3╥≈v↑æ≈v↑Bç╩≈v→è≥èV$èΦ╨╠
D0CC0ACC B80102CD 13595A58 72094075 0142035E 0BE2CCC3  ╨╠◙╠╕☺☻═‼YZXr○@u☺B♥^♂Γ╠├
03180127 0D0A496E 76616C69 64207379 7374656D 20646973  ♥↑☺'♪◙Invalid system dis
6BFF0D0A 4469736B 20492F4F 20657272 6F72FF0D 0A526570  k.♪◙Disk I/O error.♪◙Rep
6C616365 20746865 20646973 6B2C2061 6E642074 68656E20  lace the disk, and then
70726573 7320616E 79206B65 790D0A00 494F2020 20202020  press any key♪◙.IO
5359534D 53444F53 20202053 59538001 0057494E 424F4F54  SYSMSDOS   SYSÇ☺.WINBOOT
20535953 000055AA                                       SYS..U┐
```

Logical sector 0 of drive C:, bytes 0 to 511, Esc to quit

Viewing a logical disk sector with PURGE

- Fully menu-driven, providing comprehensive on-line help at all stages.

Applications

- Erasing floppy or hard disks, which might have contained confidential information at some stage, before repair, replacement or sale.
- Erasing unused space on disks, thereby securely removing any files or temporary storage areas which have been deleted and not positively erased.
- Examining files and logical and physical sectors on a disk.

Technical details

Product name: PURGE Version 5 (April 1998).

Function: Secure disk erasure utility for floppy or hard disks. Erases entire disk partition, unused space (all free sectors plus tail-ends of files) or files.

Mode of operation: Interactive or command line driven. On-line help, hexadecimal and ASCII display of individual sector contents, security procedure checking.

Erasing speed (1.44Mb floppy): 'Quick' level; 48 seconds.

'Government' level; 5 mins 51 seconds.

'Military' level; 9 mins 6 seconds.

Disk types erased: Floppy or hard.

Shredding levels: 'Quick' (1 overwrite); overwrite with a pattern.

'Government' (3 overwrites followed by a verify); overwrite with all 1s, overwrite with all 0s, overwrite with a pattern, verify the pattern.

'Military' (7 overwrites followed by a verify); overwrite with all 1s, overwrite with all 0s, overwrite with all 1s, overwrite with all 0s, overwrite with all 1s, overwrite with all 0s, overwrite with a pattern, verify the pattern.

RSA Toolkit

Relocatable object module

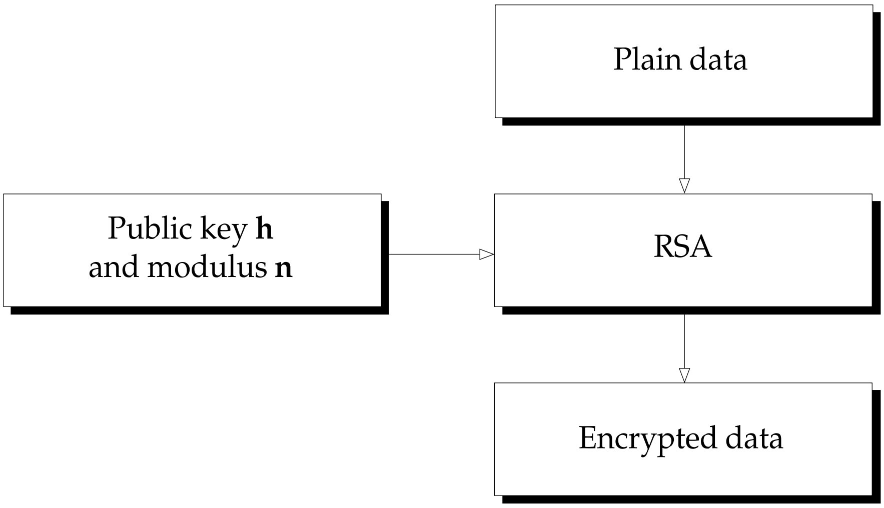
A relocatable object module which can be linked in with application programs. It provides the routines for 1024-bit encryption and decryption according to the RSA public key encryption algorithm.

Description

The RSA Toolkit is for programmers and system builders who wish to incorporate RSA public key encryption into their software. It consists of a single relocatable object module, which can be linked in to application programs. It contains four functions, which can be called at any point within a program and provide all the tools needed for encryption, decryption and RSA key generation.

The RSA Toolkit can also be used in conjunction with Sophos' DES or SPA Toolkits to produce **digital signatures** which are appended to transmitted data, to guarantee its authenticity and integrity.

One of the biggest problems in encrypted data communications is the distribution and management of encryption keys. RSA public key encryption simplifies key management problems a great deal by using **different** keys for encryption and decryption. In a typical RSA based system, anyone can encrypt data to send to a particular user, whereas only that user can decrypt it. The method can be used in a number of different ways:



RSA encryption

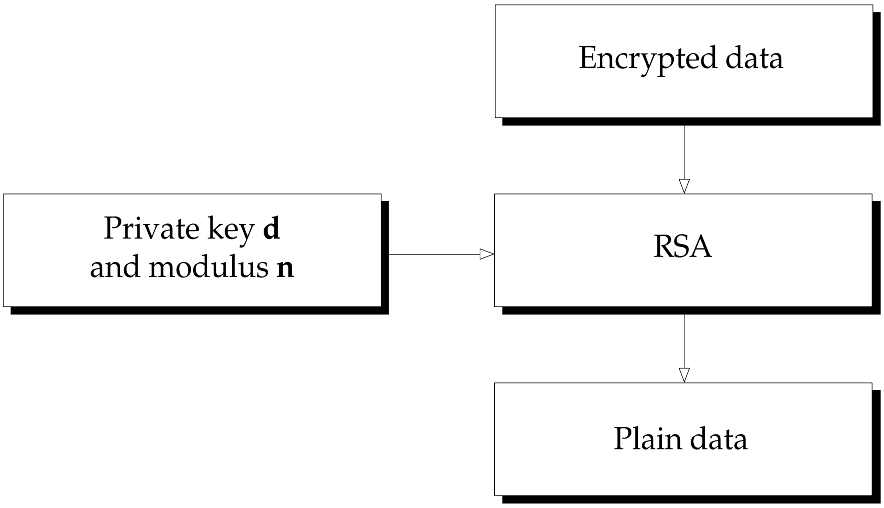
1. For the encryption of actual messages.
2. To transmit keys used with normal symmetric encryption methods such as DES.
3. To provide message authentication and digital signatures.

As DES and other algorithms are much faster to compute than RSA, (2) is generally preferable to (1).

Applications written in virtually any language such as **Fortran**, **Pascal**, **C** or **Cobol** can use the encryption facilities provided by the Toolkit. The RSA Toolkit is simply included into the application at link time.

Encryption using the RSA Toolkit transforms a plaintext block of up to 1024 bits into a ciphertext block. For the purposes of the Toolkit, these blocks are presented as arrays of consecutive integers - the precise details depend upon the operating system used and are documented in the individual user manuals.

RSA uses unique 'key-pairs' for encryption and decryption. Each user of an RSA based system has his

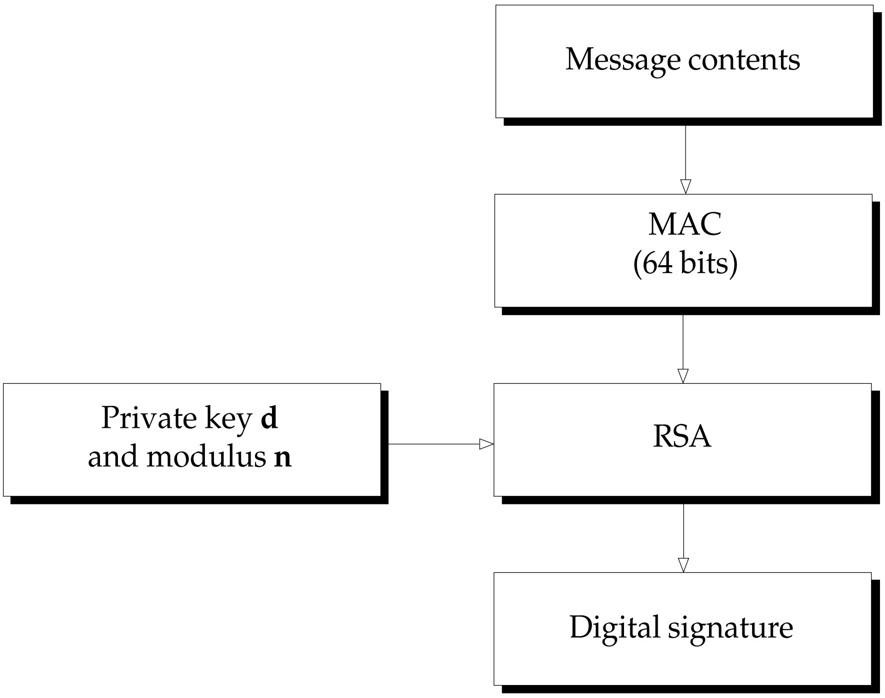


RSA decryption

own pair, consisting of a freely publicised key (referred to as **h**), and a strictly private key (referred to as **d**). The algorithm also uses a 'modulus' referred to as **n**. This is made public together with **h**, and is used both for encryption and for decryption.

The values of **h** and **n** are openly publicised, whereas **d** is kept secret. That way, anybody can encrypt a message to send to a particular person (using that person's **h** and **n**), whereas only the authorised person can decrypt it (using his own **d** and **n**). The strength of RSA lies in the fact that **d** cannot be calculated from **h** and **n**.

The security of any RSA system depends on correct and unique generation of **h**, **d** and **n**, for each user in the system. This is a highly complex task, which is catered for completely by a single function in the RSA Toolkit. The applications programmer needs only to provide a suitable source of random or pseudo-random numbers: advice is given in the user manual.



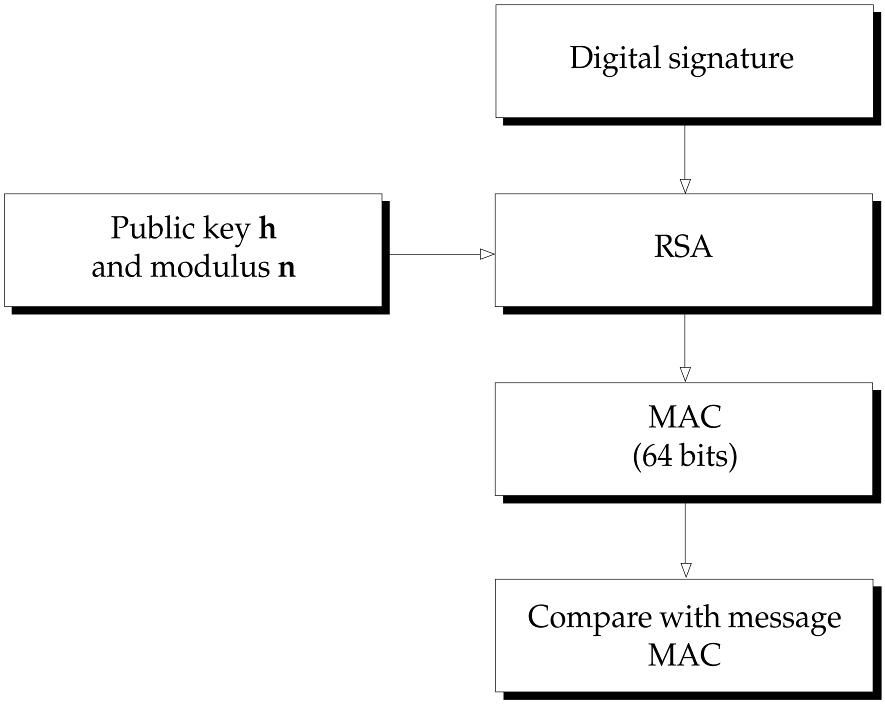
Digital signature production

To implement digital signatures, the programmer must use a one-way function to compute a MAC (Message Authentication Code, typically 64 bits) on the data to be signed. The DES Toolkit is suitable for this purpose, as it can easily be used for calculating a MAC to ANSI Standard X9.9. The MAC is then encrypted using the secret key **d**.

To verify the digital signature, the recipient uses the originator's public key **h** and modulus **n** to reconstitute the MAC from the digital signature. The recipient then compares this with the MAC which they calculate from the message received, and verifies that the two MACs are identical.

The equality of the two MACs is a positive guarantee that the message has been received correctly and that the digital signature was appended by the person who claims to have appended it.

The RSA Toolkit is tried, tested, self-contained, and requires no hardware.



Digital signature verification

Functions

RSA keys are generated by either of the two functions: **rsakey** and **srsakey**. **rsakey** requires 14 parameters, but allows the user to specify virtually all parameters relevant to the key generation. **srsakey** is a simplified version of **rsakey** requiring only 6 parameters. It supplies sensible values of non-specified parameters to **rsakey** in order to make the key generation easier. **rsakey** generates primes using Gordon's 'strong primes' method.

Before encryption or decryption can be performed the initialisation function **rsainit** must be called. Once **rsainit** has been called, all subsequent encryption or decryption can be performed using the functions **mtohmodn** or **spmtohmodn**. **mtohmodn** is called when **p** and **q** (prime factors of **n**) are not known. **spmtohmodn** is called when **p** and **q** are known. **spmtohmodn** performs the exponentiation approximately twice as fast as **mtohmodn**.

The function **rsawipe** is provided to erase any sensitive information internal to the Toolkit before program termination.

Applications

- Incorporation of RSA encryption into in-house software.
- Public key management of keys used in conjunction with other encryption algorithms (e.g. DES or SPA).
- Calculation of digital signatures for data authentication and verification.
- Incorporation of encryption into existing communications programs.

Technical details

Software name: RSATOOL.OBJ Version 2.00 (March 1997).

Function: Encryption of data using the RSA public key algorithm.

Encryption method: RSA, modulus up to 1024 bits.

RSA key generation: Routine incorporated in the toolkit. Primality testing using Rabin's test. Random number for seeding supplied by user.

VAX/VMS details: Size; 20 Blocks.

IBM/AIX details: Size; 11 Kbytes.

MS-DOS details: Size; 12.5 Kbytes.

Typical speed on 100MHz 486 PC, **h** is 65537, 512-bit **d**; 0.1 seconds for encryption of a 512-bit block, 1.6 seconds for decryption of a 512-bit block, 20 seconds for key generation.

Compatibility with various C compilers supported.

Universal
AVOMETER

RSATEST

RSA testing package

A package for verification of implementations of the RSA public key encryption algorithm.

Description

RSATEST allows the testing of hardware and software implementations of RSA, and is particularly useful during the debugging and testing of new RSA implementations.

The package consists of 4 modules: **RSAKEY** which generates RSA keys, **RSAENC** which performs RSA encryption and decryption, **PRIMEGEN** which generates prime numbers and **PRIMETST** which tests numbers for primality.

All modules can be used with numbers of up to 1024 bits and can read/write numbers in hexadecimal, ASCII and binary. It is command line driven, and so can be incorporated into batch or command files for automatic execution.

A special EFTPOS mode is provided which generates keys according to EFTPOS specifications. The PUBLIC compatibility mode allows the generation and storage of keys readable by Sophos' PUBLIC software.

Features

RSAKEY generates five numbers: the public key exponent (**h**), the public key modulus (**n**), the private key exponent (**d**), and modulus factors **p** and **q**. The number of bits in **n** can be set between 64 and 1024. RSAKEY generates primes using Gordon's 'strong primes' method. The number of bits in **r**, **s**, **t**, **p**, **q**, and the minimum difference between **p** and **q** can be set individually.

The **RSAENC** module is used to encrypt and decrypt blocks of data. RSAENC reads the public or private keys from a file, either generated by RSAKEY or supplied by the user. This is the main RSA function of modular exponentiation.

The **PRIMEGEN** module generates prime numbers with any number of bits between 64 and 1024. The source of the random numbers used by PRIMEGEN as seed values can be either truly random or pseudo-random. Pseudo-random values allow keys to be generated reproducibly. PRIMEGEN normally uses Gordon's 'strong primes' method, but allows the user to set the underlying parameters individually if required.

The **PRIMETST** module is used to test numbers (of up to 1024 bits) for primality. Rabin's test is used, performed 10 times or as set by the user.

Applications

- Verification of RSA hardware.
- Testing of RSA software.
- Testing of RSA speed on particular machines.
- Development of RSA implementations.
- EFTPOS equipment development.

Technical details

Product name: RSATEST Version 1.01 (March 1995).

Function: Testing of RSA software and hardware. Generation and testing of prime numbers.

Modules supplied: RSAKEY; RSA key generation. RSAENC; RSA encryption and decryption. PRIMEGEN; generation of prime numbers. PRIMETST; testing numbers for primality.

Input/output file format: Hexadecimal, ASCII or binary.

Random number sources: Timing of keystrokes, system clock or reproducible pseudo-random sequence.

SHRED

Positive file erasure utility

A utility for
irrecoverable erasure of
files from disk.

Description

SHRED is a utility for secure erasure of files from floppy or hard disk.

Unlike the system commands DEL or rm (under MS-DOS and UNIX respectively), SHRED not only removes the directory entry for the file you specify, but also overwrites those parts of the floppy or hard disk on which the contents of the file were stored.

This means that the contents of the file are erased positively and irretrievably. Neither standard system utilities nor special retrieval programs can reconstruct a file which has been erased with SHRED.

Shredding can be specified to one of three security levels: ‘Quick’ (1 overwrite), ‘Government’ (3 overwrites followed by a verify), and ‘Military’ (7 overwrites followed by a verify).

Technical details

Product name: SHRED Version 2.12 (May 1998).

Function: Positive file erasure utility.

Mode of operation: Command line driven.

Shredding speed (500Mb disk): ‘Quick’ level; 376 Kb/sec for 100MHz 486 Compaq DESKPRO XE 4100.

‘Government’ level; 119 Kb/sec for 100MHz 486 Compaq DESKPRO XE 4100.

‘Military’ level; 57.1 Kb/sec for 100MHz 486 Compaq DESKPRO XE 4100.

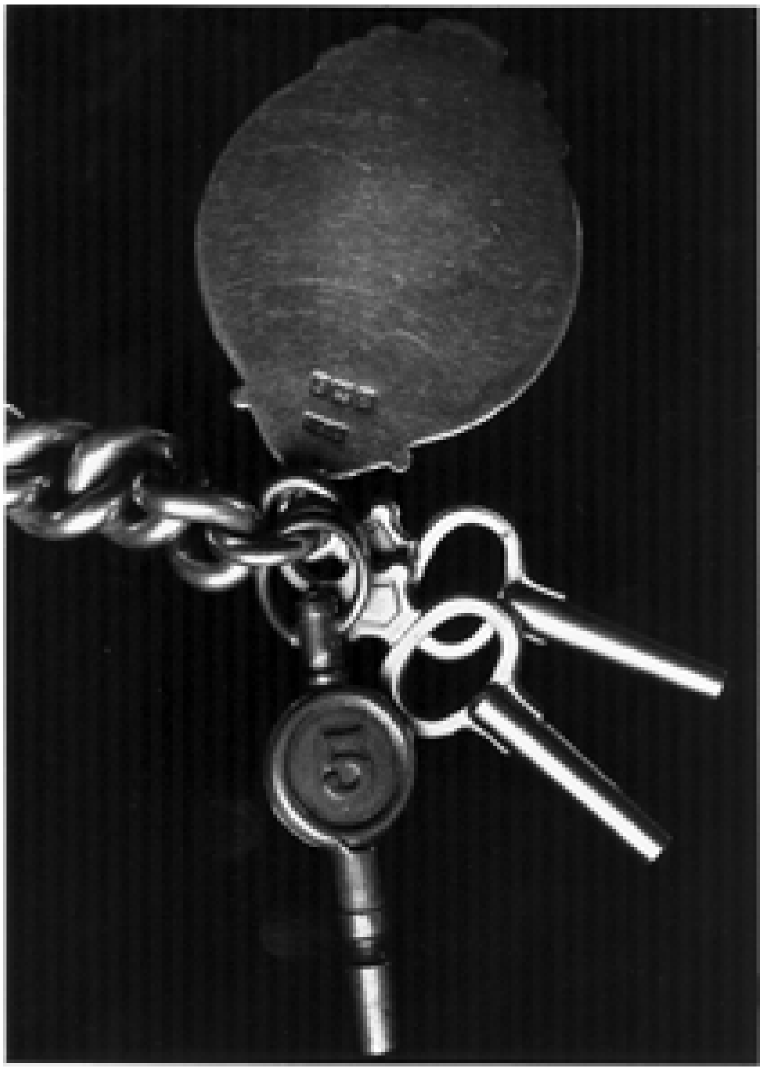
Input file format: Any file type or size.

Type of media: Any DOS compatible medium.

Shredding levels: ‘Quick’ (1 overwrite); overwrite with a pattern.

‘Government’ (3 overwrites + a verify); overwrite with all 1s, overwrite with all 0s, overwrite with a pattern, verify the pattern.

‘Military’ (7 overwrites + a verify); overwrite with all 1s, overwrite with all 0s, overwrite with all 1s, overwrite with all 0s, overwrite with all 1s, overwrite with all 0s, overwrite with a pattern, verify the pattern.



SPA Toolkit

Relocatable object module

A relocatable object module which can be linked with applications programs, to provide encryption and decryption of data using the Sophos Proprietary Algorithm (SPA).

Description

SPA (Sophos Proprietary Algorithm) is a symmetric algorithm which can be used instead of the Data Encryption Standard (DES) whenever the use of DES is inappropriate. SPA is a sophisticated algorithm which encrypts and decrypts 64-bit blocks of data on the basis of a 64-bit key.

The SPA Toolkit consists of a single relocatable object module, which can be linked in to applications programs. It contains eight functions providing all the tools needed for encryption and decryption.

Applications written in virtually any language such as **Fortran**, **Pascal**, **C** or **Cobol** can use the encryption facilities provided by the Toolkit. The SPA Toolkit is simply included into the application at link time.

Encryption according to SPA transforms a 64-bit plaintext block into a 64-bit ciphertext block. For the purposes of the Toolkit, these are blocks of 8 consecutive 8-bit bytes.

SPA uses 64-bit keys for encryption. The key is a block of 8 consecutive 8-bit bytes.

The SPA Toolkit is tried, tested, self-contained, cryptographically strong, requires no hardware and is about 10 times faster than DES (excluding file I/O).

Functions

init Before encryption or decryption can be performed, and whenever a different key is to be used, the initialisation function **init** must be called.

isweak Although there are no known weak keys in SPA, this function is provided for compatibility with the DES toolkit and always returns a 0.

encrypt and **decrypt** Once **init** has been called, all subsequent encryption or decryption can be performed using the functions

encrypt and **decrypt**. The re-initialisation is only required before using a different key.

cbcenc and **cbcdec** When several blocks of data are encrypted in succession using the same key, normal SPA encryption will produce an identical cipher block for each identical plaintext block. Repetitions in the plaintext can thus cause repetitions in the ciphertext, leading to a potential weakness in the defence against cryptanalytical attack. A recommended solution is to use Cipher Block Chaining (CBC), which makes each cipher block dependent not only on the current plaintext block and the key, but also on all previous plaintext blocks. The functions **cbcenc** and **cbcdec** automatically provide CBC for encryption and decryption.

wipe The **wipe** function is provided to erase any sensitive information internal to the Toolkit before program termination.

Example

A simple example (in C) using some of the functions provided in the Toolkit follows. It shows the basic use of the functions **init**, **encrypt** and **decrypt**. The output from the program in the example is:

```
11 22 33 44 55 66 77 88
18 2a 3b 95 8d 42 e2 a5
11 22 33 44 55 66 77 88
```

Example

```
main()
{
      static unsigned char key[]={ 0x01, 0x23,
             0x45, 0x67, 0x89, 0xab, 0xcd, 0xef };
      static unsigned char data[]={ 0x11, 0x22,
             0x33, 0x44, 0x55, 0x66, 0x77, 0x88 };

      init(key); /* intialise SPA */
      print(data);
      encrypt(data); /* encrypt 8 bytes */
      print(data);
      decrypt(data); /* decrypt 8 bytes */
      print(data);
}


void print(unsigned char s[])
{
      int i;

      for(i=0;i<8;i++) printf("%02x ",s[i]&0xff);
      printf("\n");
}
```

Applications

- Encryption of individual fields in database software.
- Professional security for financial application programs.
- Custom security software.
- Incorporation of encryption into existing programs.

Technical details

Software name: SPA.OBJ, Version 1.00 (March 1995).

Function: Encryption of data according to the Sophos Proprietary Algorithm (SPA).

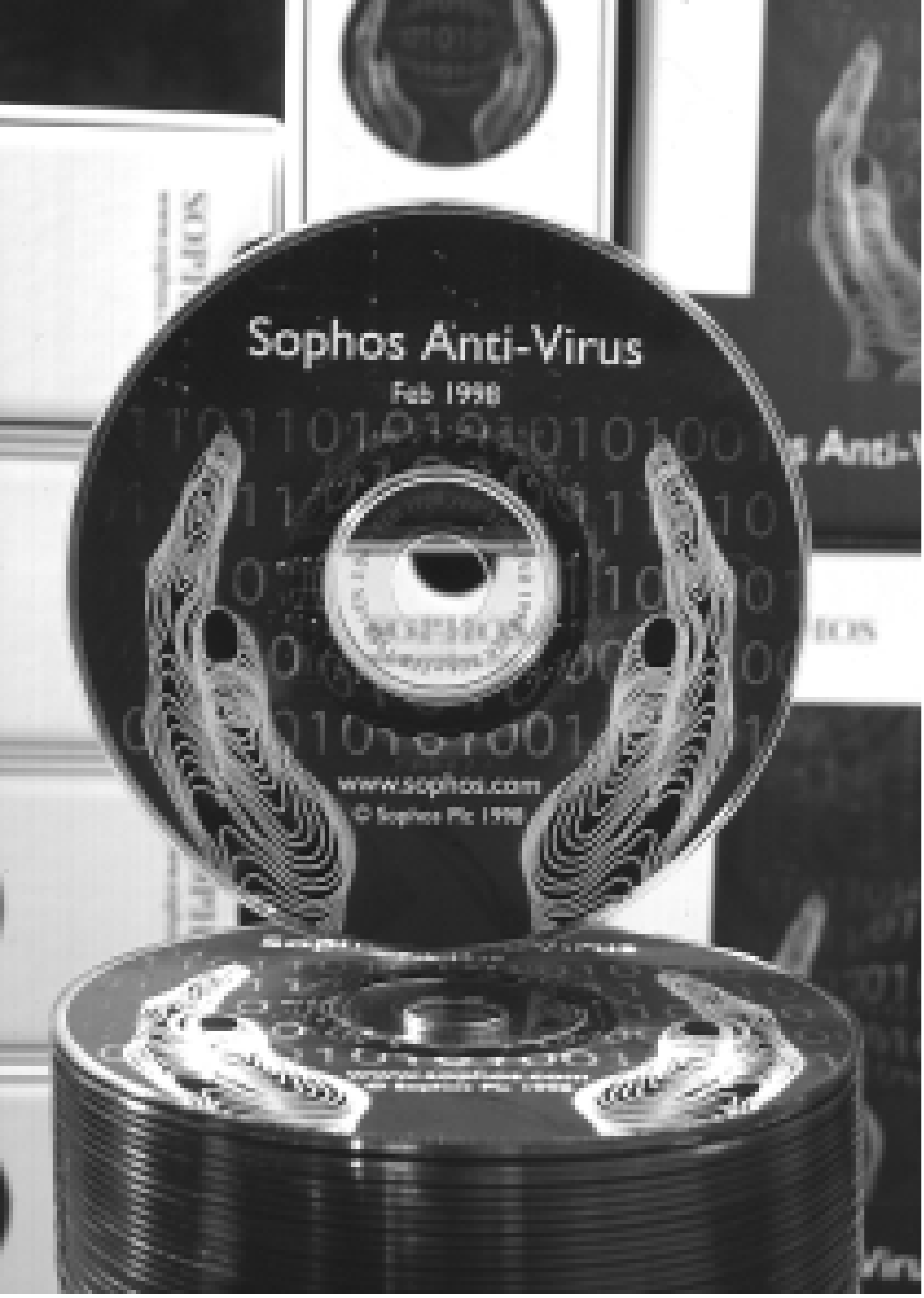
Encryption method: SPA in Electronic Code Book mode (ECB) or Cipher Block Chaining (CBC) mode.

Key checking: None, there are no known weak keys.

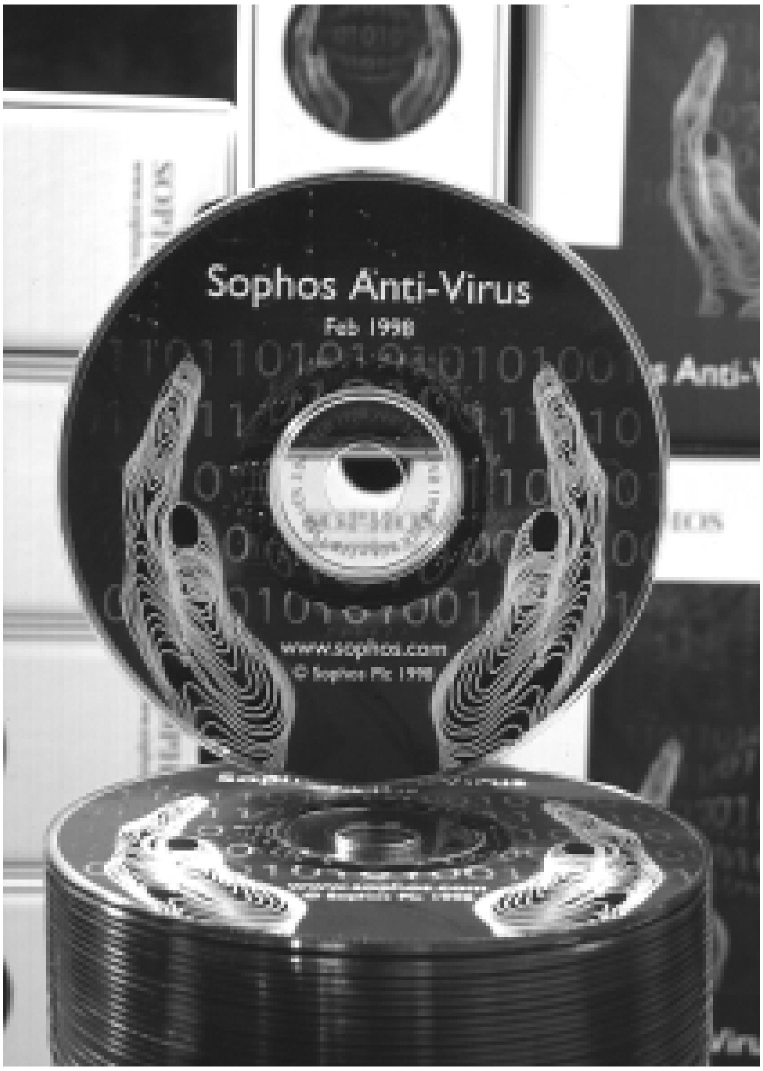
MS-DOS size: 5 K.

MS-DOS speed: 254 Kb/sec for 100 MHz 486 Compaq DESKPRO XE 4100.

Compatibility with various compilers supported.



Sophos Anti-Virus products 4


Sophos Anti-Virus
Feb 1998
www.sophos.com
© Sophos Plc 1998

Introduction to Sophos Anti-Virus

Virus detection, reporting and disinfection for
individual PCs and entire networks.

What is Sophos Anti-Virus?

Sophos Anti-Virus provides on-demand, scheduled and on-access virus checking, automatic reporting and disinfection for individual PCs and entire networks.

It includes two systems:

- **SWEEP** provides immediate and scheduled scanning, of disks, files and documents, and
- **InterCheck** checks each item as the user attempts to access it and grants access only if it is virus-free.

SWEEP can be installed on its own; the use of InterCheck is optional.

About SWEEP

SWEEP is a virus-specific virus scanner that detects all viruses known to Sophos at the time of release.

It can provide on-demand and scheduled scanning of workstations or file servers, and can also deal with requests for on-access virus-checking from networked workstations (see ‘About InterCheck’ below).

Monthly updates are available by post, email or fax, or from the Sophos Web site.

About InterCheck

InterCheck ensures that unknown files (e.g. programs, documents, email attachments or Internet downloads) and disks cannot be used until checked for viruses.

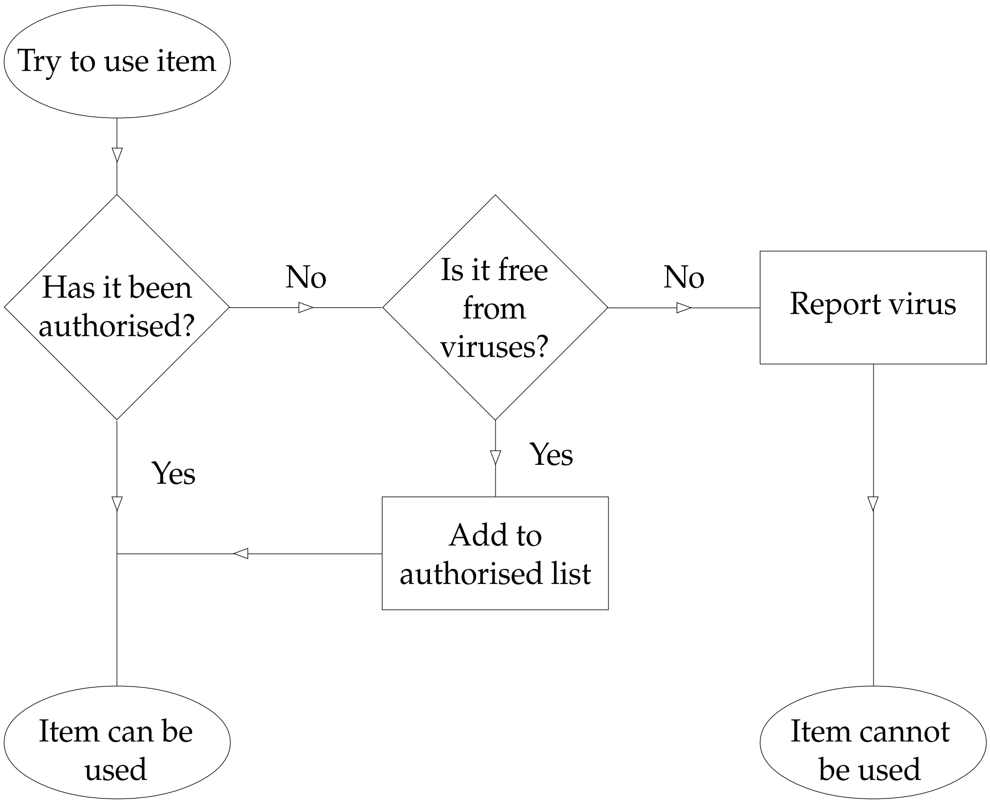
How does InterCheck work?

InterCheck splits the task of virus detection between a client and a server.

The **InterCheck client** monitors all file and disk accesses. Whenever an item is accessed, the client compares it with a list of authorised items. If a match is found, the access is permitted. If a match is not found, the file is sent for virus checking. The **InterCheck server**, using an installation of SWEEP, performs the actual virus checks.

If the item is found to be clean, it is added to the list of authorised items (a **checksum file**) and the access is allowed to continue. Any further accesses of this item are completed without the need for further authorisation, unless it is modified, in which case authorisation is again automatically requested.

However, if a virus is found, InterCheck denies access to the item, so the workstation cannot be infected.



How InterCheck works

What types of InterCheck installation are there?

There are two main types of InterCheck installation:

With networked InterCheck clients

The InterCheck client is placed on the workstation to be protected, while the InterCheck server is on a remote machine. The client sends files over the network to the server for virus checking

With stand-alone InterCheck clients

The InterCheck client does not have to communicate with a remote InterCheck server, and uses a local installation of SWEEP for virus checking.

Networked InterCheck clients

Networked InterCheck clients are easier to administer and use fewer system resources on the client workstations.

Networked InterCheck clients may also make use of a **central checksum file** stored on the server (if available). If the InterCheck client cannot find an item in its local checksum file, it will then look in the central file, which lists all items that have been authorised for use on any of the connected workstations. Thus, when one InterCheck client has had an item checked, all other InterCheck clients can access that item without further checking.

Stand-alone InterCheck clients

Stand-alone InterCheck clients generally offer faster initial authorisation of files, and can also be used on machines not always connected to the network.

On machines with network access they can be installed from the file server, and updated automatically whenever the machine is connected to the network.

Using Sophos Anti-Virus on a network

If Sophos Anti-Virus is used on a network, it is possible to:

- Update workstation installations of Sophos Anti-Virus automatically.
- Set up central reporting of virus incidents.
- Choose between server based on-access scanning (to minimise workstation overhead) and local on-access scanning (to speed up scanning and reduce network traffic). See the 'About InterCheck' section above.

Sophos Anti-Virus management tools

There are several management tools designed to make it easier to distribute, administer and update Sophos Anti-Virus on large networks.

These include SWDEPLOY and SAVADMIN for managing Sophos Anti-Virus on Windows NT networks, SWCONSOL for remote control of NetWare servers, and SGET for automated updating of Sophos Anti-Virus from the Internet.

See the 'Sophos Anti-Virus Management Tools' chapter for details.

Sophos Anti-Virus for Banyan VINES

Incorporating SWEEP and InterCheck

Virus-specific detection software
for file servers running Banyan VINES.

Description

Sophos Anti-Virus for Banyan VINES is virus-specific detection and disinfection software which is installed on a Banyan VINES server.

Virus checking

Sophos Anti-Virus for Banyan VINES incorporates SWEEP and InterCheck.

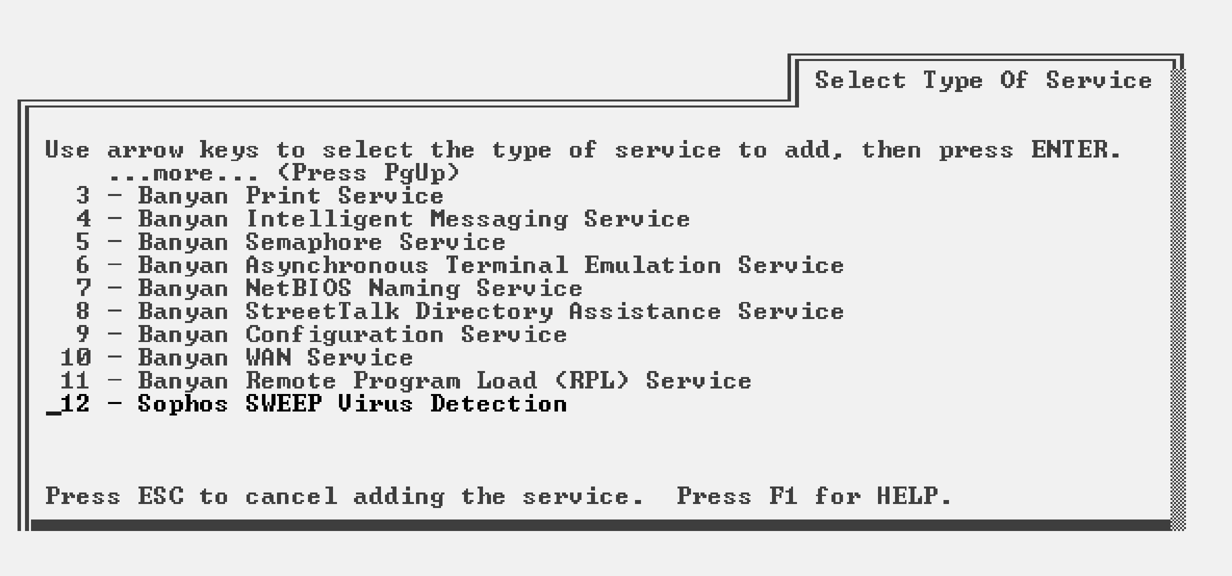
SWEEP provides on-demand and scheduled virus checking of files held on Banyan VINES file servers.

InterCheck provides server or workstation based on-access virus checking for client workstations.

Advantages of virus checking on the server

Using SWEEP for Banyan VINES to check files on the server has several advantages over running SWEEP for DOS/Windows 3.x from a workstation:

- It does not involve DOS in checking, which means that it is not susceptible to stealth techniques.



```
                                                  Select Type Of Service

Use arrow keys to select the type of service to add, then press ENTER.
    ...more... (Press PgUp)
  3 - Banyan Print Service
  4 - Banyan Intelligent Messaging Service
  5 - Banyan Semaphore Service
  6 - Banyan Asynchronous Terminal Emulation Service
  7 - Banyan NetBIOS Naming Service
  8 - Banyan StreetTalk Directory Assistance Service
  9 - Banyan Configuration Service
 10 - Banyan WAN Service
 11 - Banyan Remote Program Load (RPL) Service
_12 - Sophos SWEEP Virus Detection


Press ESC to cancel adding the service.  Press F1 for HELP.
```

Installing the SWEEP for Banyan VINES service

- It uses the local UNIX file system, rather than the network Banyan Filing System, to access files. Thus it is approximately three times faster and causes no network traffic.
- It is not subject to BFS access restrictions, and does not need to be given rights to every file in the system.

Virus elimination

SWEEP for Banyan VINES provides a virus elimination facility for dealing with viruses found on the server.

Infected files can be renamed, shredded, moved or copied.

Infected documents can be disinfected.

Viruses found on client workstations can be dealt with using the version of SWEEP specific to their operating system or SWEEP for DOS/Windows 3.x.

Features

- Checks Banyan Filing System (BFS) volumes for the presence of all viruses known to Sophos at the time of release.
- Incorporates Sophos' proprietary InterCheck client-server virus detection technology, which can provide server or workstation based on-access scanning for workstations.
- Is updated twelve times a year with the latest virus information. Urgent updates can be distributed by fax or email or downloaded from the Sophos Web site.
- Easily detects polymorphic viruses using Sophos' advanced Virus Description Language (VDL) and a built-in code emulator.
- Scans inside compressed files.

- Detects and disinfects Microsoft Word, Excel and Office 97 macro viruses.
- Offers two levels of security, allowing a ‘quick sweep’ which looks for virus identities in the parts of a file that are likely to contain a virus, and a ‘full sweep’ which looks for virus fragments in every part of a file.
- Is easy to use, and easily integrated into complex virus-checking applications, such as the automated unattended checking of file servers.
- Can be scheduled, so SWEEP can be configured to perform regular checks without any further operator action.
- Allows immediate automatic notification of virus infection.
- Has low network overhead.
- Licence includes Sophos Anti-Virus for DOS/Windows 3.x and Windows 95.

Technical details

Product name: Sophos Anti-Virus for Banyan VINES Version 3.10 (June 1998).

Function: Scanning BFS volumes and providing InterCheck service.

Detection method: ‘Quick’ or ‘Full’ mode.

Mode of operation: Banyan VINES server based service.

Viruses detected: 15,015 (June 1998).

Updates: 12 per year, sent by post at the beginning of every month, or available for download via WWW.

Security reports: A full report of the sweep can be generated and printed. In addition, a message can be broadcast to all members of a pre-defined user group if a virus is detected.

Specification of items to be checked: Memory, individual files, files with specific extensions, files not on exclusion list, and recursive searching of subdirectories.

System requirements: Banyan VINES version 5 or greater, excepting VINES 6.00(0) and VINES 6.00 (10).

VINES 6.00(0) must either be upgraded to 6.00 (10) with site specific patch V95008 or be upgraded to 6.20 (0).

VINES 6.00 (10) must have site specific patch V95008 applied, or be upgraded to 6.20 (0).

Nikon 8x23 CF
ABINGDON

Sophos Anti-Virus for DOS/Windows 3.x

Incorporating SWEEP and InterCheck

INTERCHECK™ TECHNOLOGY

Virus-specific detection software
for PCs running DOS/Windows 3.x.

Description

Sophos Anti-Virus for DOS/Windows 3.x is virus-specific detection and disinfection software which can be installed on workstations or file servers.

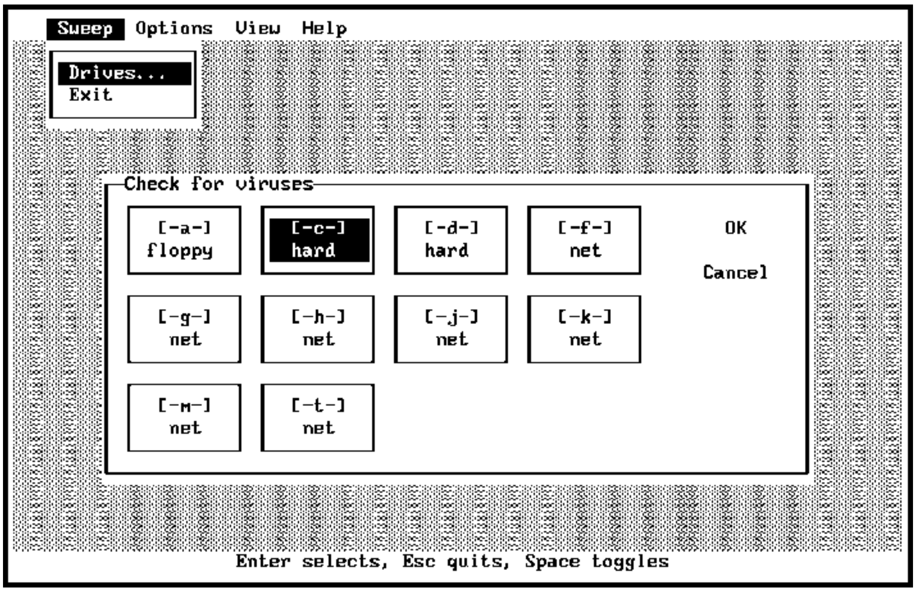
Virus checking

Sophos Anti-Virus for DOS/Windows 3.x incorporates SWEEP and InterCheck.

SWEEP provides on-demand and scheduled virus checking on workstations or file servers.

The stand-alone InterCheck client provides locally based on-access virus checking on workstations.

SWEEP for DOS/Windows 3.x can also be installed with InterCheck to provide server based on-access virus checking on servers for which Sophos does not produce ‘native’ scanning software. It can run on a ‘soft PC’ on the server, or on a dedicated workstation. This allows InterCheck to be implemented on PC networks such as AIX, AS/400, HP-UX, LANtastic, NetWare Lite, OSF/1, Solaris, UNIX and WFWG.



Enter selects, Esc quits, Space toggles

SW user interface

Menu-driven or command line control

SWEEP for DOS/Windows 3.x can be used in two ways:

- Through the menu-driven utility SW.
- By specifying command line qualifiers.

Using the former is simpler, while the latter gives greater flexibility and control.

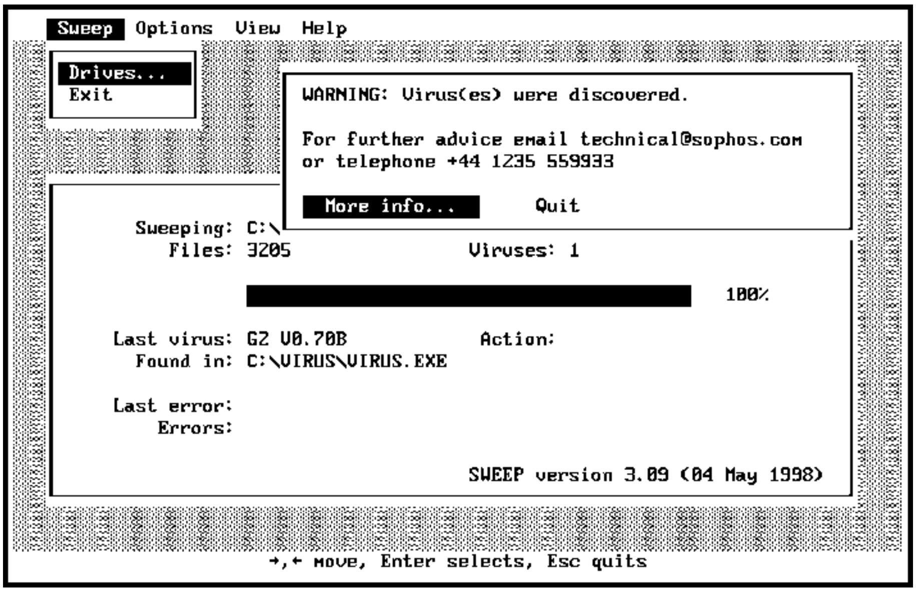
Virus elimination

SWEEP provides a virus elimination facility.

Infected files are either deleted (reversible) or shredded (irreversible).

Infected documents can be disinfected.

Infected boot sectors can be disinfected (i.e. restored to the exact state prior to infection) or neutralised. Neutralisation is performed by modifying the boot sector in such a way as to cause the machine to hang if it is booted from the neutralised disk.



```
WARNING: Virus(es) were discovered.

For further advice email technical@sophos.com
or telephone +44 1235 559933

   Sweeping: C:\
      Files: 3205                  Viruses: 1

                                                  100%

 Last virus: GZ V0.70B              Action:
   Found in: C:\VIRUS\VIRUS.EXE

 Last error:
     Errors:

                              SWEEP version 3.09 (04 May 1998)

→,← move, Enter selects, Esc quits
```

SWEEP finding a virus

Features

- Checks local hard disks, floppy disks and network drives for the presence of all viruses known to Sophos at the time of release.
- Incorporates Sophos' proprietary InterCheck client-server virus detection technology, which allows the use of server based software for checking workstations.
- Is updated twelve times a year, and urgent updates can be distributed by fax or email or downloaded from the Sophos Web site.
- Easily detects polymorphic viruses using Sophos' advanced Virus Description Language (VDL) and a built-in code emulator.
- Scans inside compressed files.
- Detects and disinfects Microsoft Word, Excel and Office 97 macro viruses.
- Offers two levels of security, allowing a 'quick sweep' which looks for virus identities in the parts of a file that are likely to contain a virus, and a 'full sweep' which looks for virus fragments in every part of a file.
- File server licence includes a scheduling utility, so SWEEP can be configured to perform regular checks without any further operator action.
- Supports background or priority operation.
- Includes an extensive on-line virus information database.
- Licence includes SWEEP for Windows 95.

Technical details

Product name: Sophos Anti-Virus for DOS/Windows 3.x Version 3.10 (June 1998).

Function: Scanning individual drives for known viruses.

Detection method: ‘Quick’ or ‘Full’ mode.

Mode of operation: Transient, non-memory-resident process. Command-line operation for batch process integration. Fully mouse-driven interactive shell for easy use.

Viruses detected: 15,015 (June 1998).

Updates: 12 per year, sent by post every month, or available for download via WWW.

Security reports: A full report of the sweep can be generated and printed.

Specification of items to be checked: Memory ranges, individual files, recursive searching of subdirectories, individual disk sectors, ranges of disk sectors, ranges of bytes within disk sectors, and absolute or logical disk sector addressing.

Specification of patterns: Strings of hexadecimal digits, up to 24 bytes per pattern can be specified by the user. Patterns can be given names.

Specification of identities: Additional identities can be added into SWEEP at run-time, either as ASCII text or as binary files, as supplied by Sophos.

Return codes: SWEEP returns exit codes to the system, which are testable with the ‘ON ERRORLEVEL’ statement in batch files. Conditions recognised:

0 No viruses discovered and no errors encountered
1 Execution interrupted
2 Errors encountered
3 Virus(es) discovered

Virus database: On-line virus database within SW allows search by virus name or alias, infected object type, memory-resident behaviour, trigger conditions or keyword.

Sophos Anti-Virus for NetWare

Incorporating SWEEP and InterCheck

Virus-specific detection software for file servers
running Novell NetWare 3.11 and later.

Description

Sophos Anti-Virus for NetWare is virus-specific detection and disinfection software which is installed on NetWare servers.

Virus checking

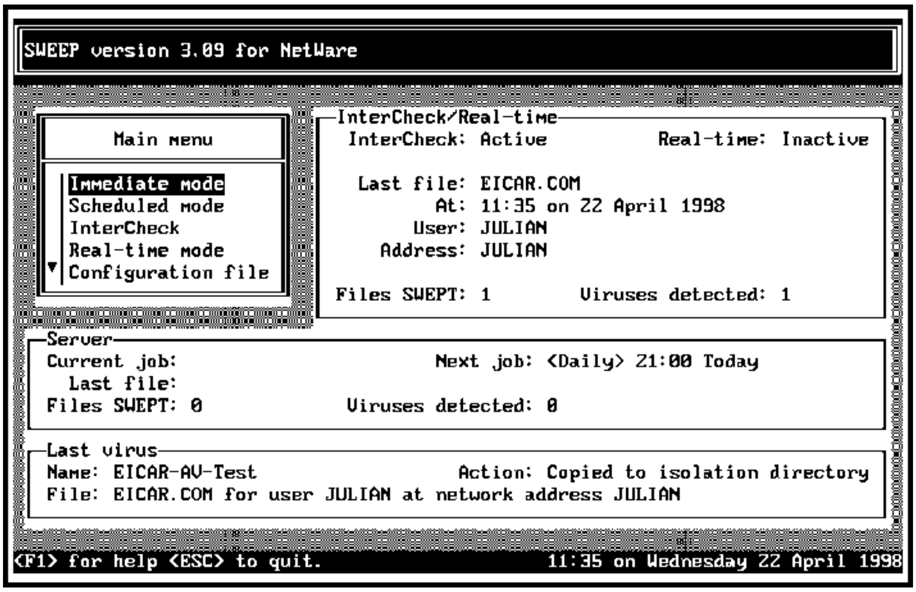
Sophos Anti-Virus for NetWare incorporates SWEEP and InterCheck.

SWEEP for NetWare provides on-demand and scheduled virus checking of volumes on the NetWare server.

InterCheck provides server or workstation based on-access virus checking for connected workstations.

Advantages of virus checking on the server

Using SWEEP for NetWare to scan volumes on the server has two advantages over running SWEEP for DOS/Windows 3.x from a workstation:



```
SWEEP version 3.09 for NetWare

  Main menu            InterCheck/Real-time
                        InterCheck: Active          Real-time: Inactive
  Immediate mode
  Scheduled mode         Last file: EICAR.COM
  InterCheck                    At: 11:35 on 22 April 1998
  Real-time mode              User: JULIAN
  Configuration file       Address: JULIAN

                       Files SWEPT: 1        Viruses detected: 1
Server
Current job:                      Next job: <Daily> 21:00 Today
  Last file:
Files SWEPT: 0              Viruses detected: 0
Last virus
Name: EICAR-AV-Test                    Action: Copied to isolation directory
File: EICAR.COM for user JULIAN at network address JULIAN
<F1> for help <ESC> to quit.                 11:35 on Wednesday 22 April 1998
```

SWEEP for NetWare

- It does not involve DOS in sweeping, which means that it is not susceptible to the stealth techniques used by some viruses.
- It runs on the file server to scan files stored there, so there is no increase in network traffic.

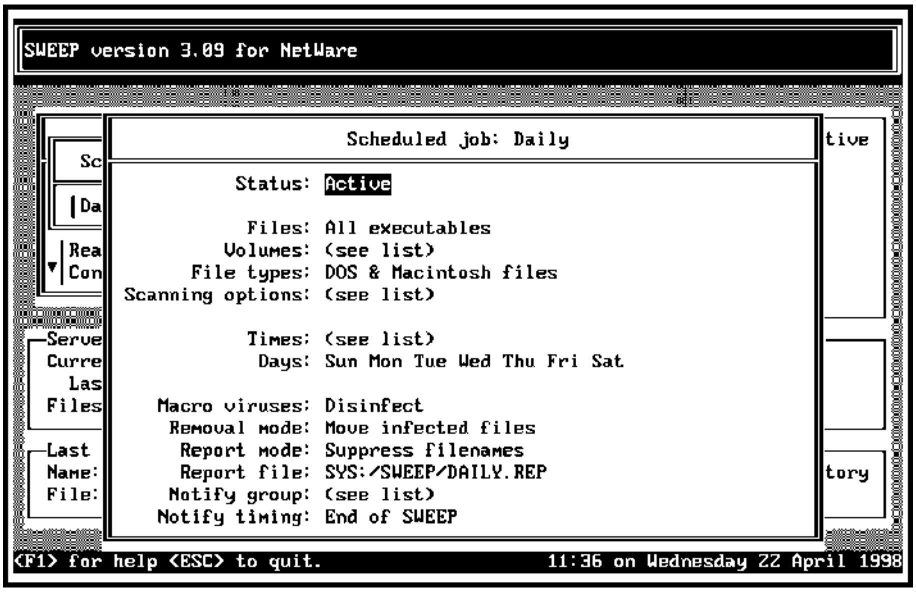
Virus elimination

SWEEP for NetWare provides an automatic virus elimination facility for viruses found on the server. Thus action can be taken as soon as a virus is detected, even if no operator is present.

Infected files can be renamed, deleted (reversible), purged (irreversible), or moved.

Infected documents can be disinfected.

Viruses found on client workstations can be dealt with using the version of SWEEP specific to their operating system or SWEEP for DOS/Windows 3.x.

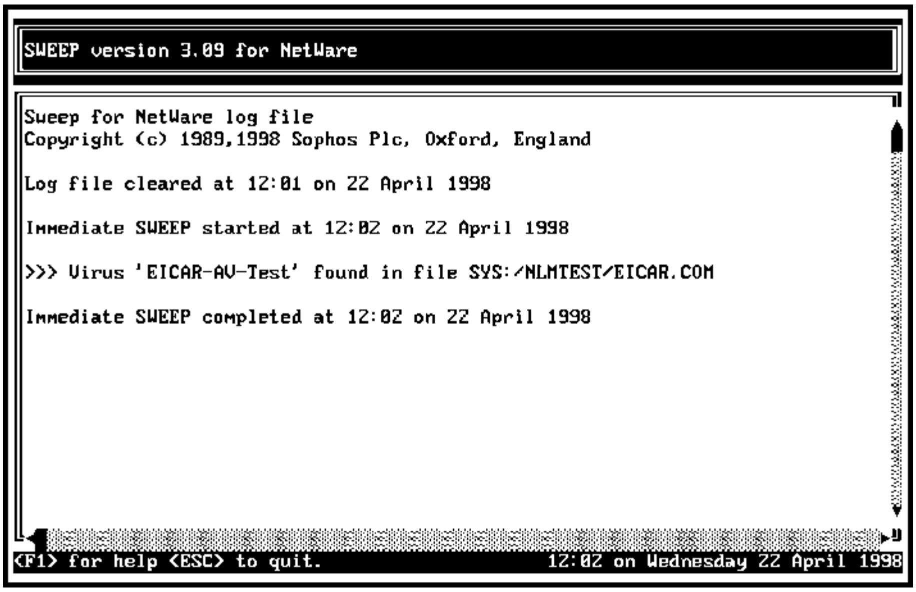


```
SWEEP version 3.09 for NetWare

                Scheduled job: Daily

          Status: Active

           Files: All executables
         Volumes: (see list)
      File types: DOS & Macintosh files
Scanning options: (see list)

           Times: (see list)
            Days: Sun Mon Tue Wed Thu Fri Sat

  Macro viruses: Disinfect
   Removal mode: Move infected files
    Report mode: Suppress filenames
    Report file: SYS:/SWEEP/DAILY.REP
   Notify group: (see list)
  Notify timing: End of SWEEP

<F1> for help <ESC> to quit.          11:36 on Wednesday 22 April 1998
```

Selecting options for a scheduled job

Features

- Checks volumes on NetWare servers for the presence of all viruses known to Sophos at the time of release, including Macintosh viruses.
- Incorporates Sophos' proprietary InterCheck client-server virus detection technology, which can provide server or workstation based on-access scanning for workstations.
- Is updated twelve times a year, and urgent updates can be distributed by fax or email or downloaded from the Sophos Web site.
- Can be updated automatically on one or more NetWare servers.
- Offers automatic updating of InterCheck clients on connected workstations.
- Easily detects polymorphic viruses using Sophos' advanced Virus Description Language (VDL) and a built-in code emulator.
- Scans inside compressed files.



```
SWEEP version 3.09 for NetWare

Sweep for NetWare log file
Copyright (c) 1989,1998 Sophos Plc, Oxford, England

Log file cleared at 12:01 on 22 April 1998

Immediate SWEEP started at 12:02 on 22 April 1998

>>> Virus 'EICAR-AV-Test' found in file SYS:/NLMTEST/EICAR.COM

Immediate SWEEP completed at 12:02 on 22 April 1998

<F1> for help <ESC> to quit.                    12:02 on Wednesday 22 April 1998
```

Viewing the SWEEP for NetWare log file

- Detects and disinfects Microsoft Word, Excel and Office 97 macro viruses.
- Offers two levels of security, allowing a ‘quick sweep’ which looks for virus identities in the parts of a file that are likely to contain a virus, and a ‘full sweep’ which looks for virus fragments in every part of a file.
- Includes full scheduling facilities, so SWEEP can be configured to perform regular checks without any further operator action.
- Supports background or priority operation.
- Licence includes Sophos Anti-Virus for DOS/Windows 3.x and Windows 95.

Technical details

Product name: Sophos Anti-Virus for Novell NetWare Version 3.10 (June 1998).

Function: Scanning individual workstations and file servers for known viruses.

Detection method: ‘Quick’ or ‘Full’ mode.

Mode of operation: Interactive, menu-driven. All options are selectable by Novell-style menus.

Viruses detected: 15,015 (June 1998).

Updates: 12 per year, sent by post every month, or available for download via WWW.

Security reports: A full report of the sweep can be generated. In addition, SWEEP can broadcast a message to all members of selected NetWare user groups when a virus is detected.

System requirements: IBM PC and compatibles running Novell NetWare 3.11 and later.

Novell certification: See the Sophos Web site at http://www.sophos.com/ for the latest certification status.

Sophos Anti-Virus for OpenVMS

Incorporating VSWEEP and InterCheck

Virus-specific detection software for OpenVMS VAX and AXP systems running Pathworks or other network software.

Description

Sophos Anti-Virus for OpenVMS is virus-specific detection and disinfection software that is installed on OpenVMS VAX and AXP systems running PATHWORKS or other network software.

Virus checking

Sophos Anti-Virus for OpenVMS incorporates VSWEEP and InterCheck.

VSWEEP provides on-demand virus checking of DOS files held on an OpenVMS system under PATHWORKS File Services, including those in FAT container files or Disk Services.

InterCheck provides server or workstation based on-access virus checking for connected workstations.

Advantages of virus checking on the server

Using VSWEEP to check files held on the server, rather than running SWEEP for DOS/Windows 3.x from a workstation, has two advantages:

- It does not involve DOS in the sweeping process, which means that it is not susceptible to the stealth techniques used by some viruses.
- No load is imposed on the network, and the process does not tie up a workstation.

Virus elimination

Viruses found on the server can be eliminated:

Infected files can be deleted.

Infected documents can be disinfected.

Viruses found on client workstations can be dealt with using the version of SWEEP specific to their operating system or SWEEP for DOS/Windows 3.x.

Features

- Checks OpenVMS file servers for the presence of all viruses known to Sophos at the time of release.
- Incorporates Sophos' proprietary InterCheck client-server virus detection technology, which can provide server or workstation based on-access scanning for workstations.
- Is updated twelve times a year, and urgent updates can be distributed by fax or email or downloaded from the Sophos Web site.
- Easily detects polymorphic viruses using Sophos' advanced Virus Description Language (VDL) and a built-in code emulator.
- Detects and disinfects Microsoft Word, Excel and Office 97 macro viruses.
- Offers two levels of security, allowing a 'quick sweep' which looks for virus identities in the parts of a file that are likely to contain a virus, and a 'full sweep' which looks for virus fragments in every part of a file.
- Offers OpenVMS managers centralised control over and notification of virus detection.
- Can be scheduled to sweep as a normal OpenVMS job, and configured to run continuously.

Technical details

Product name: Sophos Anti-Virus for OpenVMS Version 3.10 (June 1998).

Mode of operation: Interactive, batch-driven or detached.

Viruses detected: 15,015 (June 1998).

Updates: 12 per year, sent by post every month, or by fax or email, or downloaded from the Sophos Web site.

Specification of items to be checked: Individual files, wildcards and recursive searching of subdirectories. File and disk services are supported.

Sophos Anti-Virus for OS/2

Incorporating SWEEP and InterCheck

Virus-specific detection software
for OS/2 file servers or workstations.

Description

Sophos Anti-Virus for OS/2 is virus-specific detection and disinfection software which is installed on OS/2 based computers.

Virus checking

Sophos Anti-Virus for OS/2 incorporates SWEEP and InterCheck.

SWEEP for OS/2 is used to virus check files held on an OS/2 file server or an OS/2 workstation.

InterCheck, using SWEEP for OS/2, provides server based on-access virus checking for workstations.

Virus elimination

SWEEP for OS/2 provides a virus elimination facility.

Infected files can be deleted (reversible) or positively overwritten (irreversible).

Infected documents can be disinfected.

Infected boot sectors can be disinfected (for some viruses) or neutralised. Neutralisation is performed by modifying the boot sector in such a way as to cause the machine to hang if it is booted from the neutralised disk.

Features

- Checks local hard disks, floppy disks and network drives for the presence of all viruses known to Sophos at the time of release.
- Incorporates Sophos’ proprietary InterCheck client-server virus detection technology, which allows the use of server based software for checking workstations.
- Is updated twelve times a year, and urgent updates can be distributed by fax or email or downloaded from the Sophos Web site.
- Easily detects polymorphic viruses using Sophos’ advanced Virus Description Language (VDL) and a built-in code emulator.
- Scans inside compressed files.
- Detects and disinfects Microsoft Word, Excel and Office 97 macro viruses.
- Offers two levels of security, allowing a ‘quick sweep’ which looks for virus identities in the parts of a file that are likely to contain a virus, and a ‘full sweep’ which looks for virus fragments in every part of a file.
- Is easy to use, yet easily integrated into complex virus-checking applications, such as the automated unattended checking of file servers.
- Can be scheduled with the aid of utilities supplied with the network software, so SWEEP can be configured to perform regular checks without any further operator action.
- Can be set to run in the background and as a low, medium or high priority operation.
- Licence includes Sophos Anti-Virus for DOS/Windows 3.x and Windows 95.

Technical details

Product name: Sophos Anti-Virus for OS/2 Version 3.10 (June 1998).

Function: Scanning individual workstations and file servers for known viruses.

Detection method: ‘Quick’ and ‘Full’ mode.

Mode of operation: OS/2 process.

Viruses detected: 15,015 (June 1998).

Updates: 12 per year, sent by post at the beginning of each month, or available for download via WWW.

Security reports: SWEEP for OS/2 is able to generate and print security reports.

Specification of items to be checked: Individual files, recursive searching of subdirectories, individual disk sectors, ranges of disk sectors, ranges of bytes within disk sectors, and absolute or logical sector addressing.

Specification of patterns: Strings of hexadecimal digits, up to 24 bytes per pattern can be specified by the user. Patterns can be given names.

Specification of identities: Additional identities can be added into SWEEP at run-time, either as ASCII text or as binary files, as supplied by Sophos.

Return codes: SWEEP returns exit codes testable with the ‘ON ERRORLEVEL’ statement in batch or command files. Conditions recognised:

0 No viruses discovered and no errors encountered
1 Execution interrupted
2 Errors encountered
3 Virus(es) discovered

System requirements: OS/2 Version 1.1 and above, except OS/2 Ver 2.0.

File systems supported:	FAT, HPFS.
Networks supported:	Novell NetWare, IBM OS/2 LAN Server, Microsoft OS/2 LAN Manager.

Sophos Anti-Virus for Windows 95

Incorporating SWEEP and InterCheck

Virus-specific detection software
for PCs running Windows 95.

Description

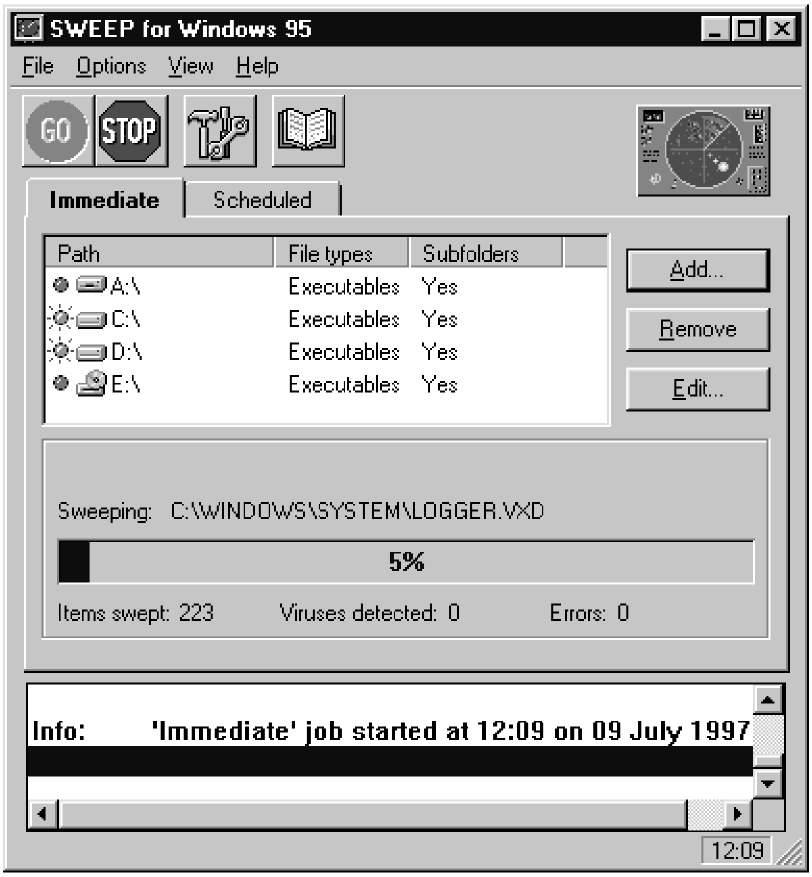
Sophos Anti-Virus for Windows 95 is virus-specific detection and disinfection software for PCs running Windows 95.

Virus checking

Sophos Anti-Virus for Windows 95 incorporates SWEEP and InterCheck.

SWEEP provides on-demand and scheduled virus checking on the workstation.

The stand-alone InterCheck client provides on-access virus checking on the workstation.



Path	File types	Subfolders
A:\	Executables	Yes
C:\	Executables	Yes
D:\	Executables	Yes
E:\	Executables	Yes

Sweeping: C:\WINDOWS\SYSTEM\LOGGER.VXD

5%

Items swept: 223 Viruses detected: 0 Errors: 0

Info: 'Immediate' job started at 12:09 on 09 July 1997

12:09

SWEEP for Windows 95 sweeping for viruses

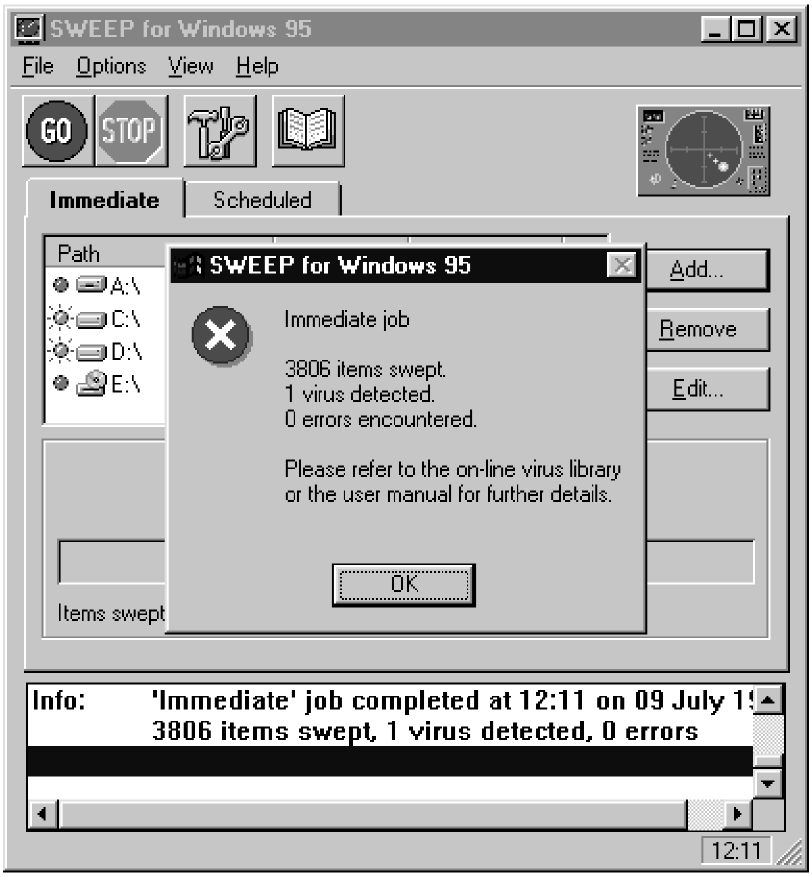
Virus elimination

In order to assist users faced with a virus infection, SWEEP provides a virus elimination facility.

Infected files can be renamed, deleted (reversible), shredded (irreversible), moved or copied.

Infected documents can be disinfected.

Infected boot sectors on most floppy disks can be disinfected (i.e. restored to the exact state prior to infection).



Immediate job

3806 items swept.
1 virus detected.
0 errors encountered.

Please refer to the on-line virus library or the user manual for further details.

Info: 'Immediate' job completed at 12:11 on 09 July 1!
3806 items swept, 1 virus detected, 0 errors

SWEEP for Windows 95 finding a virus

Features

- Checks local hard disks, floppy disks and network drives for the presence of all viruses known to Sophos at the time of release.
- Incorporates a stand-alone InterCheck client for on-access scanning of unknown items.
- Is updated twelve times a year, and urgent updates can be distributed by fax or email or downloaded from the Sophos Web site.
- Easily detects polymorphic viruses using Sophos' advanced Virus Description Language (VDL) and a built-in code emulator.
- Detects and disinfects Microsoft Word, Excel and Office 97 macro viruses.
- Can be installed automatically on multiple workstations from a login script.
- Provides automatic updating for networked PCs.
- Offers two levels of security, allowing a 'quick sweep' which looks for virus identities in the parts of a file that are likely to contain a virus, and a 'full sweep' which looks for virus fragments in every part of a file.
- Features an 'immediate' mode which allows checking on demand, along with a 'scheduled' mode which allows multiple scheduled jobs to be configured for automatic operation.
- Can notify network managers automatically, via Microsoft Exchange, if a virus is found.
- Includes an extensive on-line virus information database.
- Is a 32-bit application and is fully Windows 95 compliant.

Technical details

Product name: Sophos Anti-Virus for Windows 95 Version 3.10 (June 1998).

Function: Scanning individual drives for known viruses.

Detection method: 'Quick' or 'Full' mode.

Mode of operation: Fully Windows 95 compliant 32-bit application with a graphical user interface, extensive on-line virus information database and comprehensive on-line help.

On-access scanning: InterCheck VxD (Virtual Device Driver).

Viruses detected: 15,015 (June 1998)

Updates: 12 per year, sent by post at the beginning of each month, or available for download via WWW.

Virus alerts: SWEEP for Windows 95 is able to generate and print security reports, as well as interface with Microsoft Exchange to send them to system administrators.

Specification of items to be checked: Memory, individual files with specific extensions, files not on exclusion list, and recursive searching of subdirectories.

Virus database: Includes an extensive on-line virus information database.

Sophos Anti-Virus for Windows NT

Incorporating SWEEP and InterCheck

INTERCHECK™
TECHNOLOGY

Virus-specific detection software for Windows NT file servers or workstations.

Description

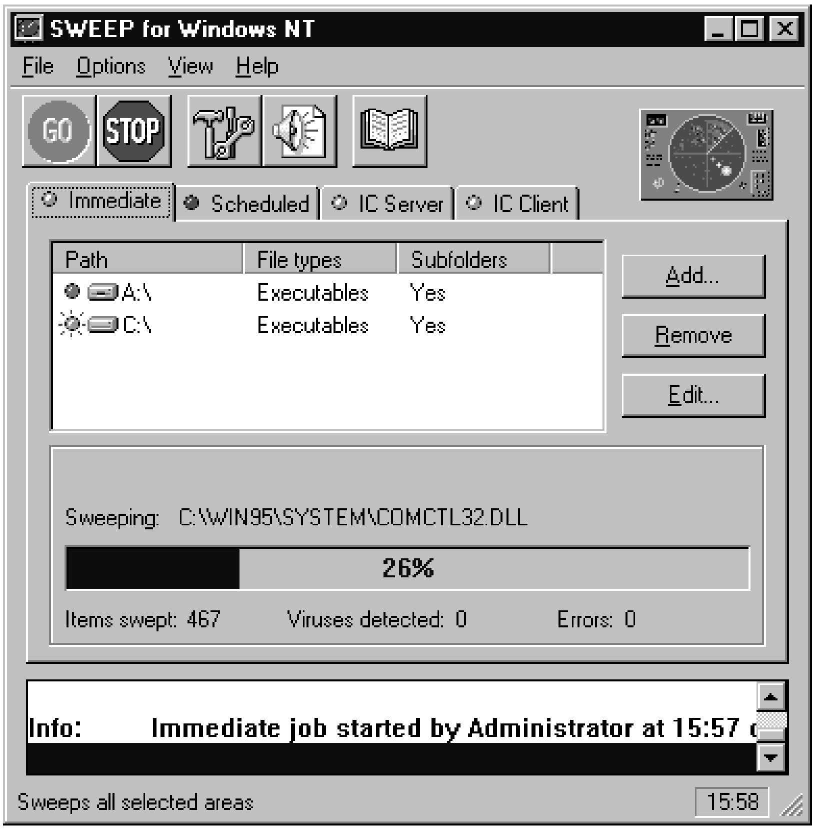
Sophos Anti-Virus for Windows NT is virus detection and disinfection software which can be installed on NT file servers and workstations.

Virus checking

Sophos Anti-Virus for Windows NT incorporates SWEEP and InterCheck.

SWEEP provides on-demand and scheduled virus checking of files on Windows NT file servers or workstations.

The stand-alone InterCheck client provides local on-access virus checking on Windows NT workstations.



Path	File types	Subfolders
A:\	Executables	Yes
C:\	Executables	Yes

Sweeping: C:\WIN95\SYSTEM\COMCTL32.DLL

26%

Items swept: 467 Viruses detected: 0 Errors: 0

Info: Immediate job started by Administrator at 15:57 c

Sweeps all selected areas 15:58

SWEEP for Windows NT sweeping for viruses

InterCheck, using a server installation of SWEEP, provides server based on-access virus checking for networked non-Windows NT workstations.

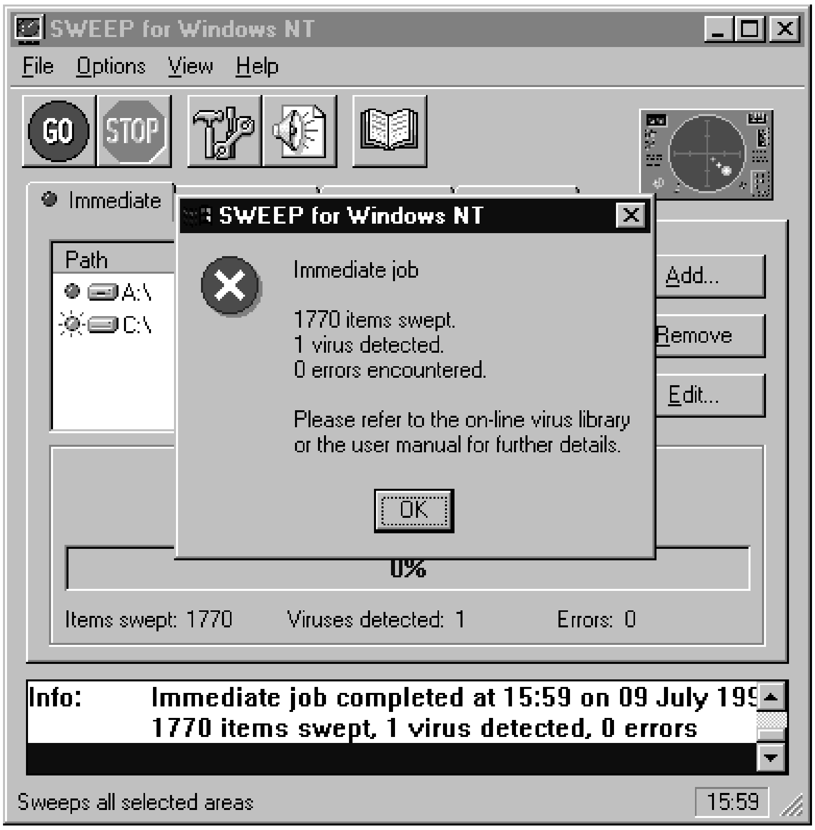
Virus elimination

Sophos Anti-Virus for Windows NT provides a virus elimination facility.

Infected files are either removed or positively overwritten.

Infected documents can be disinfected.

Infected boot sectors on most floppy disks can be disinfected (i.e. restored to the exact state prior to infection).



SWEEP for Windows NT

Immediate job

1770 items swept.
1 virus detected.
0 errors encountered.

Please refer to the on-line virus library or the user manual for further details.

Items swept: 1770 Viruses detected: 1 Errors: 0

Info: Immediate job completed at 15:59 on 09 July 19
1770 items swept, 1 virus detected, 0 errors

Sweeps all selected areas 15:59

SWEEP for Windows NT finding a virus

Features

- Checks local hard disks, floppy disks and networks for the presence of all viruses known to Sophos at the time of release, including Macintosh viruses in files stored on the server.
- Incorporates Sophos' proprietary InterCheck client-server virus detection technology, which allows server based checking of workstations, and includes a stand-alone InterCheck client for local on-access scanning of unknown items.
- Is updated twelve times a year, and urgent updates can be distributed by fax or email or downloaded from the Sophos Web site.
- Easily detects polymorphic viruses using Sophos' advanced Virus Description Language (VDL) and a built-in code emulator.
- Detects and disinfects Microsoft Word, Excel and Office 97 macro viruses.
- Provides automatic updating for networked PCs.
- Offers two levels of security, allowing a 'quick sweep' which looks for virus identities in the parts of a file that are likely to contain a virus, and a 'full sweep' which looks for virus fragments in every part of a file.
- Features an 'immediate mode' which allows checking on demand, along with a 'scheduled mode' which allows multiple scheduled jobs to be configured for automatic operation, even when no-one is logged in to the machine.
- Can notify network managers of the discovery of a virus automatically, via the event log, network messages, and SMTP email.
- Includes an extensive on-line virus information database.
- File server licences include Sophos Anti-Virus for DOS/Windows 3.x and Windows 95.

Technical details

Product name: Sophos Anti-Virus for Windows NT Version 3.10 (June 1998).

Function: Scanning individual workstations and file servers for known viruses.

Mode of operation: Fully 32-bit Windows NT process. In addition to the Graphical User Interface (GUI) version, a Command Line Interface (CLI) version with similar functionality is available.

On-access scanning: InterCheck FSFD (File System Filter Driver).

Viruses detected: 15,015 (June 1998).

Updates: 12 per year, sent by post at the beginning of each month, or available for download via WWW.

Virus alerts: Network managers can be automatically notified by a network message to specified users or machines, by SMTP email or by InterCheck log messages. In addition, the user will be alerted and an entry will be placed in the Windows NT application event log.

Specification of items to be checked: Individual files, recursive searching of subdirectories, individual disk sectors, ranges of disk sectors, and absolute or logical disk sector addressing.

Specification of patterns: Strings of hexadecimal digits, up to 24 bytes per pattern can be specified by the user.

Specification of identities: Additional identities can be added into SWEEP at run-time, either as ASCII text or as binary files.

System requirements: Windows NT Server or Workstation edition, version 3.51 or above. The CLI version requires Windows NT 3.1 or above.

File systems supported: All file systems supported by Windows NT, including NTFS, HPFS and CDFS.

Networks supported: All networks supported by Windows NT, including Microsoft LanManager, Novell NetWare, and TCP/IP.

SAVI

Sophos Anti-Virus Interface

An API that integrates
Sophos Anti-Virus with other
software developers' applications.

Description

SAVI (Sophos Anti-Virus Interface) is an Application Programming Interface that allows software developers to integrate Sophos Anti-Virus with their applications.

It offers third-party developers automated virus checking without the need to call on a command line scanner, giving greatly enhanced performance.

SAVI is an integral part of Sophos Anti-Virus for Windows NT.

Applications

- Email monitoring.
- WWW download monitoring.
- FTP download monitoring.
- Monitoring traffic through firewalls.
- Backup applications.

Features

- Requests to scan files are routed directly to SAVI's Dynamic Link Library (DLL).
- Virus database does not need to be reinitialized each time there is a request for virus-checking.
- Eliminates memory constraints by using a single multi-threading copy of the virus database to process all requests.
- Performance is typically more than ten times better than that of command line scanners, and can be up to thirty times better.
- Intel and AXP platform support.

Technical details

Software name: SAVI.DLL.

Function: External interface to NT Virus Engine.

Functions supported: SweepInit(); initialises DLL.

SweepFile (LPCSTR lpcstrFile, LPCSTR lpcstrReport); call to sweep a file with a report written to report file.

SweepTerm(); terminates use of DLL.

MS-DOS size: 51 Kb.

Further information: Contact Sophos technical support.
Email support@sophos.com.
Telephone +44 1235 559933.

Sophos Anti-Virus Management Tools

SWDEPLOY, SAVADMIN, SWCONSOL, SGET

Utilities to facilitate the deployment and use of Sophos Anti-Virus in large organisations.

SWDEPLOY

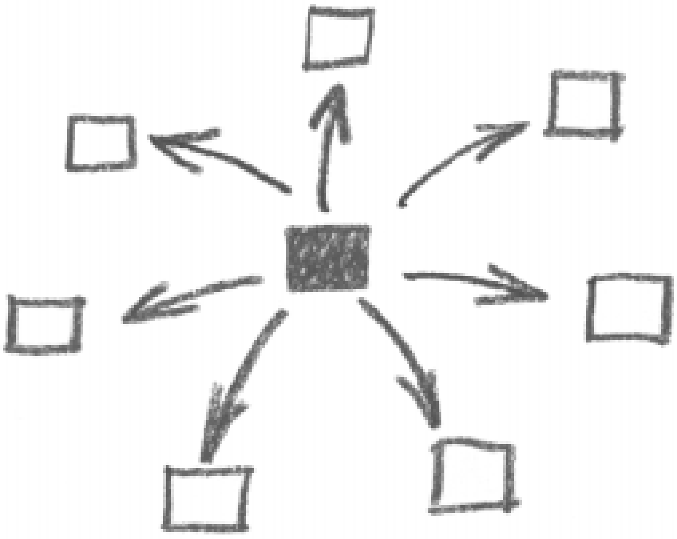
For installing Sophos Anti-Virus on Windows NT networks

SWDEPLOY is a command line utility which makes it possible to install and configure Sophos Anti-Virus across an entire NT network from a central NT workstation.

All workstations and servers can then be updated automatically from this central point.

Features

- Is a 'minimal push' tool, i.e. during inital installation, it copies the minimum of components to the client to enable a 'pull' installation from server to client.
- Makes it possible to monitor remote Sophos Anti-Virus installations.
- Integrates with third-party management tools.



SAVADMIN

For installing Sophos Anti-Virus on Windows NT networks

SAVADMIN is a GUI (Graphical User Interface) version of SWDEPLOY.

It makes it possible to install and configure Sophos Anti-Virus across an entire network from a central NT workstation.

All workstations and servers can then be updated automatically from this central point.

Features

- Intuitive user interface allows installations and emergency upgrades to individual machines, or entire domains, in a single operation.
- Is a ‘minimal push’ tool, i.e. during initial installation, it copies the minimum of components to the client to enable a ‘pull’ installation from server to client.
- Makes it possible to monitor remote Sophos Anti-Virus installations.
- Integrates easily with other management packages, such as SMS (Microsoft Systems Management Server).

SWCONSOL

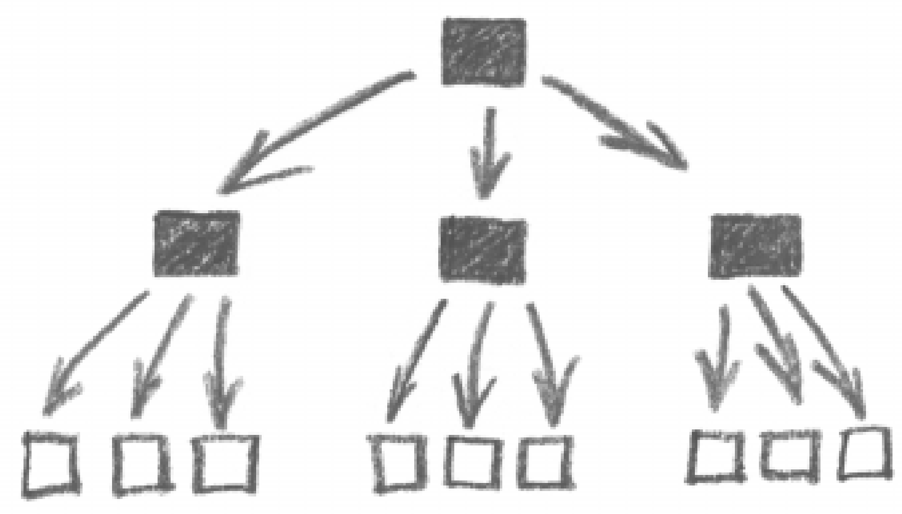
For remote control of NetWare servers

SWCONSOL is a general-purpose utility for script-based remote control of NetWare servers.

Unlike RCONSOLE, which allows interactive operation of a remote server, it is designed to operate under remote programmatic control, making it ideal for automating server operations across a multi-server network. SWCONSOL runs as an NLM on a server that needs to be controlled remotely. It watches for a control script to arrive, runs it, deletes it and goes back to watching for the next script file.

Features

- Allows invocation of console commands, such as loading or unloading other NLMs or shutting down the server.
- Enables Sophos Anti-Virus to be deployed and updated automatically over any number of servers.
- Verifies the integrity of files, making it easy to load only uncorrupted NLMs.
- Produces a log file.



SGET

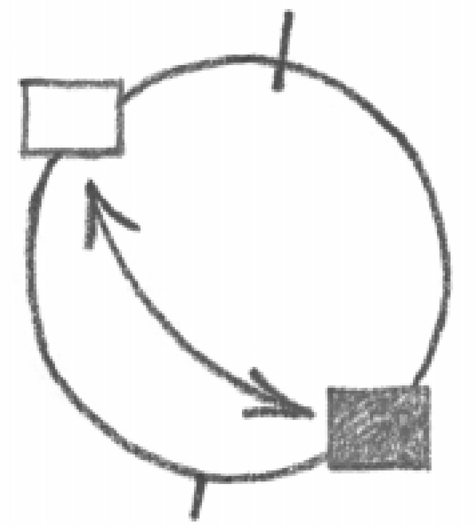
For automated updating of Sophos Anti-Virus over the Internet

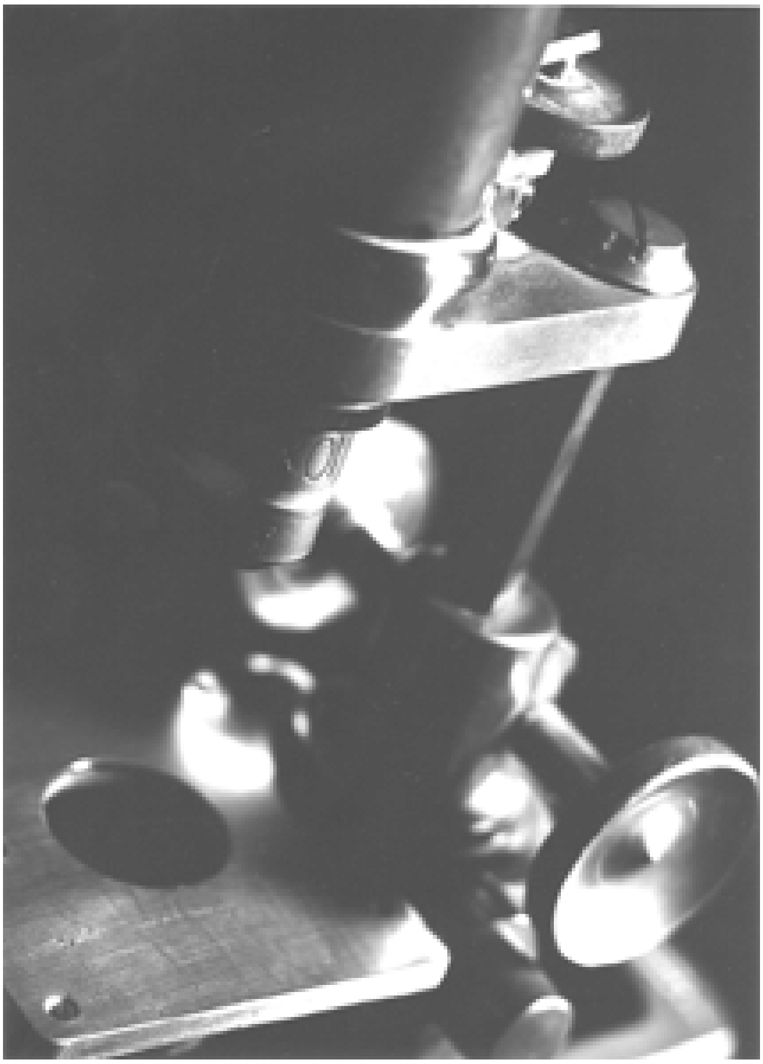
SGET allows network administrators to automate the updating of Sophos Anti-Virus products from the Sophos Web site.

Updating is initiated when a newer version of Sophos Anti-Virus than the previously downloaded one becomes available on the Web site.

Features

- Can be used for automatic updating of Sophos Anti-Virus for any platform.
- Is a pull-based system, i.e. updating is controlled by the user.
- Can be run from a scheduler or from within batch and command files.
- Is a 32-bit Windows 95/Windows NT 4.0 console mode application.





Sophos Utilities

A utility package for
dealing with a virus attack.

Description

Sophos Utilities (SU) is a package for dealing with a virus attack. The user can view absolute sectors, copy sectors, search files for a pattern, save and restore master and DOS boot sectors into and from a file, and also display the interrupt table.

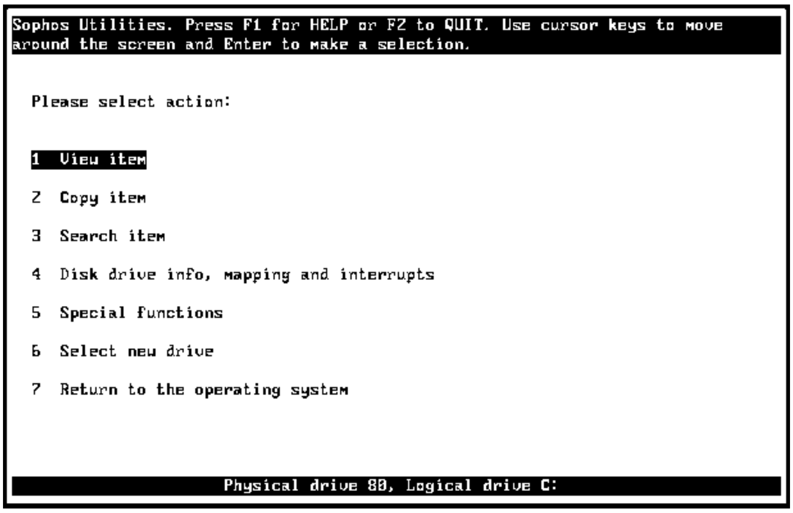
SU is not intended as a complete replacement for utility packages such as the Norton Utilities or PC Tools. However, unlike these disk management packages, SU was designed from the outset to deal with the virus problem and enables various operations to be done faster and more easily.

SU is safe to use. In its default mode, users can only overwrite files (the user is given a warning before an existing file is overwritten). More 'dangerous' operations such as copying absolute sectors or clusters, must be specifically enabled.

SU is menu-driven with on-line help.

Functions

Viewing items Absolute sectors, logical sectors, clusters, or files. Each item can be viewed in six modes: as a



```
Sophos Utilities. Press F1 for HELP or F2 to QUIT. Use cursor keys to move
around the screen and Enter to make a selection.

  Please select action:

  1  View item
  2  Copy item
  3  Search item
  4  Disk drive info, mapping and interrupts
  5  Special functions
  6  Select new drive
  7  Return to the operating system

                      Physical drive 80, Logical drive C:
```

SU initial menu

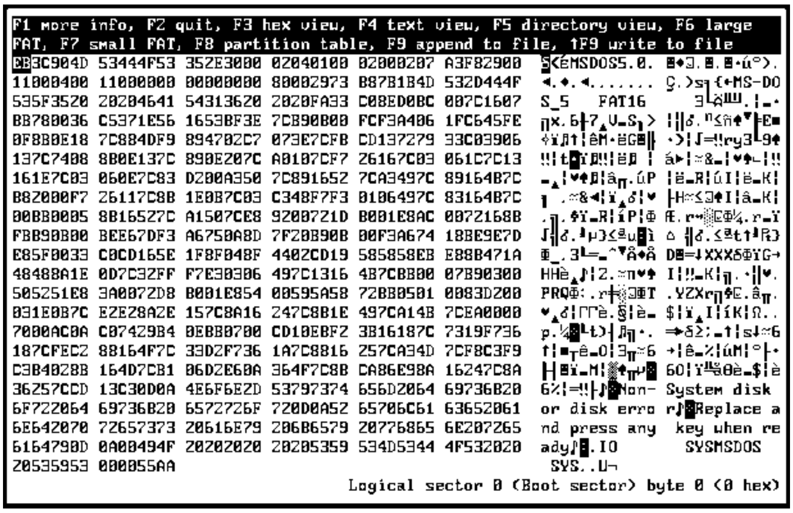
hexadecimal dump, as text, as a directory, as a large (16-bit) FAT, as a small (12-bit) FAT and as a partition table. The default mode is selected automatically according to which part of the disk you are looking at. For example, if you look at head 0, cylinder 0, sector 1 of a hard disk, SU will display the sector as a partition table. If you look at logical sector 1, SU will present the sector in the FAT format (on most disks).

Copying items Absolute sectors, logical sectors, clusters, or files can be copied into any other item.

Copying is done in units of one sector (512 bytes), so if a file containing 1 byte is copied into another file, the destination file will be 512 bytes long.

Searching items Ranges of absolute sectors, logical sectors, clusters, one file, or all files on the selected drive can be searched for a pattern. The pattern can be specified in ASCII or in hexadecimal.

When the search pattern is found, SU displays the item in which it occurs and positions the cursor at the start of the pattern. When the whole search area has been checked, SU will display the number of occurrences of the pattern.



```
F1 more info, F2 quit, F3 hex view, F4 text view, F5 directory view, F6 large
FAT, F7 small FAT, F8 partition table, F9 append to file, ↑F9 write to file
EB3C904D 53444F53 352E3000 02040100 02000207 A3F82900   δ<éMSDOS5.0. ☻♦♥.☻.☻·ú°).
11000400 11000000 00000000 80002973 B87B1B4D 532D444F   ◄.♦.◄....... Ç.)s╕{←MS-DO
535F3520 20204641 54313620 2020FA33 C08ED0BC 007C1607   S_5   FAT16   ·3└Ä╨╝.|▬•
BB780036 C5371E56 1653BF3E 7CB90B00 FCF3A406 1FC645FE   ╗x.6┼7▲V▬S┐> |╣♂.ⁿ≤ñ♠▼╞E■
0F8B0E18 7C884DF9 894702C7 073E7CFB CD137279 33C03906   ☼ï♫↑|êM∙ëG☻╟•>|√═‼ry3└9♠
137C7408 8B0E137C 890E207C A0107CF7 26167C03 061C7C13   ‼|t◘ï♫‼|ë♫ | á►|≈&▬|♥♠∟|‼
161E7C03 060E7C83 D200A350 7C891652 7CA3497C 89164B7C   ▬▲|♥♠♫|â╥.úP|ë▬R|úI|ë▬K|
B82000F7 26117C8B 1E0B7C03 C348F7F3 0106497C 83164B7C   ╕ .≈&◄|ï▲♂|♥├H≈≤☺♠I|â▬K|
00BB0005 8B16527C A1507CE8 9200721D B001E8AC 0072168B   .╗.♣ï▬R|íP|Φ Æ.r↔░☺Φ¼.r▬ï
FBB90B00 BEE67DF3 A6750A8D 7F20B90B 00F3A674 18BE9E7D   √╣♂.╛µ}≤ªu◙ì⌂ ╣♂.≤ªt↑╛₧}
E85F0033 C0CD165E 1F8F048F 4402CD19 585858EB E88B471A   Φ_.3└═▬^▼Å♦ÅD☻═↓XXXδΦïG→
48488A1E 0D7C32FF F7E30306 497C1316 4B7CBB00 07B90300   HHè▲♪|2.≈π♥♠I|‼▬K|╗.•╣♥.
505251E8 3A0072D8 B001E854 00595A58 72BB0501 0083D200   PRQΦ:.r╪░☺ΦT.YZXr╗♣☺.â╥.
031E0B7C E2E28A2E 157C8A16 247C8B1E 497CA14B 7CEA0000   ♥▲♂|ΓΓè.§|è▬$|ï▲I|íK|Ω..
7000AC0A C07429B4 0EBB0700 CD10EBF2 3B16187C 7319F736   p.¼◙└t)┤♫╗•.═►δ≥;▬↑|s↓≈6
187CFEC2 88164F7C 33D2F736 1A7C8816 257CA34D 7CF8C3F9   ↑|■┬ê▬O|3╥≈6→|ê▬%|úM|°├∙
C3B4028B 164D7CB1 06D2E60A 364F7C8B CA86E98A 16247C8A   ├┤☻ï▬M|▒♠╥µ◙6O|ï╩åΘè▬$|è
36257CCD 13C30D0A 4E6F6E2D 53797374 656D2064 69736B20   6%|═‼├♪◙Non-System disk
6F722064 69736B20 6572726F 720D0A52 65706C61 63652061   or disk error♪◙Replace a
6E642070 72657373 20616E79 206B6579 20776865 6E207265   nd press any key when re
6164790D 0A00494F 20202020 20205359 534D5344 4F532020   ady♪◙.IO      SYSMSDOS
20535953 000055AA                                        SYS..U¬
                                   Logical sector 0 (Boot sector) byte 0 (0 hex)
```

Viewing DOS boot sector in hexadecimal form

Disk drive information and interrupts

This option gives the information about the current drive as well as the contents of the interrupt table.

'Disk drive technical info' gives information on both DOS and physical characteristics of the current drive.

'Disk drive sector map' indicates which sectors are used for the disk FAT(s), root directory and file areas.

'Disk drive usage map' displays a map of which parts of the file area of the current disk are in use.

'Interrupt table' displays information about software interrupts which includes a brief description of each interrupt, the vector for the interrupt and the first 8 bytes addressed by the vector. In addition, checksums of the interrupt vectors and the first 8 bytes are displayed for interrupts 0 to 3F hex as well as for interrupts 0 to FF hex.

Special functions

The master boot sector and the DOS boot sector can be saved in a file and individually restored from a file. These are the two sectors that will have to be restored in the event of a boot sector virus attack. If a clean copy of these sectors is kept, then all known boot sector viruses can be removed simply by restoring these copies.

The file containing the boot sectors has a checksum appended to it.

Applications

- Recovery from a virus attack.
- Examining disk contents.
- Searching for information on all sectors of a disk.
- Inspecting of interrupts.
- Saving and restoring master and DOS boot sectors.

Technical details

Product name: Sophos Utilities (SU) Version 1.07 (March 1995).

Function: A utility package for dealing with a virus attack.

Mode of operation: Interactive or command-line driven.

Functions: Viewing absolute sectors, logical sectors, clusters, or files; as a hexadecimal dump, as text, as a directory, as a large (16-bit) FAT, as a small (12-bit) FAT or as a partition table.

Copying absolute sectors, logical sectors, clusters, or files into any other item.

Searching ranges of absolute sectors, logical sectors, clusters, one file, or all files on the selected drive for a pattern. The pattern can be specified in ASCII or in hexadecimal.

Displaying of disk drive information and interrupts as a sector map, disk drive usage map and the interrupt table.

Saving the master boot sector and/or the DOS boot sector in a file, and restoring them.

VACCINE

Anti-virus system

A checksumming virus-non-specific detection system for IBM PC compatibles and networks.

Description

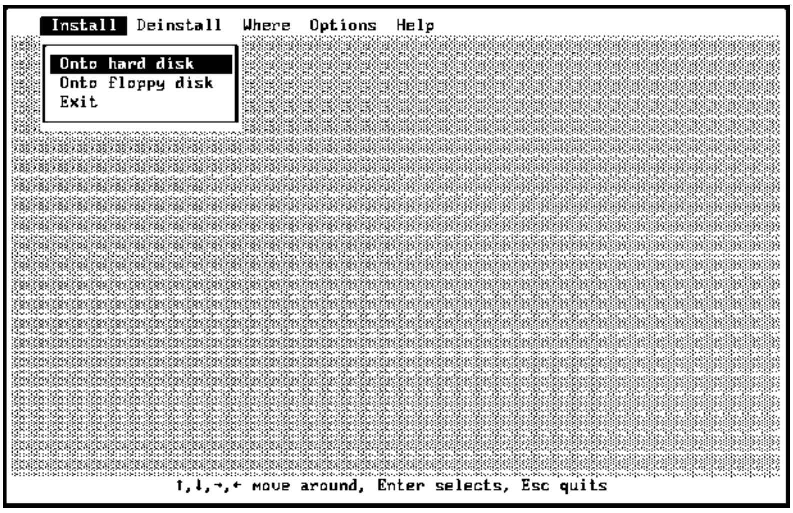
VACCINE is used to monitor the integrity of executable items and data on a PC. Viruses necessarily modify executable code, so the modification of one or more executable items can be used to detect the presence of a virus.

The main advantage of this approach is that **any** virus can be detected, without the need to update VACCINE. The main disadvantage is that the user will be notified of the change, but not which virus (if any) has caused it.

VACCINE is also a powerful tool for auditing and for version control; it can be used to detect even a single bit change to a program, data file, disk sector or memory region, as well as changes to the overall configuration of a system, such as the existence or non-existence of particular files or subdirectories.

The VACCINE system consists of 4 modules:

- SWEEP virus-scanning module for checking disks for the presence of all known viruses. This module has a limited life-span: any viruses which were not



↑,↓,→,← move around, Enter selects, Esc quits

VACCINE INSTALL module main screen

in existence when a particular version was released will not be detected.

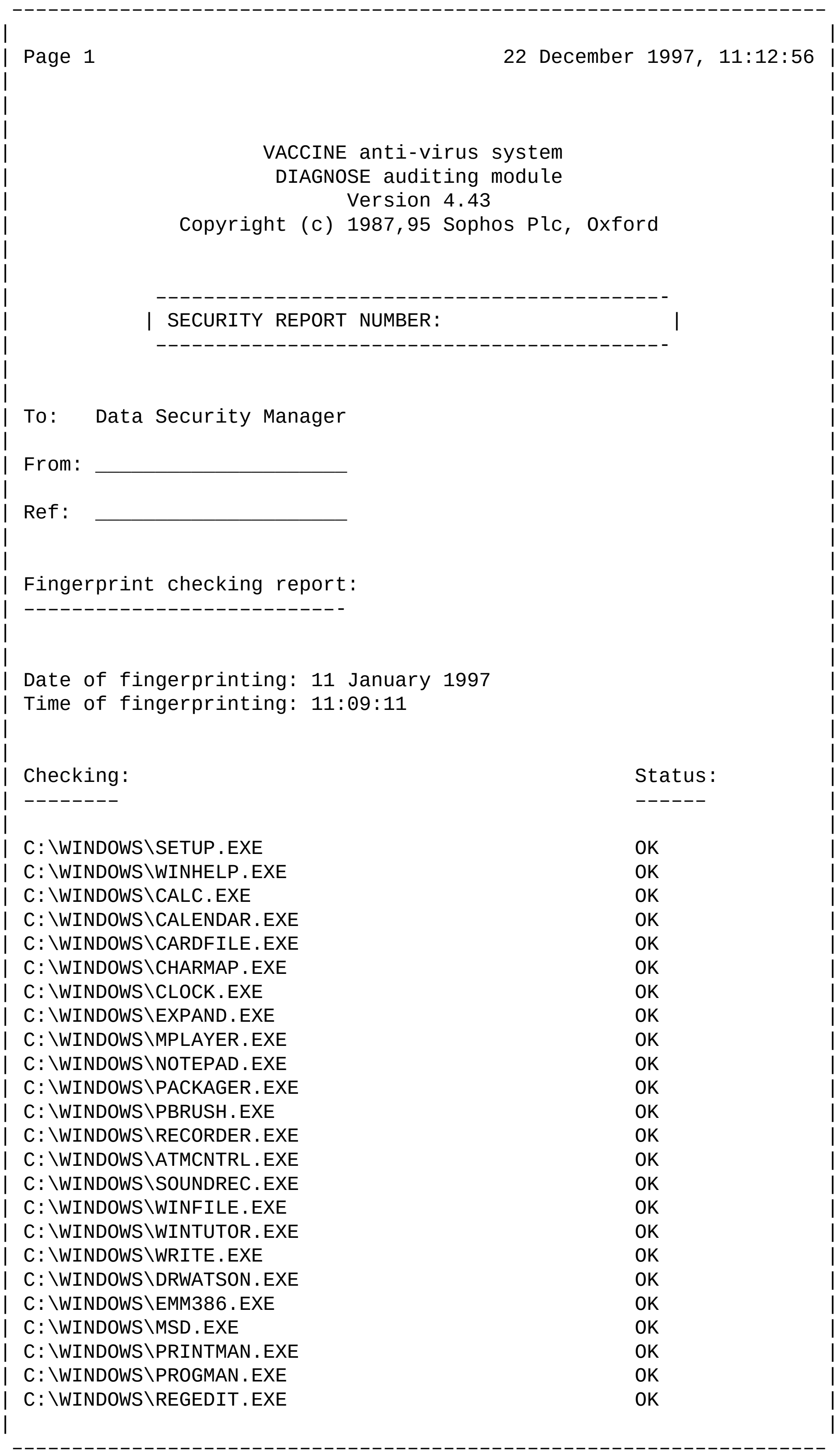
- VACCINE system fingerprinting module. Used to take fingerprints of the system.
- DIAGNOSE system checking module. Used to check the integrity of fingerprinted files, disk sectors and memory regions.
- FILEMAC file checking utility module. Used to check the integrity of individual files.

The user should start with a ‘clean’ computer system on which all files are genuine and free of viruses. In most cases it would be inconvenient or impractical to completely initialise the system disk, so the SWEEP module is provided to check the system for all known viruses instead.

Having established that the system is ‘clean’, VACCINE is used to fingerprint all critical files (such as executable files), disk sectors (such as boot sectors) and memory regions (such as the interrupt table) on the system. The fingerprints are stored in encrypted form in the file DIAGNOSE.FIN.

The DIAGNOSE module can then be used at regular intervals to check the fingerprints. Any modification to the fingerprinted files, or any changes to fingerprinted sections or memory regions will be detected and reported by the DIAGNOSE module.

The FILEMAC module gives direct access to the actual numerical values of the fingerprints, and is useful in a variety of ways. It provides a further method of checking the DIAGNOSE module as well as offering a flexible tool for incorporating fingerprinting techniques into batch or command files. It supports individual or group fingerprints, recursive searching, I/O redirection, and optional initialisation vectors for the SPA or ISO 8731 (Part 2) algorithm. With FILEMAC it is also possible to implement sophisticated challenge-response mechanisms.



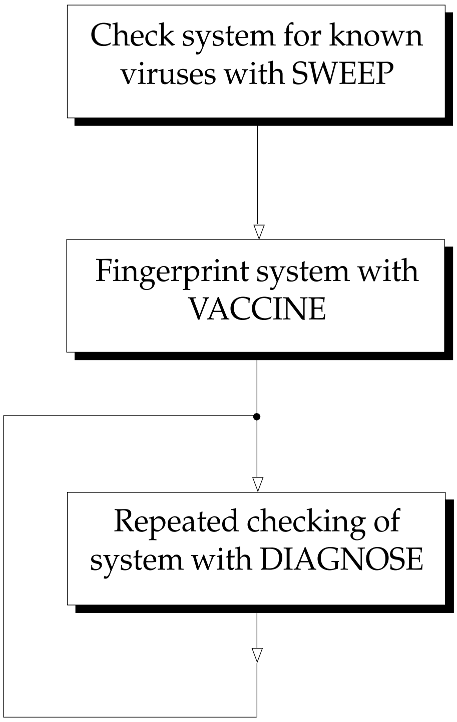
```
 --------------------------------------------------------------------
|                                                                    |
| Page 1                                  22 December 1997, 11:12:56 |
|                                                                    |
|                                                                    |
|                                                                    |
|                     VACCINE anti-virus system                      |
|                      DIAGNOSE auditing module                      |
|                            Version 4.43                            |
|              Copyright (c) 1987,95 Sophos Plc, Oxford              |
|                                                                    |
|                                                                    |
|            -------------------------------------------             |
|           | SECURITY REPORT NUMBER:                    |            |
|            -------------------------------------------             |
|                                                                    |
|                                                                    |
| To:   Data Security Manager                                        |
|                                                                    |
| From: _____________________                                        |
|                                                                    |
| Ref:  _____________________                                        |
|                                                                    |
|                                                                    |
| Fingerprint checking report:                                       |
| ---------------------------                                        |
|                                                                    |
|                                                                    |
| Date of fingerprinting: 11 January 1997                            |
| Time of fingerprinting: 11:09:11                                   |
|                                                                    |
|                                                                    |
| Checking:                                          Status:         |
| --------                                           ------          |
|                                                                    |
| C:\WINDOWS\SETUP.EXE                               OK              |
| C:\WINDOWS\WINHELP.EXE                             OK              |
| C:\WINDOWS\CALC.EXE                                OK              |
| C:\WINDOWS\CALENDAR.EXE                            OK              |
| C:\WINDOWS\CARDFILE.EXE                            OK              |
| C:\WINDOWS\CHARMAP.EXE                             OK              |
| C:\WINDOWS\CLOCK.EXE                               OK              |
| C:\WINDOWS\EXPAND.EXE                              OK              |
| C:\WINDOWS\MPLAYER.EXE                             OK              |
| C:\WINDOWS\NOTEPAD.EXE                             OK              |
| C:\WINDOWS\PACKAGER.EXE                            OK              |
| C:\WINDOWS\PBRUSH.EXE                              OK              |
| C:\WINDOWS\RECORDER.EXE                            OK              |
| C:\WINDOWS\ATMCNTRL.EXE                            OK              |
| C:\WINDOWS\SOUNDREC.EXE                            OK              |
| C:\WINDOWS\WINFILE.EXE                             OK              |
| C:\WINDOWS\WINTUTOR.EXE                            OK              |
| C:\WINDOWS\WRITE.EXE                               OK              |
| C:\WINDOWS\DRWATSON.EXE                            OK              |
| C:\WINDOWS\EMM386.EXE                              OK              |
| C:\WINDOWS\MSD.EXE                                 OK              |
| C:\WINDOWS\PRINTMAN.EXE                            OK              |
| C:\WINDOWS\PROGMAN.EXE                             OK              |
| C:\WINDOWS\REGEDIT.EXE                             OK              |
|                                                                    |
 --------------------------------------------------------------------
```

A typical report produced by DIAGNOSE

```
 --------------------------------------------------------------------
|                                                                    |
| Page 2                                  22 December 1997, 11:12:56 |
|                                                                    |
| C:\WINDOWS\SMARTDRV.EXE                            OK              |
| C:\WINDOWS\TERMINAL.EXE                            OK              |
| C:\WINDOWS\CLIPBRD.EXE                             OK              |
| C:\WINDOWS\CONTROL.EXE                             OK              |
| C:\WINDOWS\PIFEDIT.EXE                             OK              |
| C:\WINDOWS\TASKMAN.EXE                             OK              |
| C:\WINDOWS\WCURSOR.EXE                             OK              |
| C:\WINDOWS\WINVER.EXE                              OK              |
| C:\WINDOWS\WIN.COM                                 OK              |
| C:\WINDOWS\MOUSE.COM                               OK              |
| C:\SORT.EXE                                        OK              |
| C:\VDLCOMP.EXE                                     OK              |
| C:\VDLTOIDE.EXE                                    OK              |
| C:\SB.EXE                                          OK              |
| C:\SWEEPO.EXE                                      OK              |
| C:\SWEEPBIN.EXE                                    OK              |
| C:\SWEE.EXE                                        OK              |
| C:\SWEEP.EXE                                       OK              |
| C:\COMPRS.EXE                                      OK              |
| C:\FILEMAC.EXE                                     OK              |
| C:\DEDUPE1.EXE                                     OK              |
| C:\VIRPATS.EXE                                     OK              |
| C:\FILTER9.EXE                                     OK              |
| C:\SHRED.EXE                                       OK              |
| C:\IBMBIO.COM                                      OK              |
| C:\IBMDOS.COM                                      OK              |
| C:\COMMAND.COM                                     OK              |
| C:|0                                               OK              |
| +80 0 0 1                                          OK              |
|                                                                    |
| 51 files were checked.                                             |
|                                                                    |
| 2 sector ranges were checked.                                      |
|                                                                    |
| 0 memory ranges were checked.                                      |
|                                                                    |
| All fingerprints were OK.                                          |
|                                                                    |
|                                                                    |
|                                                                    |
|                                                                    |
|                                                                    |
|                            Signature: ________________________     |
|                                                                    |
|                                                                    |
| END OF SECURITY REPORT                                             |
|                                                                    |
|                                                                    |
|                                                                    |
|                                                                    |
|                                                                    |
|                                                                    |
|                                                                    |
|                                                                    |
|                                                                    |
 --------------------------------------------------------------------
```

Features

- Any part of the system can be included in the check, such as program files, batch files, data files, absolute and logical disk sectors (including groups of bytes within the sectors), and memory locations such as the interrupt table.
- Fingerprinting is normally performed using the Sophos Proprietary Algorithm (SPA), a high speed authentication algorithm based on ANSI X9.9. However, the ISO standard 8731 (Part 2) algorithm can be selected instead.
- Complex one-way cryptographic fingerprints are calculated. Any modification made to an item causes its fingerprint to change completely. It is practically impossible to 'engineer' changes in such a way as to leave the fingerprint unaffected.

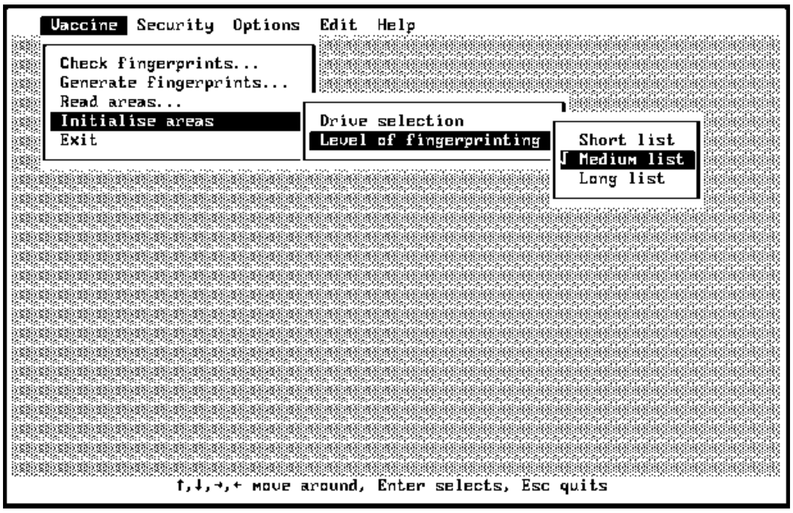

Check system for known viruses with SWEEP
Fingerprint system with VACCINE
Repeated checking of system with DIAGNOSE

Using VACCINE

- The fingerprint storage is tamper-protected and can be customised using a password and a reverse password specified by the user, which is combined with a sophisticated use of the SPA algorithm to protect the fingerprints. This allows DIAGNOSE to identify the fingerprints positively as being genuine when the system is checked.
- DIAGNOSE produces detailed printed security reports, with checklists for action in the event of any irregularity.
- The VACCINE and INSTALL modules are menu-driven, mouse-aware and have on-screen help.

Applications

- Detection of computer viruses and other attacks.
- Software version control.
- Detection of unauthorised software modifications.
- Configuration control in software production.



↑,↓,→,← move around, Enter selects, Esc quits

Fingerprinting level selection in VACCINE

- Computer systems auditing.
- Checking the integrity of payroll files.
- Ensuring authenticity of spreadsheet templates.
- Control of CAD projects by ‘freezing’ designs.

Technical details

Product name: VACCINE Version 4.43 (March 1995).

Function: Anti-virus system.

Mode of operation: Interactive or batch-driven. SWEEP module searches for known viruses, VACCINE module takes initial cryptographic fingerprints of data items, DIAGNOSE module performs subsequent checks, FILEMAC module gives access to numeric fingerprint values.

Fingerprinting method: ANSI X9.9 using SPA, or ISO 8731/2, user-selectable. Fingerprints are stored tamper-protected using encryption and optional reverse password.

Encryption method: Proprietary SPA algorithm encryption of data. DES option available. Natural language or hexadecimal key specification. Keys can be up to 64 characters. 16-digit hexadecimal keys are automatically recognised and used directly; all others are compressed to 64 bits for SPA or DES.

Printed security report: A security report can be printed, including an action checklist in the event of irregularities.

Irregularities detected: Any change to file contents, any change to file attributes, the appearance of new files, the disappearance of specified files, any change to disk sector contents, and any change to specified memory locations.

Specification of fingerprinted items: Files; individual files and recursive searching of subdirectories. The attributes and presence of a file can be checked.

Disk sectors; individual sectors, ranges of sectors, ranges of bytes within sectors. Absolute or logical sector addressing.

Memory areas; individual bytes and ranges of bytes. Memory pages can be specified.

Education and Training

Education and training for computer users
on anti-virus and data security topics.

Anti-virus workshops

Information technology systems are increasingly being threatened by computer viruses. Too few people contemplate the potentially disastrous effects of the disruption to their computer systems caused by a virus attack.

Personal Computers (PCs) are particularly vulnerable to virus attacks, because the programs on PCs are relatively unprotected.

Most computer users have no opportunity of seeing a virus until they themselves suffer an attack. Anti-Virus Workshops offer an ideal opportunity to gain **hands-on experience of a viral attack in a controlled environment.** Tuition on dealing with an attack is also given. Personal supervision (**1 PC per participant, 16 participants maximum**) ensures the maximum benefit to the delegates. Due to the sensitive nature of the workshop materials, we impose simple but strict physical controls, to ensure that no viruses leave the premises.

The Workshops are organised as a two-day course comprising the **Introductory Workshop** and **Advanced Workshop**. These can be attended either separately or together.

The **Introductory** and **Advanced Workshops** are particularly recommended for:

- Data processing managers and security managers.
- Office automation specialists.
- PC users.
- Administration managers.
- LAN managers.
- Security consultants.
- IT department advisory staff.

For **Introductory Workshops**, a knowledge of PC basics (such as use of commands like DIR) is assumed.

For **Advanced Workshops**, it is assumed that delegates will have a good understanding of the PC, including basic knowledge of the use of tools such as Norton Utilities and standard DOS commands. A brief overview of the relevant tools and commands is given before the practical sessions.

Specialist personnel assist the participants throughout the practical sessions.

Workshops are run in the Sophos training centre in Abingdon near Oxford on a regular basis; **in-house events** are offered to individual organisations.

Anti-virus seminars for software producers

PC viruses are transmitted in executable code - COM and EXE files, as well as boot sectors on floppy disks. **Software developers are particularly vulnerable to becoming an efficient instrument in virus propagation,** because they originate executable files for widespread distribution. An undiscovered virus at any stage of software production can be distributed inadvertently to hundreds or thousands of users. This has been demonstrated in the past on several occasions.

One of the defences against a virus attack is the use of ‘shrink-wrapped’ software. This means that PC users rely on reputable software producers to supply them with ‘clean’, virus-free code. A heavy burden of responsibility lies on software producers to ensure that this actually happens.

The seminar is particularly recommended for:

- Software houses writing PC applications.
- Computer magazines carrying disks on front covers.

- Individual software developers.
- Software production supervisors.
- System analysts and programmers.
- Disk duplication subcontractors.
- Security consultants.
- IT department advisory staff.
- Quality managers.

We assume an understanding of the PC, including knowledge of the use of standard DOS commands.

Seminars are run in the Sophos training centre in Abingdon near Oxford on a regular basis; **in-house events** are offered to individual organisations.

Practical NetWare Security workshop

Networks are not only increasingly complex, but also increasingly critical to the core business processes within an organisation. This means that today's networks must not fail, even though their growing complexity offers ever more potential failure points.

Good security is the key to the fundamentals of a successful network: availability, confidentiality and integrity. Unfortunately, network administrators are usually forced to spend nearly all of their time dealing with the day-to-day problems experienced during 'normal' network operation. This means that they often have to wait until they suffer a real security breach before they are able to evaluate and manage the risk.

The **Practical NetWare Security** workshop offers the opportunity to experience actual network attacks and defence in a realistic but controlled environment. Rather than simply learning about theoretical security issues in a series of lectures, participants work in a live multi-server network domain using NetWare 3.x and 4.x.

Participants finish the workshop with active, real-world experience, which could only otherwise be obtained as the result of an actual disaster.

Practical NetWare Security is particularly recommended for:

- Security managers in organisations where Novell network servers are in use.
- Administrators of NetWare networks.
- Security consultants.

An understanding of the fundamentals of NetWare networking is assumed. However, a brief overview of the NetWare tools (for server versions 3.x and 4.x) used during the course is given, and specialist trainers assist the participants throughout so that unfamiliarity with specific NetWare applications will not distract participants from the core components of the workshop.

Seminars are run in the Sophos training centre in Abingdon near Oxford on a regular basis; **in-house events** are offered to individual organisations.

Video 'Viruses on Personal Computers'

To prevent a virus attack, all Personal Computer users should receive guidance and training about the threat. Unfortunately, time is often short and personnel may not be available to teach correct anti-virus procedures.

'Viruses on Personal Computers' is a **training video** which can be shown to all PC users in an organisation. It is easy to follow, humorous, and shows clearly what should be done to avoid virus infection.

The video consists of two parts, each approximately 15 minutes long:

Part 1 Explains the problem, demonstrates some possible effects of a virus attack and shows users what to do and what to avoid. This part is appropriate for:

- Personal Computer users.
- Sales personnel.
- **All new employees**.

Part 2 Deals with virus prevention, detection and containment and is suitable for:

- Office managers.
- Personal Computer support specialists.
- Computer auditors.
- Data processing managers.
- Security officers.
- Training managers.
- General managers.

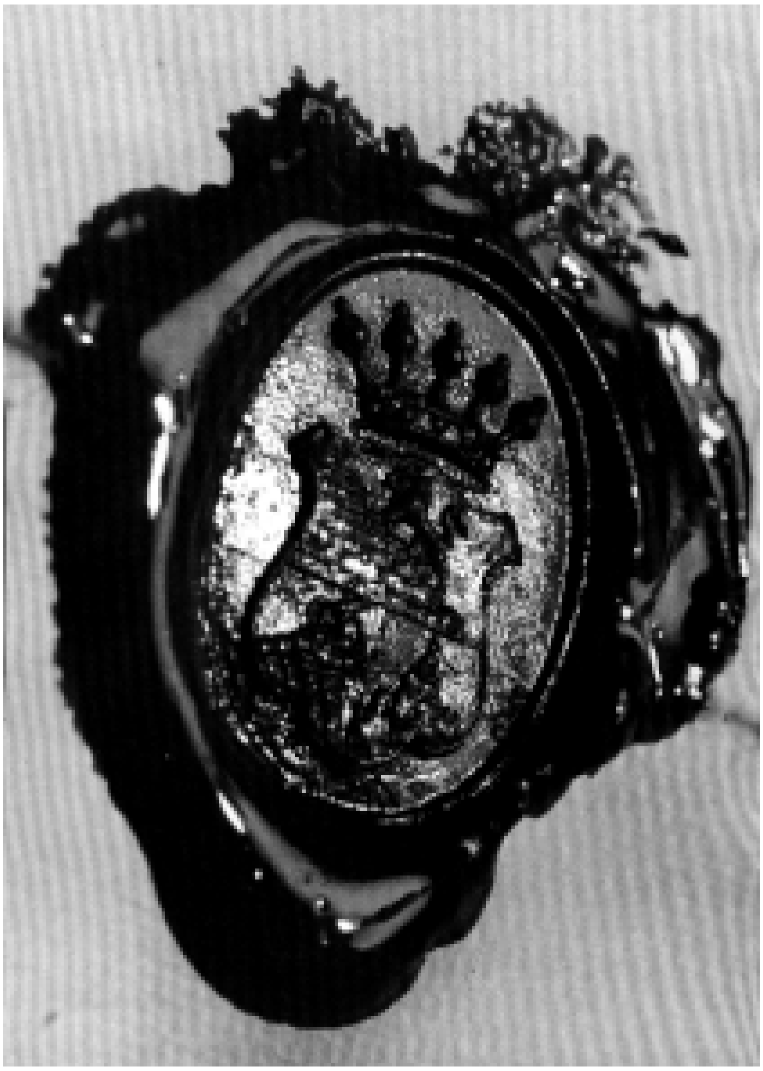
Parts 1 and 2 can be viewed separately or in succession.

The video package includes 10 **course handbooks**, which summarise the material covered in the video, as well as giving additional information.

The video explodes popular myths about viruses and gives practical guidance. It is an invaluable tool for training managers and even temporary staff can be shown the correct anti-virus procedures.

The video is available in **English**, **Dutch, Finnish, French**, **German, Italian, Norwegian, Polish** and **Slovenian** in various formats such as PAL and NTSC.

Additional information 5



Security standards and legislation

This appendix gives details of the following legislation:

- The UK Data Protection Act 1984.
- The EU Data Protection Directive (forthcoming).
- The UK Computer Misuse Act 1990.

It also summarises the following security evaluation criteria and standards:

- The US Department of Defense 'Orange Book' ('Trusted Computer Systems Evaluation Criteria (TCSEC)').
- UK security product certification.
- The Information Technology Security Evaluation Criteria (ITSEC).
- BS7799: British Standard for information security.

UK Data Protection Act 1984

Important! At the time of writing, the UK Data Protection Act is still in force. However, the EU Data Protection Directive will be implemented during 1998, extending data protection in particular areas. See below for further details.

If you process personal data on a computer in the UK, whether on your own equipment or through a bureau, the Data Protection Act 1984 applies to you.

Personal data is defined as ‘information about a living individual, including expressions of opinion about him or her, but excluding any indication of the intentions of the data user in respect of that individual’.

Manual or paper records are not within the definitions. Word processing information is excluded only if used for the preparation of texts or documents. Information such as details of earnings, deductions, loans, benefits, pensions, services provided or training and education records are, however, within the Act.

Failure to comply with the Act can result in unlimited fines, being sued for compensation and being forced to stop using data.

Complying with the Act

To comply, you should do the following:

- Appoint a Data Protection Co-ordinator.
- Inform your staff of the implications of the Act for your organisation.
- Survey all automatic data processing systems within your organisation.
- Decide which data processing systems will require registration.
- Register with the Data Protection Registrar (see details below).
- Familiarise your staff dealing with personal data with their obligations under the Act.
- Set up a procedure for subject access to data.
- Set up security procedures for data protection.
- Establish a procedure for maintaining registration details.

The Data Protection Registrar

To register with the Data Protection Registrar call the registration line on 01625 545740. The registration fee is £75 and registration is valid for three years.

A registration form is available free, together with 'Guidelines on the Act'. Leaflets, posters, information packs, stickers and bookmarks are also available free.

The Registrar's office will answer enquiries on the telephone, by fax, email or mail. Contact:

The Data Registrar
Wycliffe House
Water Lane
Wilmslow
Cheshire SK9 5AF
Tel (01625) 545700 Fax (01625) 524510.
Email data@wycliffe.demon.co.uk.

The Data Protection Act 1984 - ISBN 0 10 543 5848 is available from HMSO.

Data protection principles

The Data protection principles as set out in the 1984 Data Protection Act are:

- Obtain and process personal data fairly and lawfully.
- Hold it only for the purposes specified in your register entry.
- Use it only for the purposes, and disclose it only to the people, listed in your register entry.
- Hold only data which is adequate, relevant and not excessive in relation to the purpose for which it is held.
- Ensure personal data is accurate and where necessary, kept up to date.
- Hold it for no longer than necessary.

- Allow individuals access to information held about them and, where appropriate, correct it or erase it.
- Take security measures to prevent unauthorised or accidental access to, alteration, disclosure, or loss and destruction of information.

Implications of the Act

Users are liable for damages if personal data is lost, destroyed or disclosed to unauthorised persons.

Firstly, a means of protecting personal data must be chosen. This protection should be effective against unauthorised disclosure, access, alteration and destruction. Storing floppy and removable hard disks in a safe is the conventional method, but it has severe limitations; much data storage is now on fixed hard disks, most of which cannot be removed from the computer. Transmitted data, such as files sent across a computer network, cannot be protected using a safe.

A versatile solution is to use **encryption**, as described in this guide, in combination with taking suitable backup copies.

Having chosen the method of enforcing security, it is essential to establish procedures which dictate how the security is to be applied, which data must be protected, which staff may have access to the data, and how that access is controlled. These procedures should be adhered to conscientiously at all times: a method of protection is only effective if used correctly.

EU Data Protection Directive

In January 1998, the UK government published the Data Protection Bill. When this is passed it will implement the EU Data Protection Directive into UK law. This must be achieved by 24 October 1998.

The EU Directive's provisions go beyond those of the present UK Data Protection Act. In particular, the Directive:

- Extends individuals' rights to information about the use of their data.
- Introduces new rules for the transfer of personal data outside the EU.
- Provides individuals with a wider entitlement to go to court for any breach of the law.
- Provides wider rights to compensation.

UK Computer Misuse Act 1990

This is *'An Act to make provision for securing computer material against unauthorised access or modification; and for connected purposes.'*

The Act is a wide-sweeping measure passed to facilitate the prosecution of hackers, virus-writers and other computer crime perpetrators, and specifies several computer misuse offences.

Firstly, it states that a person is guilty of an offence if they cause a computer to perform any function with intent to secure access to any program or data held in any computer, if the access they intend to secure is unauthorised and if they know at the time when they cause the computer to perform the function that that is the case. The penalty on summary conviction is imprisonment of up to six months, or a fine of up to level 5 on the standard scale, or both.

Secondly, a person is guilty of an offence if they commit an offence as described above with intent to commit or facilitate commission of further offences. The penalty on summary conviction is imprisonment of up to six months or a fine of up to the statutory maximum, or both, and on conviction or indictment imprisonment of up to five years, or a fine, or both.

Thirdly, the Act specifies that it is an offence to modify computer material without authorisation. Modification specifically includes the impairment of the operation of any computer, prevention or hindering of access to any program or data and the impairment of the operation of any program or the reliability of data. The penalty on summary conviction is imprisonment of up to six months, or a fine of up to the statutory maximum, or both, and on conviction or indictment, imprisonment of up to five years, or a fine, or both.

The Act specifically states that it is immaterial to guilt whether the accused was a British citizen at the time of the offence. The Police also have extradition powers under the Extradition Act 1870 for offences outlined in 'Secondly ...' and 'Thirdly ...' above, any conspiracy to commit such an offence and any attempt to commit an offence outlined in 'Thirdly ...' above.

Section 4 of the Act states that its jurisdiction, apart from covering the United Kingdom, also extends to other countries, provided that a significant link exists with the UK. That means that the Act applies if the suspect is in the UK and the victim is abroad, the suspect is abroad and the victim in the UK or even if both the suspect and the victim are abroad, but the offence was committed through a channel passing through the UK.

The **Computer Crime Unit** of New Scotland Yard in London deals specifically with computer crime in the United Kingdom. The unit is keen that individuals and organisations should report computer crime cases, including computer virus attacks. Contact Computer Crime Unit, New Scotland Yard, 2 Richbell Place, London, WC1N 3LA, UK, Tel 0171 230 1177, Fax 0171 230 1275.

The 'Orange Book'

The 'Orange Book', published by the U.S. Department of Defense ('Trusted Computer Systems Evaluation Criteria (TCSEC)', CSC-STD-001-S3, 1983) is an attempt by the US military to formalise the criteria for the evaluation of the security of a computer system. The book does not mention issues such as encryption or authentication but it provides a useful framework for assessing access control. Each higher classification includes the requirements of all the lower ones. The security scale, from lowest to highest, is:

D **Minimal Protection:** The system has been evaluated, but failed to meet the requirements for a higher evaluation class.

C1 **Discretionary Security Protection:** Discretionary security requirement by separating users and data. Access limitations on an individual basis are implemented.

C2 **Controlled Access Protection:** More finely-grained access control than C1 systems. Auditing of security-relevant events and resource isolation.

B1 **Labelled Security Protection:** Data labelling and mandatory access control over named subjects and objects must be present. Capability for labelling exported information.

B2 **Structured Protection:** A clear formal security policy model is required, with access control enforcement for all subjects and objects in the system. Covert channels are addressed.

B3 **Security Domains:** The system must be tamper-proof and small enough for analysis and testing. There must be a security administrator, more audit mechanisms, system recovery procedures.

A1 **Verified Design:** Functionally equivalent to those in class B3. The design and implementation must be proven mathematically to be correct, using a formal language.

Commercial operating systems currently have ratings between D and B1.

UK security product certification

In a number of European countries including the UK, criteria have been developed for the evaluation of security in information technology products and systems, covering both hardware and software. In the UK, these criteria are put into practice through a national certification scheme run by CESG (the Communications-Electronics Security Group at GCHQ), together with the Department of Trade and Industry.

Under the UK scheme, manufacturers can submit IT security products for evaluation by an approved CLEF (Commercially Licensed Evaluation Facility). Products are assessed against a set of security claims made for them, to one of seven possible 'confidence levels' of assurance. In outline, these are:

UKL0 **Unassured:** There are no requirements for this level; it is reserved for systems which are severely flawed or about which there is insufficient information to form an evaluation opinion.

UKL1 **Vendor-assured:** Security requirements and development/test results must be documented to good industry standards. Configuration control must likewise follow good industry practice. Confidence at this level is based largely upon the vendor's demonstrated competence.

UKL2 **Independently Tested:** Intelligently planned functional and penetration testing by CESG-approved evaluators. Disciplined development of system architecture, good documentation and enforced configuration control.

UKL3 **Independently Assured:** Distinct trusted and untrusted architectural features; trusted components are assessed as well as tested. Careful documentation of interfaces between trusted and untrusted items.

UKL4 **Structurally Sound:** System architecture must be developed in a structured manner. Important security requirements must be precisely defined. Changes to architecture must be strictly controlled.

UKL5 **Rigorous Design:** The architecture must be defined in a rigorous manner and the implementation must be performed in a structured manner. Development process and environment are assessed. Developer's testing is tested.

UKL6 **Assured Design:** Proven fulfilment by architecture of security requirements. Rigorous implementation. Highest levels of documentation and configuration control throughout.

This certification scheme has now been superseded by the European ITSEC scheme.

ITSEC

The Information Technology Security Evaluation Criteria (ITSEC) is an effort to establish a comprehensive set of security requirements for widespread international use. ITSEC is intended to be a superset of TCSEC (described in the Orange Book) with the rating equivalent with the TCSEC evaluation classes.

ITSEC separates *functionality* (what does the target of evaluation claim to do?) from *assurance*. Assurance is further separated into *effectiveness* (is what the target of evaluation claims to do useful?) and *correctness* (does the target of evaluation do what it claims to do?).

The *functionality* of the Target Of Evaluation (TOE) is documented in its security target. The level of information contained in the security target depends on the target evaluation level. The security target generally contains security objectives (why the functionality is wanted), security enforcing functions (what functionality is actually provided) and security mechanisms (how the functionality is provided). The security enforcing functions are usually grouped into eight generic headings: identification and authentication, access control, accountability, audit, object reuse, accuracy, reliability of service and data exchange. A number of standard functionality classes (F-C1, F-C2, F-B1, F-B2 and F-B3) have also been defined to correspond closely to the functionality requirements of the TSEC classes C1 to A1.

The *effectiveness* aspect of the evaluation of assurance concerns whether the TOE provides an acceptable level of security in the context of its actual or proposed operational use. The requirements for effectiveness do not vary with the evaluation level, but the level of rigour does. Effectiveness criteria include four construction aspects (suitability of functionality, binding of functionality, strength of mechanisms and construction vulnerability), and two operational aspects (ease of secure use and operational vulnerability).

The *correctness* aspect of assurance concerns whether the TOE accurately reflects the security target and meets the requirements for construction and operation at the designated assurance level. Depending upon the evaluation level, correctness evidence includes the security target, architectural design, detailed design, implementation, development environment, operational documentation, and operational environment.

Seven evaluation levels are defined in respect of the confidence in the correctness of a TOE. These are described in the *Information Technology Security Evaluation Criteria* (Luxembourg, 1991) as:

E0 This level represents inadequate assurance.

E1 At this level there shall be a security target and an informal description of the architectural design of the TOE. Functional testing shall indicate that the TOE satisfies its security target.

E2 In addition to the requirements for level E1, there shall be an informal description of the detailed design. Evidence of functional testing shall be evaluated. There shall be a configuration control system and an approved distribution procedure.

E3 In addition to the requirements for level E2, the source and/or hardware drawings corresponding to the security mechanisms shall be evaluated. Evidence of testing of those mechanisms shall be evaluated.

E4 In addition to the requirements for level E3, there shall be an underlying formal model of security policy supporting the security target. The security enforcing functions, the architectural design and the detailed design shall be specified in a semiformal style.

E5 In addition to the requirements for level E4, there shall be a close correspondence between the detailed design and the source code and/or hardware drawings.

E6 In addition to the requirements for level E5, the security enforcing functions and the architectural design shall be specified in a formal style, consistent with the specified underlying formal model of security policy.

E2 through E6 correspond roughly to the assurance aspects of C2, B1, B2, B3 and A1 TCSEC ratings.

BS 7799

BS7799 is a British Standard for information security. Its stated objectives are:

- To serve as a single reference point for identifying the range of controls needed for most situations encountered in industry and commerce.
- To enable mutual trust to be established between networked sites and trading partners, and provide a basis for management of facilities between IT users and service providers.

The Standard contains about 100 suggested controls. These are intended for guidance, and the Standard acknowledges that every organisation will have a different set of requirements. The controls are divided into security policy, security organisation, assets classification and control, personnel security, physical and environmental security, computer network management, system access control, systems development and maintenance, business continuity planning, and compliance. There are ten controls considered to be essential, and these are designated as Key Controls:

Formal security policy; this document sets out a company's basic security principles, and should be issued to every employee.

Allocation of IT security responsibilities; explicitly defines and allocates individual security roles and responsibilities within the company.

Staff education and training; users must understand what the security procedures are, why they are necessary, and the implications of a breach of security.

Reporting of security incidents; events which have, or could have, led to a breach of security have to be recorded at a nominated focal point to ensure that incidents do not go unreported.

Virus controls; virus detection and prevention measures should be implemented in every organisation where personal computers are in use.

Continuity plans; these should be drawn up and tested, so that business operations can be maintained in the event of failure of vital services.

Software copyright controls; no copyright material should be copied without the owner's consent, and staff should be briefed on the implications of licence agreements for proprietary software.

Safeguarding of company records; important records should be safeguarded from loss, destruction and falsification, and guidelines for the retention, storage, handling and disposal of corporate information should be issued.

Compliance with data protection legislation; users must be familiar with the UK Data Protection Act, and receive guidance on what constitutes personal data.

Compliance with security policy; the procedures and standards adopted by a company should be followed and regularly reviewed to ensure that they remain adequate.

Sources of information

This appendix lists sources of information on data security, including periodicals, books, conferences, Web sites and newsgroups.

Data security periodicals

Computer Audit Update

Elsevier Science Ltd, The Boulevard, Langford Lane,
Kidlington, Oxford, OX5 1GB, UK.
Tel +44 1865 843000, Fax +44 1865 843010.

Computer Fraud & Security Bulletin

Elsevier Science Ltd. For contact details, see 'Computer Audit Update'.

Communication Law

Tolley Publishing Co Ltd, Tolley House,
2 Addiscombe Road, Croydon, Surrey, CR9 5AF, UK.
Tel +44 181 686 9141, Fax +44 181 686 3155.

Computer Law & Security Report

Elsevier Science Ltd. For contact details, see 'Computer Audit Update'.

Cryptologia

Rose-Hulman Institute of Technology, Wabash Ave
Terre Haute, Indiana 47803, USA.
Tel +1 812 877 1511, Fax +1 812 877 3198.

Datenschutz Berater

Verlagsgruppe Handelsblatt GmbH, Kasernstr. 67,
D-40213 Dusseldorf, Postfach 10 11 02, Germany.
Tel +49 211 887-0, Fax +49 211 887-1400.

Information Security Monitor

Infotech Publishing, PO Box 2619, London W1A 3PT,
UK. Tel +44 171 362 0219, Fax +44 171 691 9422.

International Journal of Forensic Computing

Third Floor, Colonnade House, High Street,
Worthing, West Sussex, BN11 1NZ, UK.
Tel +44 1903 209226, Fax +44 1903 233545.

Journal of Cryptology

Springer-Verlag New York Inc., 175 Fifth Ave,
New York, NY 10010, USA.
Tel +1 212 460 1500, Fax +1 212 473 6272.

SECURE Computing

West Coast Publishing Ltd, William Knox House,
Britannic Way, Llandarcy, Swansea, SA10 6EL, UK.
Tel +44 1792 324000, Fax +44 1792 324001.

Virus Bulletin

Virus Bulletin Ltd, The Pentagon, Abingdon Science
Park, Abingdon, OX14 3YP, UK.
Tel +44 1235 555139, Fax +44 1235 531889.

Recommended books on data security

General titles

The Computer Security Reference Book by K. Jackson, J. Hruska and D. Parker (Butterworth) is an overview of data security by 40 prominent authors in the field.

The Data and Computer Security Dictionary of Standard Concepts and Terms by D. Longley and M. Shain (Macmillan), and the *Butterworth Security Dictionary: terms and concepts* by J.J. Fay (Butterworth) define computer security terms and give short explanations.

Datapro reports on Microcomputer Security is a set of comprehensive reports on security for microcomputers and the products on the market.

Protecting Information on Local Area Networks by J.A. Schwitzer (Butterworth) deals with the security of LANs (Local Area Networks).

QUE'S Guide to Data Recovery by S. Mueller (QUE) and *Hard Disk Secrets* by J.M. Goodman (IDG) discuss the security of information on magnetic disks.

Cryptography

Cipher Systems - the Protection of Communications by H. Beker and F. Piper (Van Nostrand Reinhold) gives an excellent account of DES and RSA, as well as other cryptographic topics.

Security for Computer Networks by D.W. Davies and W.L. Price (John Wiley & Sons) is an overview of encryption methods and their use in practice.

Cryptograms and Spygrams by Norma Gleason (Dover) is a light account of elementary cryptography.

Cryptography by C. Myer and S. Matyas (John Wiley & Sons) is an authoritative text on computer security and contains DES test patterns.

Secure Information Transfer by K. Jackson (Butterworth) contains a detailed description of encryption, key management and their use in practice.

Security Mechanisms for Computer Networks by S. Muftic (Ellis Horwood) is a discussion of up-to-date topics on the subject.

Seminumerical Algorithms, Vol 2, by D. Knuth (Addison Wesley) is a good place for the mathematically-minded to discover the charms of the large number arithmetic used in RSA and similar algorithms.

Computer viruses

Computer Viruses - A High-Tech Disease by R. Burger (Abacus) contains listings of a number of viruses for PCs and mainframes.

Computer Viruses and Anti-Virus Warfare by J. Hruska (Ellis Horwood) is a practical guide on how to avoid and combat a virus infection.

Rogue Programs: Viruses, Worms and Trojan Horses by L.J. Hoffman (Van Nostrand) is a collection of essays by well-known authors on the problem of viruses.

A Pathology of Computer Viruses by D. Ferbrache (Springer-Verlag) is a very good overview of the subject, including its history.

A Short Course on Computer Viruses by F. Cohen (ASP Press) is an excellent account of virus writing and anti-virus defences.

Exhibitions and conferences

COMPSEC London, UK (November).

Elsevier Seminars, The Boulevard, Langford Lane,
Kidlington, Oxford, OX5 1GB, UK,
Tel +44 1865 843000, Fax +44 1865 843010.

Crypto Santa Barbara, California, USA (August).

Burt Kaliski, RSA Data Security Inc, 10 Twin Dolphin
Drive, Redwood City, CA 94065, USA,
Tel +1 415 595 8782.

Infosec Paris (May).

MCI, 19 Rue d'Athènes, 75009 Paris, France,
Tel +33 44 59 72 20.

Infosecurity London (April/May).

Reed Exhibition Companies Ltd, 26 The Quadrant,
Richmond, Surrey, TW9 1DL, UK,
Tel +44 181 910 7910.

Securicom Paris, France (March).

Securicom, 8 rue de la Michodiere, 75002 Paris,
France, Tel +33 1 47 42 21 00.

Eurocrypt Held in different locations (April).

Virus Bulletin Conference Munich, Germany (October 1998).

Virus Bulletin, The Pentagon, Abingdon Science Park,
Abingdon, UK, OX14 3YP, Tel +44 1235 555139,
Fax +44 1235 531889.

On-line information

World Wide Web

Virus Bulletin

The Virus Bulletin Web site located at http://www.virusbtn.com/

- Is an official WWW archive site for Joe Wells' WildList (the most widely regarded guide to which viruses are at large in the user community).
- Allows access to Project VGrep (the virus name cross-referencing system).
- Includes links to anti-virus information and vendor sites.

Sophos

The Sophos Web site is located at http://www.sophos.com/. It is updated regularly with the latest developments at Sophos and news from the virus world. Sections currently include:

- Product information: Full details of Sophos anti-virus software, data security software, workshops and training, and a facility for submitting enquiries directly to Sophos.
- Product downloads: Evaluation versions of Sophos anti-virus software, limited access to Sophos beta software, unrestricted access to all the latest Sophos documentation, and (for Sophos customers only) full versions of the latest Sophos software.
- Technical support: Frequently asked questions about Sophos products and about dealing with viruses, a facility for submitting queries to Sophos technical support, and the latest virus identities.
- Virus information: Analyses of the most common and interesting viruses, along with details of hoaxes, scares and misunderstandings, regular features, and all Sophos technical reports.
- About Sophos: Information on Sophos, the international distributors, exhibition and trade fair dates, press releases and employment opportunities.

USENET news

alt.comp.virus

An unmoderated ‘free-for-all’ newsgroup. Contributors include novices, aspiring experts, and anti-virus professionals, and the topics covered include: the removal of specific viruses, discussion of anti-virus issues, or debate about the purpose of the group.

Virus-L/comp.virus

A moderated mailing list and its USENET newsgroup equivalent.

Glossary

Access Control: The process of ensuring that systems are only accessed by those authorised to do so, and only in a manner for which they have been authorised.

Active Attack: An attack on a system which either injects false information into the system, or corrupts information already present on the system. See also Passive Attack.

Algorithm: An algorithm is a set of rules which specifies a method of carrying out a task (e.g. an encryption algorithm).

ANSI: American National Standards Institute; the organisation which issues standards in the US.

ASCII: American Standard Code for Information Interchange; the standard system for representing letters and symbols. Each letter or symbol is assigned a unique number between 0 and 127.

Asymmetric Encryption: Encryption which permits the key used for encryption to be different from the key used for decryption. RSA is the most widely used asymmetric encryption algorithm.

Audit Log: See Audit Trail.

Audit Trail: Audit trails provide a date and time stamped record of the usage of a system. They record what a computer was used for, allowing a security manager to monitor the actions of every user, and can help in establishing an alleged fraud or security violation.

Authentication: The process of assuring that data has come from its claimed source, or of corroborating the claimed identity of a communicating party.

Authorisation: Determining whether a subject is trusted for a given purpose.

Availability: The prevention of unauthorised withholding of information or resources.

Back Door: An undocumented means of bypassing the normal access control procedures of a computer system.

Background Operation: The name applied to a program running in a multi-tasking environment over which the user has no direct control.

Backup: A copy of computer data that is used to recreate data that has been lost, mislaid, corrupted or erased.

Bad Sectors: During formatting of MS-DOS disks, all sectors are checked for usability. Unusable sectors are labelled as 'bad' and are not used by DOS. The remaining areas can then still be used. Viruses sometimes label good sectors as bad to store their code outside the reach of the users and the operating system.

BAT: The extension given to 'batch' file names in MS-DOS. A batch file contains a series of MS-DOS commands, which can be executed by using the name of the file as a command. AUTOEXEC.BAT is a special batch file which is executed whenever a PC is switched on, and can be used to configure the PC to a user's requirements.

BBS: Bulletin Board System; a computer with one or more modems attached which can be used remotely via the telephone system. Most bulletin boards act as repositories for downloadable software.

Bell-LaPadula Model: An access security model couched in terms of subjects and objects. Information shall not flow to a lesser or non-comparable classification.

Biba Model: An integrity model in which there can be no contamination by a less trusted or non-comparable subject or object.

Binary: A number system with base 2. The binary digits (bits) are 0 and 1. Binary arithmetic is used by today's computers because the two digits can be represented with two electrical or magnetic states, for example the presence and absence of a current.

Biometrics: A technique for identifying a person by one of their personal characteristics e.g. retina pattern, fingerprint etc.

BIOS: The Basic Input/Output System of MS-DOS which constitutes the lowest level of software which interfaces directly with the hardware of the microcomputer.

Bit: The smallest unit of information. It can only have the value 0 or 1. The word 'bit' is derived from the initial and final letters of the phrase 'Binary Digit'.

Bit Copying: A technique for making a copy of a disk by reading all of the individual bits on each track of the disk, and making a direct copy of each track onto a new disk. A bit copying program has no knowledge of the file structure being used on a disk.

Block Cipher: A cipher which provides encryption and decryption by operating on a specified size of data block, e.g. 64 bits.

Boot Protection: Method used to prevent bypassing security measures installed on a hard disk by booting a microcomputer from a floppy disk.

Boot Sector Virus: A type of computer virus which subverts the initial stages of the booting-up process. A boot sector virus attacks either the master boot sector or the DOS boot sector.

Booting: A process carried out when a computer is first switched on or reset, where the operating system software is loaded from disk.

Boot Sector: Part of the operating system which is first read into memory from disk when a PC is switched on (booted). The program stored in the boot sector is then executed, which in turn loads the rest of the operating system into memory from the system files on disk.

Bug: A small electronic device used for covert eavesdropping. Different types are available to listen to voice conversations, data being transmitted across a network, or telephone lines. A fault in a computer program is also called a bug. The two meanings are entirely separate.

Byte: A set of 8 bits which is the amount of information sufficient to store one character. It is usually the smallest individual unit that can be read from or written to memory.

Cache: High-speed data storage used to hold data retrieved from a slow device. Using a cache increases the overall performance of a system.

CBC: Cipher Block Chaining; a mode of use of a block cipher.

CCC: Chaos Computer Club; an infamous group of German hackers based in Hamburg.

CCTA: Central Computer and Telecommunications Agency; the UK Government agency responsible for computer purchases (amongst other duties).

CESG: Communications-Electronics Security Group; a UK government COMPUSEC agency (CCTA is another).

CFB: Cipher Feedback; a mode of use of a block cipher.

CGI: Common Gateway Interface; a protocol used to communicate information from Web pages to programs stored and run on a Web server.

Checksum: A value calculated from item(s) of data which can be used by a recipient of the data to verify that the received data has not been altered. Usually 32 or 64 bits long.

Cipher: Encryption algorithm.

Ciphertext: A term used to describe text (or data) that has previously been encrypted. See also Encryption.

CMOS: Complementary Metal-Oxide Semiconductor; a technology used to manufacture chips which have very low power consumption. CMOS chips are used in battery-backed applications such as the time-of-day clock and for the non-volatile storage of parameters.

COM: The extension given to a type of executable file in MS-DOS. A COM file is similar to an EXE file, but can only contain up to 64K of code and data. In operating systems other than DOS, the extension COM can have a different significance.

Companion Virus: A virus which ‘infects’ EXE files by creating a COM file with the same name which contains the virus code. It exploits the DOS property that if two programs with the same name exist, the operating system will execute a COM file in preference to an EXE file.

Compiler: A computer program which translates programs written in a high-level language that can be readily understood by humans, into low-level instructions that can be executed by a computer's CPU.

Compressed File: See File Compression.

COMPSEC, COMPUSEC: Often used abbreviations for COMPuter SECurity.

Computer Crime: This phrase has two meanings: any crime mediated by a computer; or any crime that attacks a computer system as part of the process of committing the crime. The meaning used in any particular situation is context dependent, and not always clear.

Confidentiality: The process of ensuring that data is not disclosed to those not authorised to see it. Also known as secrecy.

Conventional Memory: The bytes of PC memory addressable by the 8086 instruction set.

Co-processor: Specialised computer hardware used in conjunction with a CPU to perform a specific task very efficiently e.g. floating point arithmetic, matrix multiplication.

Copy Protection: A method which makes it difficult (if not impossible) to make copies of a computer program. Copy protection tries to prevent software theft.

CPU: Central Processing Unit; the heart of every PC, the device which takes instructions from memory and executes them. In most PCs, the CPU is a single microprocessor.

CRC: Cyclic Redundancy Check; a mathematical method for verifying the integrity of data. It is a form of checksum, based on the theory of maximum length polynomials. While more secure than a simple checksum, CRCs do not offer true cryptographic security. See Cryptographic Checksum.

Cryptanalysis: The study of an encryption system, often with the intention of detecting any weakness in the encryption algorithm.

Cryptographic Checksum: A checksum calculated by using a cryptographically based algorithm. It is impossible to 'engineer' changes to data in such a way as to leave a cryptographic checksum unchanged.

Data Protection: A group of techniques used to preserve three desirable aspects of data: Confidentiality, Integrity and Availability. Also a legal term with specific meaning (somewhat different to the above definition).

Deciphering: The same as decrypting; see Decryption.

Decryption: Decryption is the process of transforming ciphertext back into plaintext. It is the reverse of encryption.

Decryption Key: See Key.

DES: Data Encryption Standard; an algorithm for encrypting or decrypting 64 bits of data using a 56-bit key. DES is widely used in the financial world.

Device Driver: A program used to ‘handle’ a hardware device such as a screen, disk, keyboard etc. This allows the operating system to use the device without knowing specifically how the device performs a particular task.

Digital Signature: A means of protecting a message from denial of origination by the sender, usually involving the use of asymmetric encryption to produce an encrypted message or a cryptographic checkfunction.

Diskless Node: A terminal, or a PC, on a network which does not possess internal floppy or hard disk drive(s).

Diskless Workstation: A PC which does not contain a floppy disk drive and is connected to a network.

DNS: Domain Name System; the distributed database used to translate human-readable Internet addresses (e.g. ‘sophos.com’) into numeric IP addresses (e.g. 193.82.145.1). IP addresses are difficult to remember and change if a machine moves, unlike DNS names. Domain names are hierarchical; for example ‘elbereth.sophos.com’ is the machine ‘elbereth’ in the network ‘sophos’, which belongs to the ‘com’ top-level domain for commercial entities.

Dongle: A hardware security product which must be plugged into a computer system before a particular application program will execute. A dongle aims to prevent illegal copying of a computer program.

DOS: Disk Operating System. See MS-DOS and PC-DOS.

DOS Boot Sector: The boot sector which loads the BIOS and DOS into PC RAM and starts their execution. Common point of attack by boot sector viruses.

Downloading: A process where data is transferred electronically from a 'host' computer to an intelligent terminal or PC.

Dropper: An EXE or a COM file which infects a PC with a virus, but which itself does not replicate. Commonly used for spreading boot sector viruses via bulletin boards.

EAROM: Electrically Alterable Read Only Memory; a particular type of EEPROM, in which individual bytes can be altered by electrical pulses.

ECB: Electronic Code Book; a mode of use of a block cipher.

EEPROM: Electrically Erasable Programmable Read Only Memory; a non-volatile memory which can be written to and read from many times. It is erased by an electrical pulse. EEPROMs are used for storing data which does not change frequently, e.g. setup parameters.

Electronic Mail: Messages exchanged over a computer communications network.

Email: See Electronic Mail.

Enciphering: The same as encrypting; see Encryption.

Encryption: A process of disguising information so that it cannot be understood by an unauthorised person.

Encryption Key: See Key.

EPROM: Electrically Programmable Read Only Memory, a non-volatile memory which can be programmed (written to) once, and read from many times. Most types of EPROM can be erased by exposure to ultra-violet light. EPROMs are used for storing data which is unlikely to be changed.

EXE: The extension given to executable files in MS-DOS. These are similar to COM files, but can contain more than 64K of code and data.

Exhaustive Key Search: Finding out which key was actually used by an encryption system by testing all possible keys in turn.

Expanded Memory: PC memory which conforms to the industry standard specification EMS (Expanded Memory Specification), and enables the CPU to access more than 640K of memory.

Extended DOS Partition: An area of the hard disk assigned to DOS. It is usually subdivided into logical disks. The first logical disk can be made bootable though this is not usual.

Extended Memory: Memory in PCs which lies above 1 Mb in a 80286 (or above) machine.

False Negative: An existent event reported as non-existent, e.g. the absence of a virus when the virus is present.

False Positive: A non-existent event reported as existent, e.g. the presence of a virus when no virus is present.

FAT: File Allocation Table; a mnemonic term used by the MS-DOS operating system (and others) to describe the part of a disk which contains information describing the physical location on the disk of the chains of clusters forming the files stored on that disk.

File Compression: The compacting of a file through the process of recoding its bit structure into a shorter form. File compression must be reversible.

File Encryption: The transformation of a file's contents (in plaintext) into an unintelligible form by means of some form of cryptographic system or manipulation.

File Integrity: Techniques used to provide 'safe' backup files for recovery purposes in the event that critical files have become contaminated through some accidental or intentional mechanism (e.g. computer virus attack).

File Labelling: The classifying of the sensitivity level of a file either by external (visible outside marking) or internal (magnetic coding of the header label) coding, or by a combination of these two methods.

File Server: A central data repository for a computer network, which may provide other centralised services such as shared printer control.

Firewall: A firewall is the Internet equivalent of a front door and security guard. It sits between the Internet and an organisation's network, and only passes authorised network traffic.

Firmware: Jargon for a computer program stored in a non-volatile memory such as an EPROM or an EEPROM.

Floppy Disk: Interchangeable magnetic disk used to store computer data.

FTP: File Transfer Protocol; a system built on top of TCP whereby Internet users can connect to remote sites and upload or download files.

Hacker: An individual whose interests, motivated for benign or malicious reasons, concern 'breaking into' computer systems. The word hacker is also used to denote someone who produces prodigious amounts of software. The two meanings are completely distinct, and often confused.

Hard Disk: A hermetically sealed magnetic disk, generally fixed within a computer, which is used to store data.

Hardware: Any component of a computer system that has physical form. It is a term used to draw a distinction between the computer itself (hardware), and the programs which are executed on the computer (software).

Hash Function: A function which maps a set of variable size data into objects of a single size. Widely used for fast searching.

Hashing: The process of calculating a hash function.

Hexadecimal: A system of counting using number base 16. The numbers 10 to 15 are represented by the characters 'A' through 'F' respectively. Hexadecimal is often abbreviated to Hex. Each Hex digit is equivalent to four bits (half a byte) of information.

HPFS: High Performance File System; a file system used by OS/2 and supported by Windows NT.

HTML: Hyper-Text Markup Language; the format for most documents on the World Wide Web. Web browsers interpret HTML and display the document as formatted text with included graphics etc. Areas of the document called *anchors* contain pointers (called URLs) to other documents on the Web, allowing users to jump quickly to related information.

HTTP: Hyper-Text Transport Protocol; built on top of TCP, this protocol is used by Web servers to serve documents to Web browsers. A simple protocol; a new connection is established for every command. The most common is GET to retrieve a document.

IC: Integrated Circuit; an electronic device containing many discrete electronic components such as transistors, resistors and the wire links which interconnect them. ICs are usually made in very large numbers and in miniaturised form, on a common base or substrate of silicon.

ID: An identification code, username, identification card or an identification token.

IDE: The extension given to a file containing a virus identity encoded with Sophos' Virus Description Language (VDL). It will appear as a string of ASCII characters.

Integrity: A security protection aimed at ensuring that data cannot be deleted, modified, duplicated or forged without detection.

InterCheck: Proprietary Sophos technology which enables a server-based virus scanner to be used for scanning workstations connected to the network.

Internet: An internetwork or internet is a network consisting of many connected networks. *The* Internet is by far the largest of these, and the term is rarely used in the original, more general sense. The Internet consists of several million computers and a huge number of organisations, and is worldwide in scope. The Internet (or Net) uses the TCP/IP protocols exclusively.

Interrupt: A mechanism by which a process can attract the immediate attention of the CPU, usually in order to serve an urgent request from an external device. The interrupt table on 8086 microprocessors occupies the bottom 1Kb of RAM.

I/O Port: A computer communicates with the outside world through Input/Output (I/O) ports. Examples are the RS-232 serial port and printer ports on a PC.

IP: Internet Protocol; the base level of the TCP/IP system. It is a connectionless, unreliable datagram service. 'Datagram' means that all communications are made up of packets; 'connectionless', that each network packet is separate and individually routed; and 'unreliable' means that packets are not guaranteed to get through. An IP packet contains two IP addresses for its source and destination.

IP Address: A numeric Internet address; a 32-bit binary number, normally written in dotted-decimal notation; e.g. '194.82.145.1'.

ISO: International Organisation for Standardisation; the worldwide federation of international standards bodies.

IV: Initialisation Variable; a value used to initialise modes of use of certain block ciphers.

K: Shorthand for a thousand (1,000), but in computing it is often used to mean 1,024 (2^{10}, approximately 1,000). For example, 64 Kb or 64 Kbytes refers to 64*1,024 (= 65,536) bytes.

KDC: Key Distribution Centre; a central system for the generation and distribution of encipherment keys. A KDC ensures that in cases where a large number of systems need to talk to each other, they do not need to share a unique key between each pair of systems.

KEK: Key Encrypting Key; a key which is used exclusively for the encryption of other encryption keys.

Key: When used in the context of encryption, a series of numbers which are used by an encryption algorithm to transform plaintext data into encrypted (ciphertext) data, and vice versa. Confusingly, key can also refer to a physical token which gives access to a system.

Key Gun: A device for the secure transport of encipherment keys from the point of generation to the point of application.

Key Management: The process of securely generating, transporting, storing and destroying encryption keys.

Key Space: All the possible keys that can be used by an encryption algorithm.

Keystream Generator: A system that generates keys continuously.

Keystream Sequence: The output of a keystream generator.

KTC: Key Translation Centre; a central system which can receive an enciphered encipherment key, decipher it and re-encipher it under a new key.

Laptop: A portable microcomputer small enough to be used 'on the lap'.

LAN: Local Area Network; a data communications network covering a limited area (up to several kilometres in radius) with moderate to high data transmission speeds.

LCD: Liquid Crystal Display; a type of display used mainly in portable PCs because of its low power consumption.

Letter Bomb: A logic bomb contained in electronic mail, which will trigger when the mail is read.

Link Virus: A virus which subverts directory entries to point to the virus code.

Logic Bomb: A program modification which causes damage when triggered by some condition such as the date, or the presence or absence of data.

M: Shorthand for a million (1,000,000), but in computing it is often used to mean 1,048,576 (2^{20}, approximately one million). For example, 1 Mb or 1 Mbyte refers to 1,048,576 bytes.

MAC: Message Authentication Code; a cryptographically calculated checksum. Particularly used to protect against spoofing and replays. Unlike a digital signature, a MAC requires knowledge of a secret key for verification.

Macro: Many software products allow their data files to contain instructions to carry out program commands automatically. These instructions are often called macros, and they generally allow access to a substantial range of functions such as the opening, manipulating and closing of files.

Macro Virus: A virus which uses macros in a data file to become active in memory and attach itself to other data files. Unlike conventional viruses, macro viruses can be written relatively easily with little specialist knowledge, and can also attain a degree of platform independence.

Mainframe: Large computer systems, often occupying purpose-built facilities, used for IT applications requiring extremely fast processing speeds or large quantities of data. Typical processing speeds are of the order of 100 MIPS.

Mandatory Security: Mandatory security products have effect without an explicit request for action being made. Discretionary security is the opposite of mandatory security, as it must be specifically requested.

Mapped Directory Path: A network drive known by its locally mapped name, e.g. the UNC directory path \\MAIN\USERS\ might be mapped to F:\ on one particular computer on the network.

Master Boot Sector: The first physical sector on the hard disk (sector 1, head 0, track 0) which is loaded and executed when the PC is booted. It contains the partition table as well as the code to load and execute the boot sector of the 'active' partition. Common point of attack by boot sector viruses.

Media-less Microcomputer: See Diskless Node, and Diskless Workstation.

Meet-in-the-middle Attack: A cryptographic attack based on an optimised exhaustive search.

Memory-resident Virus: A virus which stays in memory after it has been executed and infects other objects when certain conditions are fulfilled. Non-memory-resident viruses are active only while an infected application is running.

Menu-driven: Software which presents the user with a fixed 'menu' of command choices, often requiring only a single key or mouse button depression to select the required option.

Message Authentication: The process of calculating and then subsequently verifying a message authentication code.

Message Digest: Same as hash function.

Microprocessor: An integrated circuit which condenses the essential elements of a computer's CPU into a single device.

Minicomputer: A fixed, generally multi-user, computer designed for use as a communal information processing system.

MIPS: Millions of Instructions Per Second.

Mirroring: A technique where data is written to two (or more) disks simultaneously, with the intention of enabling data retrieval even when one of the disks fails.

Modem: A MOdulator/DEModulator is a device which translates digital computer data into a form suitable for transmission over an analogue telecommunications path such as a telephone line, radio channel or satellite link.

Mode of Operation: A set of rules which defines a particular way in which an encryption algorithm should be used.

Mouse: A data input device which, when moved by hand on the surface of a desk, conveys the direction and amount of movement to a computer. A mouse is commonly equipped with one, two or three press-buttons to actuate commands on the computer.

MS-DOS: The Disk Operating System sold by Microsoft. It is the most common microcomputer operating system in the world, and operates on the IBM PC. See also PC-DOS.

Multi-level Security: A system which simultaneously processes information of different security levels, for users who have different levels of security clearance. The system is relied upon ('trusted') to uphold the security policy, and to control the flow of information correctly.

Multipartite Virus: A virus which infects both boot sectors and executable files, thus exhibiting the characteristics of both boot sector viruses and parasitic viruses.

Multi-tasking: The ability of a computer to divide its processing time amongst several different tasks. Although most computers contain only one CPU, they can switch between operations so quickly that several processes appear to run simultaneously.

Nibble: A set of 4 bits.

NCSC: National Computer Security Center; the US Government body responsible for encouraging the development of trusted computer systems suitable for use by the US Government in the processing of classified information.

Need-to-know Principle: The dissemination of information only to those people who actually need the information to carry out a particular authorised task.

NLM: NetWare Loadable Module; a program which runs as a process on a Novell NetWare file server.

Non-interference Model: A type of security model based on the concept that subjects in different security domains should not interfere with each other's operation in any way which violates the security policy of the system.

Non-repudiation: An authentication that with high assurance can be asserted to be genuine, and not subsequently refuted.

Non-volatile Memory: Integrated circuits which retain their content when their normal power source is switched off. The main types are ROM, EPROM, EEPROM and battery-backed CMOS RAM.

Notebook: A colloquial term for a computer (usually a PC) which is even smaller than a laptop computer.

NTFS: NT File System; the Windows NT file system.

OFB: Output Feedback; a mode of use of a block cipher.

Off-site Backup: A backup stored at a geographically remote location.

One-way Function: A function that can readily be calculated, but whose inverse is very difficult to calculate. The knowledge of the result does not allow anybody to determine the parameters used to produce that result.

Operating System: The computer program which performs basic housekeeping functions such as maintaining lists of files, running programs etc. PC operating systems include MS-DOS and OS/2, while minicomputer and mainframe operating systems include UNIX, VMS and MVS.

Optical Disk: A storage device using a laser to record and read data from a rotating disk.

OS/2: An operating system for 80286+ based IBM compatibles. It allows true multi-tasking.

OSI: Open Systems Interconnection; a set of standards defining the protocols for communication between open (non-proprietary) systems.

OVL: The extension commonly given to overlay files in MS-DOS. Overlay files are used with large programs which cannot fit into RAM: parts of the program are loaded as and when needed. Overlay files can have any extension, not just OVL.

Page of Memory: A page of memory is the basic unit of memory used by Windows and is 4K in size.

Parasitic Virus: A computer virus which attaches itself to another computer program, and is activated when that program is executed. A parasitic virus can attach itself to either the beginning or the end of a program, or it can overwrite part of the program.

Partition Table: A 64-bit table found inside the master boot sector on hard disks which contains information about the starting and ending of up to four partitions on the hard disk. The partition table also contains information on the type of the partition, e.g. DOS partition, UNIX partition etc.

Passive Attack: An attack on a system which extracts information and makes use of it, but never injects false information or corrupts any information (which would be an active attack).

Password: Sequences of characters which allow users access to a system. Although they are supposed to be unique, experience has shown that most people's choices are highly insecure. People tend to choose short words such as names, which are easy to guess.

PC: Personal Computer; a desktop or portable single-user computer usually comprising a CPU, memory, screen, keyboard, and disk drive(s). PC has become synonymous with IBM compatible computer, even though this definition is not strictly correct.

PC-DOS: Microcomputer operating system originally used by IBM for its PCs. It is functionally identical to MS-DOS.

Peripheral: External device connected to a computer. Examples include printers, plotters, disk drives, external modems, and a mouse.

Peripheral Access Control: Technique to restrict the use of certain computer peripherals to authorised users.

Pest Program: A collective term for programs with deleterious and generally unanticipated side effects, e.g. Trojan horses, logic bombs, viruses, and malicious worms.

Plaintext: Data before it has been enciphered. The opposite of ciphertext.

Polyalphabetic Cipher: An encryption algorithm in which every letter in the plaintext is substituted with a different letter in the ciphertext.

Polymorphic Virus: Self-modifying encrypting virus.

Port Access Control: Restricting the use of computer data ports to authorised users only.

Positive Erasure: See Secure Erasure.

Primary DOS Partition: A portion of the hard disk assigned exclusively to DOS. It is usually the bootable partition for DOS.

Processor: A unit of hardware that is capable of executing instructions contained in a computer program.

Program: A precise sequence of instructions that specifies what action a computer should perform. 'Software' is often used to describe a computer program.

Proprietary Encryption Algorithm: An encryption algorithm designed to a proprietary (and usually secret) specification.

PS/2: A series of computers from IBM designed to replace the PC/XT/AT range. All models, except model 30, support the 'microchannel architecture'. Cards designed for the IBM PC/XT/AT are not compatible with PS/2 machines.

Public Domain: Two totally distinct meanings exist: the area which is outside government security arrangements; or something which is neither subject to copyright nor a trademark.

RAM: Random Access Memory; volatile memory which can be written to, and read from, at high speed. It is normal to load programs from disk into RAM, and then to execute them. The operating system takes care of the allocation of RAM to executing programs.

Reverse-engineering: The process of deducing how something works without having access to the design details.

ROM: Read Only Memory; a form of non-volatile memory in a computer. Data is embedded into a ROM during manufacture. A ROM is usually used to store the startup software which is executed by a PC on power-up (see Booting).

RS-232: The most widely used standard for serial data communication. The speed of communication is measured in bits per second (baud).

RSA: An asymmetric encryption algorithm, invented by Rivest, Shamir and Adleman in 1976. Unlike DES, the RSA algorithm can be implemented using any number of bits.

Scrambling: Encryption.

Secret Key: Encryption key that must not be disclosed. If it is revealed, the security offered by the encryption algorithm is compromised. Not all encryption keys have to be kept secret, e.g. public keys in asymmetric encryption.

Secure Erasure: Erasure of files from magnetic media performed in such a way that they cannot be recovered. Sophos define three standards: 'Quick', 'Government' and 'Military' secure erasure.

Security: Protection against unwanted behaviour. The most widely used definition of (computer) security is *security = confidentiality + integrity + availability*.

Security Policy: A security policy is the set of rules, principles and practices that determine how security is implemented in an organisation.

Security Server: A special LAN station which runs software that monitors LAN usage, and controls access independently of the LAN operating system.

Sensitive Information: A general term used to describe the information in a system which must be protected by the system security. For example, the information might be personal, medical, financial, commercially sensitive; or in a Government environment, classified.

Server: See File Server and Security Server.

Smart Disk: A device in the shape of a 3.5" floppy disk which contains a microprocessor and memory. It can be read from and written to in a standard floppy disk drive.

SMTP: Simple Mail Transport Protocol; the delivery system for Internet email.

Software: See Program.

Spoofing: Pretending to be someone or something else (e.g. entering someone else's password).

Stealth Virus: A virus which hides its presence from the PC user and anti-virus programs, usually by trapping interrupt services.

Stream Cipher: A cipher which provides encryption and decryption by operating on a continuous stream of data, without imposing limits on the length of the data.

Symmetric Algorithm: An algorithm in which the key used for encryption is identical to the key used for decryption. DES is the best known symmetric encryption algorithm.

SYS: The extension given to system file names in MS-DOS. An example is the file CONFIG.SYS which sets up various configuration parameters for the operating system on power-up.

TCP: Transmission Control Protocol; a reliable, connection-oriented, stream-type service built on top of IP. IP is too 'raw' a protocol for applications to use, so other protocols like TCP sit above it providing additional functionality. TCP provides a two-way data stream, and port numbers in order to differentiate between different TCP connections. Well-known services exist on standard port numbers; SMTP email is on port 25, and HTTP is port 80. Application protocols such as Telnet, FTP and HTTP are built on top of TCP.

TCP/IP: Transmission Control Protocol/Internet Protocol; the collective name for the standard Internet protocols, even though they contain more than TCP and IP.

Telnet: A protocol built on top of TCP providing a remote login and terminal service.

Terminal: A device which consists of a VDU and keyboard. It allows a user to interact with a computer.

Time Bomb: A logic bomb set to trigger at a particular time.

Timeout: A logical access control feature which automatically logs-off users of terminals which do not exhibit signs of activity for a certain duration of time.

Token: A physical object, sometimes containing sophisticated electronics, which is required to gain access to a system. Some tokens contain a microprocessor, and are called intelligent tokens, or smart cards.

Trapdoor: A hidden flaw in a system mechanism that can be exploited to circumvent the system's security.

Trojan Horse: A computer program whose execution would result in undesired side-effects, generally unanticipated by the user. The Trojan horse program may otherwise give the appearance of providing normal functionality.

TSR: Terminate and Stay Resident; a term used to describe an MS-DOS program which remains in memory after being executed. A TSR can be reactivated either by a specific sequence of keystrokes, or at some specific time, or by some specific signal from an I/O port.

UDP: User Datagram Protocol; essentially IP with port numbers and checksums for data integrity. Used for simple protocols where the overhead of TCP is too great.

UMB: Upper Memory Block. DOS=UMB statement in the CONFIG.SYS file specifies that DOS should maintain a link between conventional memory and the upper memory area. This must be specified if programs or device drivers are loaded there.

UNC: Universal Naming Convention; a standard system for naming network drives, e.g. the UNC directory \\MAIN\USERS\ would refer to the USERS directory on the server called MAIN.

UNIX: UNIX is a multi-user operating system, developed by AT&T. Several versions of UNIX exist, which do not all achieve compatibility with each other.

Uploading: The process of transferring data from a remote computer to a central host.

UPS: Uninterruptible Power Supply; a device which detects mains failure and provides power from an internal battery supply for a limited period.

URL: Universal Resource Locator; a World Wide Web ‘address’.

VDL: Virus Description Language; a proprietary Sophos language used to describe virus characteristics algorithmically. It has extensive facilities to cope with polymorphic viruses.

VDU: Visual Display Unit; a computer peripheral which displays text and/or graphics on a television screen.

Virus: Sometimes explicitly referred to as a computer virus, a program which makes copies of itself in such a way as to ‘infect’ parts of the operating system and/or application programs. See Boot Sector Virus and Parasitic Virus.

Virus Identity: An algorithm describing various characteristics of a virus and used for virus recognition. Sophos describe viruses using the proprietary Virus Description Language (VDL).

Virus Mutation: A version of a polymorphic virus.

Virus Pattern: A sequence of bytes extracted from a virus and used for virus recognition.

Virus Signature: An identifier recognised by the virus as meaning ‘this item is already infected, do not reinfect’. It can take different forms such as the text ‘sURIV’ at the beginning of the file, the size of the file divisible by a number or the number of seconds in the date stamp set to 62. Some viruses do not recognise their signatures correctly.

WAN: Wide Area Network; a set of computers that communicate with each other over long distances.

Web: See World Wide Web.

Web Browser: The client side of the World Wide Web; the program used by individuals to access information on the Web.

Web Server: A computer connected to the Internet that serves Web documents, generally using HTTP.

Windows NT: A multi-tasking operating system from Microsoft. NT stands for 'New Technology'.

Workstation: An ill-defined term used to describe a powerful single-user, high-performance, minicomputer or microcomputer, which is used by individuals for tasks involving intensive processing, perhaps CAD or simulation.

World Wide Web: A distributed hyper-text system for the reading of documents across the Internet. World Wide Web documents are generally written in HTML and served by Web servers using the HTTP protocol.

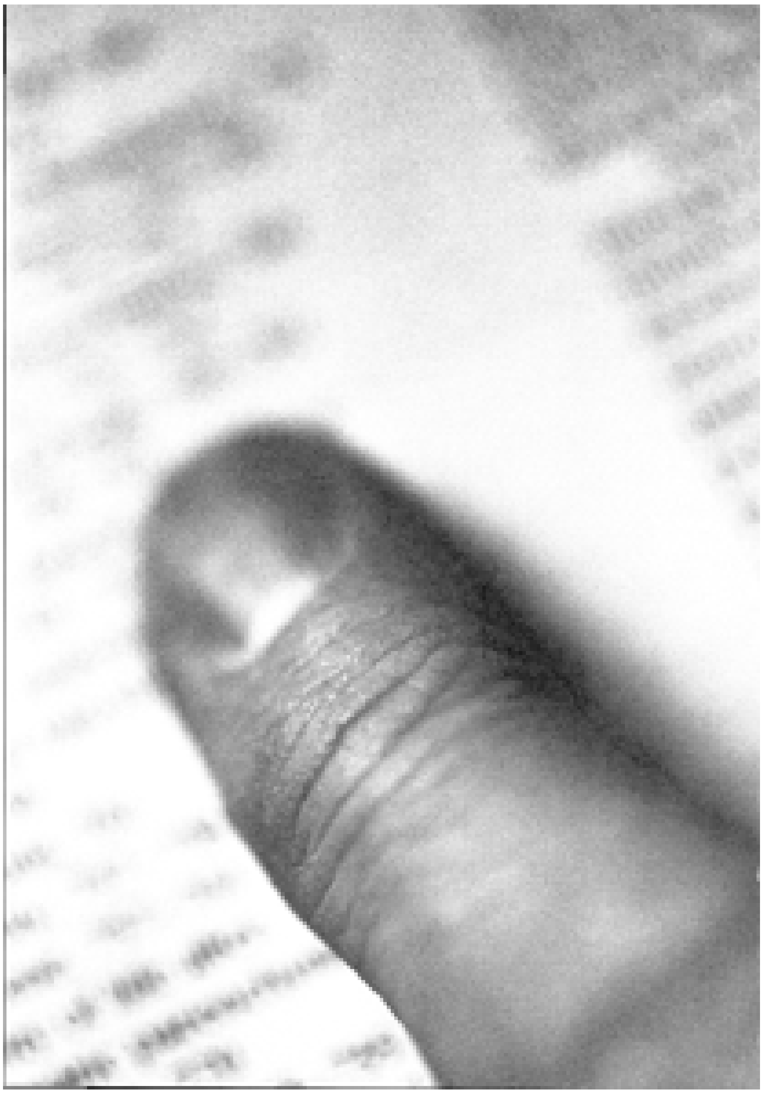
Worm: A program that distributes multiple copies of itself within a system or across a distributed system.

Worm Attack: Interference by a program that is acting beyond normally expected behaviour, perhaps exploiting security vulnerabilities or causing denials of service. See Worm.

WWW: See World Wide Web.

XOR: An abbreviation of the logical operation known as Exclusive-or. An exclusive-or function is defined as having the value true when either of the input conditions (but not both) is true.

X Window System: Networkable graphics and windowing system developed at MIT and commonly used on UNIX systems. It has the property that a program running on one machine can display graphics on another machine if there is a network connection between them.



Index

Australia:

Doctor Disk
Level 7
418A Elizabeth Street
Surry Hills NSW 2010
Australia
Email sales@drdisk.com.au
http://www.drdisk.com.au/
Tel 02 9281 2099 · Fax 02 9281 9740 · Code +61

Bahrain:

International Information Systems
PO Box 3086
Flat 31, Building 123 Block 320
Exhibition Road
Manama
Bahrain
Tel 293821, 292040 · Fax 293408 · Code +973

Belgium:

Software Marketing Group
rue E. Van Ophemstraat 40
B-1180 Brussels
Belgium
Email pbuysse@netdirect.be
Tel 02 376 57 42 · Fax 02 376 09 85 · Code +32

Brazil:

Datasafe Produtos de Informática e Serviços Ltda
Rua Santa Justina, 336 Gr. 108
Itaim
04545-041 Sao Paolo SP
Brazil
Email datasafe@originet.com.br
Tel 011 822 1129 · Fax 011 822 1129 · Code +55

Channel Islands:

Softek Services Ltd
20 Peter Street
St Helier
Jersey
JE2 4SP
Email sales@softek.co.uk
http://www.softek.co.uk/
Tel 01534 811182 · Fax 01534 811183 · Code +44

Croatia:

Qubis d.o.o.
Nova Cesta 1
10000 Zagreb
Croatia
Email qubis@zg.tel.hr
Tel 01 391461 · Fax 01 391294 · Code +385

Denmark:

Lamb Soft & Hardware
Lille Strandstraede 14
1254 Copenhagen K
Denmark
Email info@lamb-soft.dk
http://www.lamb-soft.dk/
Tel 3393 4793 · Fax 3393 4793 · Code +45

Finland:

Oy Protect Data Ab
PL 21
FIN-00701 Helsinki
Finland
Email karlerik.heimonen@protectdata.fi
http://www.protectdata.fi/
Tel 09 7525 2440 · Fax 09 7525 2210 · Code +358

France:

Racal-Datacom S.A.
18 Rue Jules Saulnier
93206 Saint-Denis Cedex
France
Email infos@racal-datacom.fr
Tel 01 49 33 58 00 · Fax 01 49 33 58 33 · Code +33

Germany:

NoVIR DATA
Hochofenstrasse 19-21
23569 Lübeck
Germany
Email 100141.2044@compuserve.com
Tel 0451 306 066 · Fax 0451 309 600 · Code +49

Hong Kong:

Racal-Datacom Limited
Sun House
181 Des Voeux Road
Central Hong Kong
Email w_chu@racal.com.hk
Tel 28158633 · Fax 28158141 · Code +852

Ireland:

Renaissance Contingency Services Ltd.
The Mews
15 Adelaide Street
Dun Laoghaire
Co Dublin
Ireland
Tel 01 280 9410 · Fax 01 280 8302 · Code +353

Italy:

Telvox s.a.s.
Via F.lli Cairoli 4-6
40121 Bologna
Italy
Email telvox.teleinf@bologna.nettuno.it
http://www.nettuno.it/fiera/telvox/telvox.htm
Tel 051 252 784 · Fax 051 252 748 · Code +39

Japan:

Computer Systems Engineering Co. Ltd.
23-2 Maruyamacho
Aletsusa Bldg.
Shibuya-ku
Tokyo 150-0044
Japan
Email pws@cseltd.co.jp
http://www.cseltd.co.jp/sweep/
Tel 03 3463 5633 · Fax 03 3496 7477 · Code +81

Malta:

Shireburn Co. Ltd.
Carolina Court
Guze Cali Street
Ta'Xbiex, Msd 14
Malta
Email info@shireburn.com
http://www.shireburn.com/
Tel 319977 · Fax 319528 · Code +356

Netherlands:

CRYPSYS Data Security
P.O. Box 542
4200 AM Gorinchem
The Netherlands
Email info@crypsys.nl
http://www.crypsys.nl/
Tel 0183 62 44 44 · Fax 0183 62 28 48 · Code +31

Forum Data Security
WG Plein 202
1054 SE Amsterdam
The Netherlands
Email info@forum-ds.nl
http://www.forum-ds.nl/
Tel 20 685 3486 · Fax 20 612 9702 · Code +31

New Zealand:

Wang New Zealand Ltd
P O Box 6648
Wellington
New Zealand
Email sophos@wang.co.nz
Tel 04 382 0100 · Fax 04 385 6067 · Code +64

Norway:

Protect Data Norge AS
Brobekkveien 80
0583 Oslo
Norway
Email pdn@protect.no
http://www.protect.no/
Tel 022 071500 · Fax 022 071501 · Code +47

Poland:

Safe Computing Ltd.
ul. Targowa 34
03-733 Warszawa
Poland
Email info@safecomp.com
http://www.safecomp.com/
Tel 022 6198956 · Fax 022 6700756 · Code +48

Portugal:

Década Informática s.a.
Apt. 7558
Estr. Lisboa/Sintra, Km 2,2
2720 Alfragide
Portugal
Email amandio.sousa@decada.mailpac.pt
Tel 01 471 2045 · Fax 01 471 2191 · Code +351

Singapore:

Racal Electronics (S) Pte. Ltd.
26 Ayer Rajah Crescent #04-06/07
Singapore 139944
Email sales@racal.com.sg
http://www.racal.com.sg/
Tel 779 2200 · Fax 778 5400 · Code +65

Slovakia:

Protect Data Slovakia
Kukolova 1
831 07 Bratislava
Slovak Republic
Email protectd@ba.sanet.sk
Tel 07 541 1527 · Fax 07 541 2210 · Code +421

Slovenia:

Sophos d.o.o.
Zwittrova 20
8000 Novo mesto
Slovenia
Email slovenia@sophos.com
Tel 068 322977 · Fax 068 322975 · Code +386

Spain:

Sinutec Data Security Consulting S.L.
Traversera de Gracia 54-56 Entlo. 3 y 4
08006 Barcelona
NIF B-60062502
Spain
Email sinutec@ysi.es
http://www.sinutec.com/
Tel 3-414.49.19 · Fax 3-202.14.25 · Code +34

Sweden:

Protect Datasäkerhet AB
Humlegardsgatan 20, 2tr
Box 5376
102 49 Stockholm
Sweden
Email info@protect-data.se
http://www.protect-data.se/
Tel 08 459 54 00 · Fax 08 459 54 10 · Code +46

Switzerland:

Performance System Software SA
Rue Jean-Pelletier 6
1225 Chene-Bourg
Geneva
Switzerland
Email jlt@pss.ch
http://www.pss.ch/
Tel 022 860 1030 · Fax 022 349 4775 · Code +41

Turkey:

Logic Bilgisayer Ltd
Esentepe Cad. Techno Centre 10/2
Mecidiyekoy
Istanbul
Turkey
Tel 0212 212 3664 · Fax 0212 212 3669 · Code +90

United States of America:

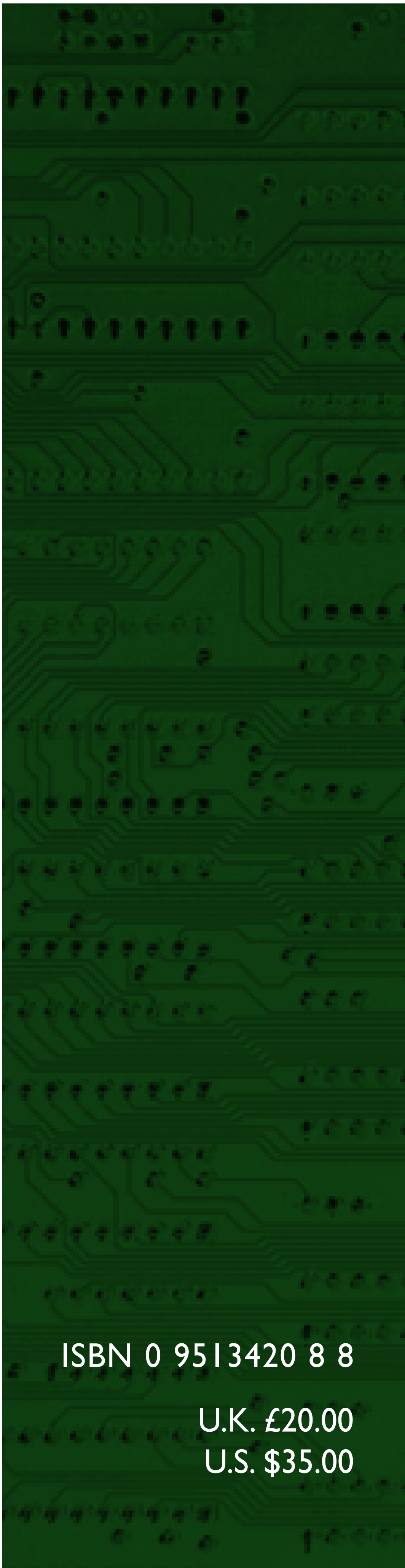
ACT
7908 Cin-Day Rd, Suite W
West Chester
Ohio 45069
USA
Email farrell@altcomp.com
http://www.altcomp.com/
Tel 513 755 1957 · Fax 513 755 1958 · Code +1

Uruguay:

Datasec
Patria 716
Montevideo 11300
Uruguay
Tel 02 7115878 · Fax 02 7115894 · Code +598

Sophos Plc • The Pentagon • Abingdon Science Park • Abingdon • OX14 3YP • England • Tel 01235 559933 • Fax 01235 559935
Sophos Plc • 2, Place de la Défense • BP 240 • 92053 Paris la Défense • France • Tel 01 46 92 24 42 • Fax 01 46 92 24 00
Sophos GmbH • Am Hahnenbusch 21 • D-55268 Nieder-Olm • Germany • Tel 06136 91193 • Fax 06136 911940
Sophos Inc • 18 Commerce Way • Woburn • MA 01801 • USA • Tel 781 932 0222 • Fax 781 932 0251
Email sales@sophos.com • http://www.sophos.com/

SOPHOS



ISBN 0 9513420 8 8

U.K. £20.00
U.S. $35.00

ISBN 0-9513420-8-8

9 780951 342084 02000