August 1990 ISSN 0956-9979

VIRUS BULLETIN

THE AUTHORITATIVE INTERNATIONAL PUBLICATION
ON COMPUTER VIRUS PREVENTION,
RECOGNITION AND REMOVAL

CONTENTS

EDITORIAL

A Corporate Strategy Emerges

Last month, *Shell UK* and the *Trustees Savings Bank* (TSB) made public their respective policies regarding computer viruses. At the *British Computer Society's* annual security conference in Solihull, representatives of these companies outlined the procedures which they have adopted to minimise the virus threat.

Both organisations have emphasised *education, training* and *awareness* as the key components to successfully controlling the threat posed by malicious software. Disciplinary measures have also been included reflecting Shell and TSB's philosophy that ignorance is no excuse for sloppy security. Employees at both organisations risk their pay rises and promotion prospects if they load unapproved software onto company computers.

The TSB has been hit by a spate of virus attacks in the last six months which prompted the formulation of a mandatory code of conduct. The bank announced that employees had loaded the AIDS Information Diskette onto company computers in December of last year which prompted security officials to devise company-wide anti-virus instructions. A further virus outbreak in March of this year by the WDEF virus on the bank's Apple Macs convinced management that a code of conduct should be drafted and distributed to the bank's 2,000 PC users.

All software for use on TSB computers must now be acquired by the bank's *personal computing support team*. The use of public domain software has been forbidden, experimentation and development of software by users has been outlawed and the use of external databases, whether online or by tape/disk transfer, will be controlled by the personal computing support team.

For some months *VB*, in accordance with security specialists familiar with the corporate sector, has recommended *training* and *awareness* as the single most vital component to limiting the computer virus problem. The informed user is always the first line of defence. *Vigilance and caution provide higher protection than even the best anti-virus software*. *VB* has also urged the implementation of *software validation* and *software quality assurance sections* to screen incoming software and oversee program development and installation. It is encouraging that many businesses are adopting exactly these measures to combat computer viruses and other rogue software.

Shell and TSB are to be congratulated, both for their honesty in discussing the virus problem and for adopting the correct measures best suited to solving it. A number of large organisations are now taking this straightforward approach in tackling the problems associated with malicious software.

At a recent *IBC* conference in London, Mr. Charles Brookson of *British Telecom's General Directorate of Security* and Mr. David Evans, a computer security specialist with the *Inland Revenue* spoke openly about computer virus outbreaks in their respective organisations and the proven methods to combat the threat. Both organisations have adopted similar strategies, which again emphasise *education, training, awareness* and *software validation.*

The shared conclusions of computer security specialists in the corporate sector consistently stress the importance of these four factors above any purely technical solution. At last, a cost-effective corporate strategy is emerging. Glasnost has prevailed and a problem which seemed insoluble is becoming manageable.

WORLDWIDE

NIST Anti-Virus Initiative

The US *National Institue of Standards & Technology* is planning a joint effort with industry to develop comprehensive anti-virus protection for computer systems. Dennis Steinauer of NIST described the proposed venture as being a central clearing-house for computer virus information which would collate all existing research. The consortium would also develop technology and techniques to combat the threat. Negotiations between NIST and ADAPSO, the computer software and services industry association are currently taking place. *IBM*, *Microsoft* and *Lotus Development Corporation* have been reported as interested in the venture.

Information: NIST, Computer Security Division, A-216, Gaithersburg, MD 20899, USA. Tel (301) 975 3359.

Japanese Virus Survey

The Kyodo news service has reported the results of a computer virus survey undertaken by the *Japanese Information Technology Promotion Agency*. 13.2 percent of 379 respondents described virus infections as causing "serious damage". The survey was directed at 500 business, academic and research organisations. Apple Macintosh viruses were most prevalent with 46 cases reported.

Australian Research Centre

The *Queensland University of Technology* is to receive government sponsorship to undertake computer virus research, software evaluation and to collate information from around the world about the virus threat for distribution to Australian government departments and commercial businesses.

Information: Professor W Caelli, Director ISRC, Queensland University of Technology, PO Box 2434, 2 George Street, Brisbane, Queensland 4001, Australia.

STOP-PRESS

Apocalypse Now? The PC Today Incident

At 7:47 pm on July 24th an alert appeared on the UK CIX virus/general conference. The report stated that the UK magazine *PC Today* (*Vol. 4, No 4*) contained a floppy disk infected by the (Ogre) Disk Killer boot sector virus. *The situation appeared grave as a total of 40,000 such disks were reported as being distributed*.

VB commenced 'in-house' testing of two suspect disks purchased from local newsagents on the morning of July 25th. Jim Bates of the *Virus Information Service* also obtained two disks for analysis which were later supplemented by two further disks obtained directly from the publisher - *Database Publications, Europa House, Adlington Park, Adlington, Macclesfield SK10 5NP, UK*.

VB contacted *Database Publications* and spoke to David Hirst, the company's Commercial Director. A total of 56,500 disks had been duplicated and distributed to subscribers and newsagents. The findings at two test sites showed the virus code, which was present on the four initial disks examined, was inactive. A *VB* fax warning was deemed unnecessary. At 2:42 pm *VB* received a fax message from Ian Sharpe, Features Editor of *PC Today*, confirming these findings.

Technical Analysis

Disk label: *PC Today, Volume 3, August 1990 Disc Library. PowerMenu v5.3 The easy way to harness the hidden power of MSdos*. Supplied on a single 5.25 inch floppy disk with the UK magazine *PC Today*.

Virus code (Disk Killer) was present on all 5.25 inch disks examined. Three bad clusters were indicated in both copies of the FAT as 288, 289 and 290. They contained no executable code, and the character F6H (Format Filler) was found throughout. The files on the disks did not all occupy contiguous clusters, some of which were not sequential. Three files were fragmented in two sections - MENU.ECT, PCTODAY.EXE and EXTRAS.COM - the last of these straddled the bad sectors.

The Bug that Averted Disaster

The following (previously unnoticed) bug in the Disk Killer prevented the disaster in this case. If a completely blank floppy disk is infected with the virus, an uninitialised counter in the routine which searches through the FAT for free clusters will cause the wrong 3 clusters to be labelled as bad. The Disk Killer will store the rest of its code in clusters labelled as free and available to DOS. If sufficient data is then copied onto the disk, the virus code residing in free clusters will be overwritten by the data. The only working part of the virus which will stay on the disk will be the code in the boot sector. The examination of the bad sectors will reveal that they are filled with hex F6, which is there as the result of the original FORMAT.

This must have been the sequence of events which led to the infection of the master prepared by the magazine. A PC infected by the Disk Killer was used to FORMAT the floppy. The virus infected the disk at that stage. The rest of the programs were then transferred onto the disk, overwriting the non-boot part of the virus.

The Actual Effects

The actual *PC Today* disk is unlikely to be dangerous and is certainly not infectious. Minor problems will arise if an attempt is made to boot from it but it is unlikely that damage will be sustained. The stored sector record within the virus code did not match the position of the bad sectors on the disks examined. This record in Disk killer is a word located at offset 42 (hex) pointing to cluster 32 which is part of a file DISKMAN.OVL containing no virus code. The first byte of this file is an illegal op-code and cannot be executed. Such op-codes usually generate an error and either halt processing or return through an error trapping routine. The error will occur before most error traps are loaded. This caused test machines booted from the *PC Today* disk to 'hang'.

Conclusions

PC Today's Features Editor Ian Sharpe described this incident as a 'nightmare' in an interview published in *The Guardian* newspaper on the 25th July. *Database Publications Ltd.* have now implemented procedures to screen all software which is distributed with the company's publications. It was ironic that *PC Today's* July edition contained an anti-virus program which, in this case, was either unused or ineffectual.

The company should conduct an enquiry to find out:

1. *How did virus code enter the disk preparation system and remain undetected?*

2. *How could virus code be passed for duplication without detection?*

3. *Why was the code not identified by security controls at the duplication stage?*

Analysis confirmed that the incident was far from apocalyptic. However, a future incident involving the duplication of thousands of disks containing a destructive and functioning virus (particularly a parasitic virus) would prove an unmitigated disaster. **In the light of this incident, the closest to a catastrophe so far, PC magazine publishers must act responsibly and impose proper controls on the development and distribution of software.** (*See item 1, End-Notes, p. 24*).

FOR MANAGEMENT

The Enemy Within

In 1985 Donald Burleson, a 35 year old programmer of Fort Worth, Texas, USA, added a routine to his company's payroll processing program which checked for the existence of his own payroll number. Burleson was later dismissed from the insurance company, the *USPA & IRA Co*., (*United Services Planning Association, Inc*. and *The Independent Research Agency for Life Insurance, Inc. Texas*) an insurance brokerage firm. The subsequent removal of his number from payroll calculations triggered numerous 'logic bombs' on 21 September, 1985. **In this manner Burleson erased 168,000 client records.**

The case amply demonstrates the devastation that the 'insider' - an attacker with intimate knowledge of a site's computer systems - can unleash. In the catalogue of malicious programs, computer viruses are relative newcommers. This article outlines some other forms of programmed attack.

Time Bomb

A simple trigger routine added to an existing program or contained within virus code which monitors the operating systems internal clock. Time bombs are a sub-set of logic bombs (*see below*). Due to their simplicity and compact coding, 'site-specific' time bombs are easily concealed within the hundreds of thousand of lines of machine code contained in business mainframe or network systems.

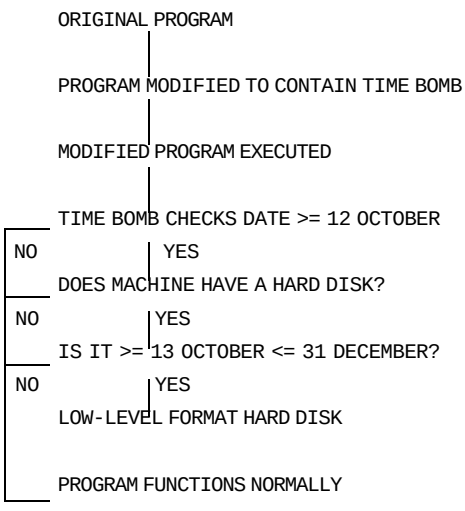

Figure.1 Simple time bomb similar to that found in the Datacrime virus

Logic Bomb

A logic bomb is a section of code embedded in an existing program which checks whether pre-specified conditions exist. The programmer dictates both the set of conditions to be met and the effects which take place after the logic bomb triggers.

Logic bombs are the traditional weapon of the crooked systems analyst or programmer, providing a means for extortion or blackmail. They are incorporated in a specific system making each logic bomb unique in its structure and location.

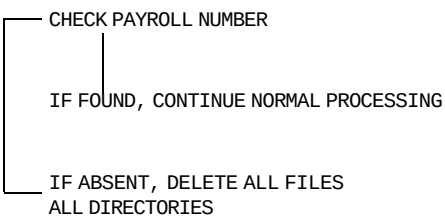

Figure.2 Classic logic bomb

The coding is usually compact and extremely difficult to detect. A classic logic bomb will search for the programmer's payroll ID number and trigger should that number fail to appear in two consecutive payroll calculations.

Logic bombs can be configured in any way that the programmer so wishes. Conditions that might trigger a logic bomb are limitless - the absence or presence of files, specific keyboard entry sequences, memory capacity, a particular node in use at a specified time and so on.

Logic bombs which are triggered by input data usually trigger after a specific input transaction is submitted. In other cases, the extortionist will provide an input routine to *defuse* a bomb program once the victim organisation submits to his/her demands. Another devious tactic is to introduce a combined time/logic element whereby, for instance, the payroll ID number is absent for a period of six months before the program triggers. This provides a sufficient time lapse for the attacker to avert suspicion.

Bomb programs are nearly always implanted by software developers or other 'insiders' with legitimate access to the computer system. Logic bombs are primarily a hazard in

mainframe/minicomputer environments and are not yet regarded as a PC problem. This is because most mainframe and mid-range computer software is developed internally whereas microcomputer software is delivered in compiled form.

Security efforts should be aimed at preventing the installation of bomb programs. The insidious, virtually undetectable nature of these programs and their capacity to cause crippling side effects makes them (*along with site-specific Trojan horses, see below*) a particular concern.

Control is lost once skilled programmers gain access to a system (even if they are granted restricted systems privileges).The need for vigorous vetting prior to the employment of computer staff (particularly analysts, programmers and systems managers) cannot be over-emphasised.

Trojan Horses

In traditional computer security, a Trojan horse is a program containing illicit functions designed to increase the attacker's systems privileges or divulge information labelled at a higher level of confidentiality than the attacker's rights permitted.

These programs were named by Dan Edwards of the US *National Security Agency* in 1972. Their name is derived from the wooden horse in Greek mythology within which hid a party of Greek soldiers. The story is related in Homer's *Iliad.* The citizens of Troy wheeled the horse into their besieged city believing it to be an offering of surrender. At night the Greek soldiers crept out of the horse to open the city gates and welcomed an invasion force - a classical example of compromised security by illicit means.

However, the definition of a Trojan horse as a tool to assist information disclosure has been expanded to encompass all programs which perform services beyond those stated in their specifications. Trojan horses are usually modified programs containing concealed, often destructive, functions. The most sophisticated Trojans comprise functioning programs specifically written to contain such code.

PC Trojan Horses

Trojan horses are common as jokes and malicious pranks within a variety of programming environments, not least in MS-DOS and PC-DOS computers. Notification of many malicious PCs programs is made by means of the 'The Dirty Dozen'. This is a comprehensive, updated list of Trojanised and 'hacked' programs describing their effects. 'The Dirty Dozen' was originally compiled to assist Bulletin Board SysOps screen malicious code from their systems.

One of the principal means for Trojan horse distribution is via bulletin boards. Most Trojans appear as useful and, in most cases, fully-functioning games or utilities. This increases the likelihood that they will be downloaded, copied and executed thereby maximising the number of victims. Unfortunately, it easy to rename a Trojanised program thus providing it with a new lease of life after its initial discovery which undermines attempts at notification. It is also relatively easy to insert destructive code into legitimate programs.

ARC513.EXE	This hacked version of SEA's ARC.EXE appears normal. However, it writes over track 0 of your [hard] disk upon usage, destroying the disk's boot sector.
BACKTALK	This once beneficial utility will write/destroy sectors on your [hard] disk drive. Use this with caution if you acquire it, because it is more than likely that you got a bad copy.
COOKIES.EXE	This file, which purports to explain the secret of Mrs. Field's cookies, really scrambles FAT tables.
DANCERS.BAS	This trojan shows some animated dancers in color, and then proceeds to wipe out your [hard] disk's FAT table. There is another perfectly good copy of DANCERS.BAS ON BBS's around the country; apparently the author altered a legitimate program to do his dirty work.
EGABTR	BEWARE! Description says something like "improve your EGA display", but when run it deletes everything in sight and prints "Arf! Arf! Got you!"
SECRET.BAS	BEWARE!! This may be posted with a note saying it doesn't work, and would someone please try it. If you do try it, however, it will format your disks.

Figure 3. A selection of PC Trojans from 'The Dirty Dozen'. This is a comprehensive updated listing of Trojan horses, 'hacked' and pirated programs to help BBS SysOps screen uploaded files.

It is available on many BBSs or as a $10.00 mailing list from *The Dirty Dozen List, c/o Eric Newhouse, 40 Whitney Tavern Rd, Weston, MA 02193, USA.*

Trojan Horses and Confidentiality

The original definition of a Trojan horse was of a program that 'leaked' confidential information, e.g. creating a 'blind copy' of a sensitive file for the creator. A *Trojan horse direct release* is a program which causes classified data to be released to a user with a lesser security classification. The Trojan re-labels data with a lower security classification and thus leaks this information to the attacker. This can be done by:

- changing the security label

- placing data in a field or record with lower classification

- failing to update a security classification as the sensitivity of data increases

The Trojan horse route of penetration is fundamentally impossible to prevent on nearly all computer systems.

Four essential steps are involved in a Trojan horse direct release program.

1. Create / modify program to perform illicit act. Do not arouse user suspicion. Make the program enticing.

2. Make the program accessible by installing it, or getting a user to install it in a system library.

3. The victim must run the program, either accidentally (the Trojan replaces an existing program) or intentionally (it is invoked).

4. The attacker reaps the benefits of the illicit action. Private information is copied into a repository which the attacker later accesses.

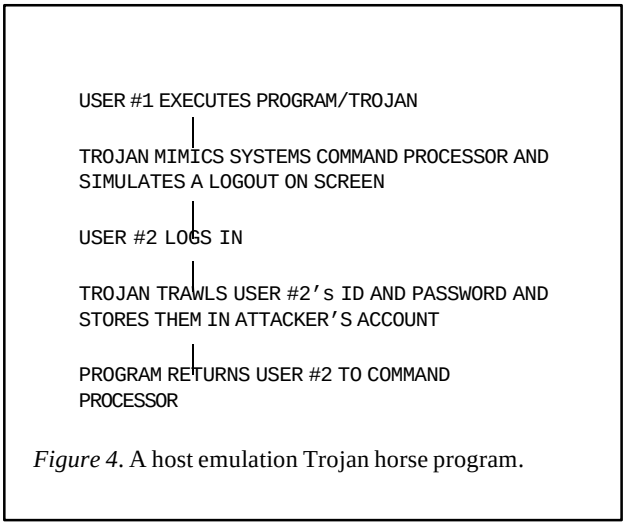

Figure 4. A host emulation Trojan horse program.

Conclusions

Malicious or fraudulent programs developed 'in house' by computer staff who have intimate knowledge of the computer systems are 1) *extremely difficult to prevent* 2) *extremely difficult to detect* 3) *extremely pernicious.*

Programs of this sort are usually implanted in complex processing environments and are designed to defraud or inflict damage. Preventing these programs is dependent upon preventing a potential attacker from gaining access to your computer systems. Extensive and comprehensive pre-employment vetting, including character references and past employment records, will help to screen applicants for sensitive computing posts. Contractors and short term employees are particularly dangerous. No vetting of such transient staff is usually undertaken. Audit and security software (RAC-F, ACF-2 etc.) are powerful weapons, but a determined and devious insider will find a way to circumvent such controls.

Systems controls can be enforced by restricting 'in-house' programming. Eliminating user programming involves getting rid of compilers, assemblers, interpreters and similar applications. However, eliminating the ability to write any type of program imposes severe limitations. Remember also, that the people most likely to devise a bomb or Trojan program will be programmers or analysts who by definition must have access to the system. Moreover, networked systems are vulnerable (regardless of programming restrictions) to Trojan programs installed on uncontrolled nodes and spread over the network.

'In-house' software development should be subject to tight controls including source code verification and extensive testing. Equally, change control should be vigorous during system and software enhancement.

Users should be taught to recognise and report irregularities in program libraries or directories. Under no circumstances should unfamiliar 'foreign' programs be executed. Remember, also, that a well conceived Trojan will disguise itself, often appearing under a legitimate name and emulating a legitimate program.

Software from external sources can be screened for obvious defects or virus code by standard software validation. However, searching for malign code (as opposed to virus code) is neither a viable or practical proposition in a business environment. This level of security applies to sensitive military or government systems where software and upgrades are supplied from the developer as source code which is analysed and then compiled internally.

In next month's issue: *David Ferbrache, Information Systems Integrity & Security Ltd., explains the compromise of classified data in high security systems through the use of Trojan horse programs which exploit covert channels.*

KNOWN IBM PC VIRUSES

This is a list of the known viruses affecting IBM PCs and compatibles, including XTs, ATs and PS/2s. The first part of the list gives aliases and brief descriptions of viruses which have been seen, while the second part lists viruses which have been reported. Each entry consists of the virus group name, its aliases and the virus type (See "Type codes" table). This is followed by a short description (if available) and a 10 to 16 byte hexadecimal pattern which can be used to detect the presence of the virus by the "search" routine of disk utility programs such as *The Norton Utilities* or your favourite disk scanning program. Offset (in hexadecimal) normally means the number of bytes from the virus entry point. For parasitic viruses, the infective length (the amount by which the length of an infected file has increased) is also given.

Type codes:

C = Infects COM files **E** = Infects EXE files **D** = Infects partition boot sector (Logical sector 0 on disk)

M = Infects disk boot sector (Track 0, head 0, sector 1 on disk) **N** = Not memory-resident after infection

R = Memory-resident after infection

SEEN VIRUSES

8 Tunes - CER: The virus probably originates in Germany and infects COM and EXE files. The length of the virus code is 1971 bytes. When triggered, it will play one out of 8 different tunes. The virus attempts to deactivate two anti-virus programs: Bombsquad and Flushot+.

```
8 Tunes          33F6 B9DA 03F3 A550 BB23 0353 CB8E D0BC ; Offset variable
```

405 - CN: Infects one COM file (on a different disk) each time an infected program is run by overwriting the first 405 bytes. If the length of the file is less than 405 bytes, it will be increased to 405. The virus only infects the current directory and does not recognise a file already infected.

```
405              26A2 4902 26A2 4B02 26A2 8B02 50B4 19CD ; Offset 00A
```

800 - CR: Infective length is 800 bytes. The virus code is written into random location of the infected file. Like Number of the Beast, it uses an undocumented DOS function to obtain the original INT 13H address, and instead of intercepting INT 21H, it intercepts INT 2A, function 82. The virus is encrypted. (*VB June 90*)

```
800              B981 0151 AD33 D0E2 FB59 3115 4747 E2FA ; Offset 00E
```

5120 - CEN: This is one of the largest viruses known, 5120 bytes long. When an infected program is run, it will search recursively for EXE and COM files to infect. Infected programs will terminate with an "Access denied" message after 1st June 1992. Parts of the virus seem to have been written in compiled BASIC.

```
5120             40B1 04D3 E88C DB03 C305 1000 8ED8 8C06 ; Offset 026
```

4K, 4096, Frodo, IDF, Israeli Defence Forces - CER: Infective length is 4096 bytes. The virus may occasionally cause damage to files, as it manipulates the number of available clusters, which results in files becoming crosslinked. If the virus is resident in memory, it disguises itself from detection by pattern-searching or checksumming programs. Infected systems hang on 22nd September. (*VB May 90*)

```
4K               E808 0BE8 D00A E89A 0AE8 F60A E8B4 0A53 ; Offset 239
```

Agiplan - CR: Infective length is 1536. The virus attaches itself to the beginning of COM files. Agiplan has only occurred on one site and may be extinct.

```
Agiplan          E9CC 0390 9090 9090 9C50 31C0 2E38 26DA ; Offset 0 (?)
```

AIDS - CN: Not to be confused with the AIDS Trojan, this virus overwrites COM files and is about 12K long. When an infected program is executed, the virus displays "Your computer now has AIDS" and halts the system.

```
AIDS             0600 AE42 6E4C 7203 4600 0004 00A0 1000 ; Offset 2C7F
```

Alabama - ER: Infective length is 1560 bytes. May cause execution of wrong files and FAT corruption.

```
Alabama          8CDD 33DB 8EDB 8B07 0B47 0274 7489 1F89 ; Offset 109
```

Ambulance - CN: The major effect of this virus is to display an ambulance on the screen. The virus is 796 bytes long.

```
Ambulance        0001 8A07 8805 8B47 0189 4501 FFE7 C3E8 ; Offset 016
```

Amoeba - CER: Virus adds 1392 bytes to the length of the infected files. It does not have any known side-effects.

```
Amoeba           CF9C 502E A107 0140 2EA3 0701 3D00 1072 ; Offset 0D1
```

Amstrad - CN: Adds 847 bytes to the front of any COM file in the current directory. The rest contains an advertisment for Amstrad computers. (*VB June 90*). Cancer is a 740 byte long mutation, which infects the same files repeatedly.

```
Amstrad          C706 0E01 0000 2E8C 0610 012E FF2E 0E01 ; Offset 114
```

Armagedon - CR: A 1079 byte virus from Greece, which interferes with the serial port. It will produce control strings for Hayes-compatible modems, dialling number 081-141 (speaking clock in Crete). Virus name is spelt with a single 'd'.

```
Armagedon        018C CBEA 0000 0000 8BC8 8EDB BE00 01BF ; Offset 3F0
```

Brain, Ashar, Shoe - DR: Consists of a bootstrap sector and 3 clusters (6 sectors) marked as bad in the FAT. The first of these contains the original boot sector. In its original version it only infects 360K floppy disks and occupies 7K of RAM. It creates a label "(c) Brain" on an infected disk. There is a variation which creates a label "(c) ashar".

```
Brain            A006 7CA2 097C 8B0E 077C 890E 0A7C E857 ; Offset 158
```

Cascade, Fall, Russian, Hailstorm - CR: This encrypted virus attaches itself to the end of COM files, increasing their length by 1701 or 1704 bytes. The encryption key includes the length of the infected program, so infected files of different lengths will look different. After infection it becomes memory-resident and infects every COM file executed, including COMMAND.COM. The original version will produce a "falling characters" display if the system date is between 1st October and 31st December 1988. The formatting version will format the hard disk on any day between 1st October and 31st December of any year except 1993. Both activations occur a random time after infection with a maximum of 5 minutes. (*VB Sept 89*)

```
Cascade (1) 01   0F8D B74D 01BC 8206 3134 3124 464C 75F8 ; Offset 012, 1701 bytes, Falling characters
Cascade (1) 04   0F8D B74D 01BC 8506 3134 3124 464C 75F8 ; Offset 012, 1704 bytes, Falling characters
Cascade (1) Y4   FA8B CDE8 0000 5B81 EB31 012E F687 2A01 ; Offset 000, 1704 bytes, Falling characters
Cascade format   0F8D B74D 01BC 8506 3134 3124 464C 77F8 ; Offset 012, 1704 bytes, Formats hard disk
```

Dark Avenger - CER: The virus infects when a file is opened and closed as well as when it is executed. This means that a virus-scanning program will cause it to infect every program scanned. Infective length is 1800 bytes. It only infects if program is at least 1775 bytes long and it may overwrite data sectors with garbage. There is a mutation which extends the file by 2000 bytes. (*VB Feb 90*)

```
Dark Avenger     A4A5 8B26 0600 33DB 53FF 64F5 E800 005E ; Offset variable
```

Datacrime - CN: The virus attaches itself to the end of a COM file, increasing its length by 1168 or 1280 bytes. On execution of an infected program, the virus searches through the full directory structure of drives C, D, A and B for an uninfected COM file which will be infected. Files with 7th letter D will be ignored (including COMMAND.COM). If the date is on or after 13th October of any year, the first 9 tracks of the hard disk will be formatted. The format is low level after displaying the message:

```
DATACRIME VIRUS
RELEASED: 1 MARCH 1989
```

This message is stored in an encrypted form in the virus. (*VB Aug 89*)

```
Datacrime (1)    3601 0183 EE03 8BC6 3D00 0075 03E9 0201 ; Offset 002, 1168 bytes
Datacrime (2)    3601 0183 EE03 8BC6 3D00 0075 03E9 FE00 ; Offset 002, 1280 bytes
```

Datacrime II - CEN: This encrypted virus attaches itself to the end of a COM or EXE file, increasing their length by 1514 bytes. The virus searches through the full directory structure of drives C, A and B for an uninfected COM or EXE file. It ignores any file if the second letter is B. If the date is on or after 13th October of any year, but not a Monday, a low level format of the first 9 tracks will be done on the hard disk after displaying the message ''DATACRIME II VIRUS'' which is stored in encrypted form. Datacrime IIB displays the message ''* DATACRIME *''. (*VB Aug 90*)

```
Datacrime II     2E8A 072E C605 2232 C2D0 CA2E 8807 432E ; Offset 022, 1514 bytes
Datacrime IIB    2BCB 2E8A 0732 C2D0 CA2E 8807 43E2 F3BD ; Offset 01B
```

dBASE - CR: Transposes bytes in dBASE (DBF) files. Creates the hidden file BUGS.DAT in the root directory of drive C and generates errors if the absolute difference between the month of creation of BUGS.DAT and the current month is greater or equal to 3. Infective length is 1864 bytes. The destroy version destroys drives D to Z when the trigger point is reached. (*VB Dec 89*)

```
dBASE            50B8 0AFB CD21 3DFB 0A74 02EB 8A56 E800 ; Offset 636, 1864 bytes
dBASE destroy    B900 01BA 0000 8EDA 33DB 50CD 2658 403C ; Offset 735, 1864 bytes
```

December 24th - ER: A mutation of the Icelandic (3) virus. It will infect one out of every 10 EXE files run, which grow by 848-863 bytes. If an infected file is run on December 24th, it will stop any other program from running and display the message "Gledileg jol" (Merry Christmas in Icelandic).

```
December 24th      C606 7E03 FEB4 5290 CD21 2E8C 0645 0326 ; Offset 044
```

Den Zuk, Search - DR: The majority of the virus is stored in a specially formatted track 40, head 0, sectors 33 to 41. When Ctrl-Alt-Del is pressed, the virus intercepts it and displays "DEN ZUK" sliding in from the sides of the screen. This does not happen if KEYBUK or KEYB is installed. Den Zuk will remove Brain and Ohio and replace them with copies of itself.

```
Den Zuk            FA8C C88E D88E D0BC 00F0 FBB8 787C 50C3 ; Offset 0
```

Devil's Dance - CR: A simple virus which infects COM files, adding 951 bytes at the end of infected files. The virus is believed to have originated in Spain or Mexico. It monitors the keyboard and will destroy the FAT after 5000 keystrokes.

```
Devil's Dance      B800 0150 8CC8 8ED8 8EC0 C306 B821 35CD ; Offset 011
```

Disk Killer, Ogre - DR: The virus infects floppy and hard disks and if the computer is left on for more than 48 hours, it will encrypt the contents of the bootable disk partition. The infection of a disk occurs by intercepting a disk read - INT 13H function 2. When the virus triggers, it displays the message "Disk Killer — Version 1.00 by Ogre Software, 04/01/1989. Warning !! Don't turn off the power or remove the diskette while Disk Killer is Processing!". (*VB Jan 90*)

```
Disk Killer        2EA1 1304 2D08 002E A313 04B1 06D3 E08E ; Offset 0C3
```

Do-nothing - CR: A badly-written virus from Israel that assumes a 640K system.

```
Do nothing         8CCA 8EDA BA00 988E C2F3 A41E B800 008E ; Offset 020
```

Durban, Saturday 14th - CER: Adds 669 bytes to the end of infected files. On any Saturday 14th the first 100 logical sectors of drives C, then B and then A are overwritten.

```
Durban             B911 00A4 E2FD B4DE CD21 80FC DF74 47C6 ; Offset 02F
```

Dyslexia, Solano - CR: Virus adds 1991 bytes in front of the infected file and 9 bytes at the end. Occasionally transposes two adjacent characters on the screen.

```
Dyslexia           B4C0 CD21 3D34 1275 0E2E 8B0E 0301 1E07
```

Eddie-2, 651 - CER: A non-destructive virus from Bulgaria. It marks infected files with a value of 62 in the seconds field of the timestamp, which makes them immune from infection by Vienna or Zero Bug. Infected files grow by 651 bytes, but this will not be seen if a DIR command is used - the virus intercepts the find-first and find-next functions, returning the correct (uninfected) length. (*VB June 90*)

```
Eddie-2            D3E8 408C D103 C18C D949 8EC1 BF02 00BA ; Offset 02D, 651 bytes
```

E.D.V. - DR: E.D.V. marks infected disks with "EV" at the end of the boot sector and stores the original boot sector code in the last sector of the last track on 360K disks, just like the Yale virus. Program crashes and data loss have been reported on infected systems.

```
E.D.V.             0C01 5083 EC04 B800 01CF B601 B908 2751 ; Offset 0C1
```

Fellowship - ER: This 1019 byte virus attaches itself to the end of EXE files, damaging them by overwriting the last 10 bytes or so. Other effects are being analysed.

```
Fellowship         33DB 8EDB FF36 0000 FF36 0200 C706 0000 ; Offset 039
```

Fish 6 - CER: A partial mutation of 4K having an infective length of 3584 bytes. The virus is encrypted and the decryption routine is so short that it is impossible to extract a hex pattern longer than 14 bytes. The virus seems to activate in 1991, but the exact effects are yet unknown.

```
Fish 6             E800 005B 81EB A90D B958 0D2E 8037 ; Offset 0
```

Flip - CER: The primary effect of this 2343 byte virus is to "flip" the screen by rotating it through 90 degrees. The virus is encrypted and self-modifying. **No search pattern is possible**.

Form - BR: A boot sector virus from Switzerland infecting hard disks and floppy disks. On the 24th day of every month the virus produces a small delay when keys are pressed.

```
Form               B106 D3E0 8EC0 33FF B9FF 00FC F3A5 06B8 ; Offset 074
```

Fu Manchu - CER: The virus attaches itself to the beginning of a COM file or to the end of an EXE file. Infective length is 2086 bytes (COM) and 2080 (EXE). It is a rewritten version of the Jerusalem virus, but the marker is "rEMHOr" and the preceding "sU" is "sAX" (Sax Rohmer, creator of Fu Manchu). After installing itself as memory-resident, it will infect any COM or EXE file, except COMMAND.COM. EXE files are infected only once, unlike the original Jerusalem. One in sixteen times on infection a timer is installed, which will trigger a display "The world will hear from me again" after a random number of half-hours (max. 7.5 hours). The machine then reboots. The same message is also displayed on pressing Ctrl-Alt-Del, but the virus does not survive the reboot. If the date is after 1st August 1989, the virus monitors the keyboard buffer and adds

derogatory comments to the names of politicians (Thatcher, Reagan, Botha and Waldheim), overstrikes two four-letter words, and displays "virus 3/10/88 - latest in the new fun line!" if "Fu Manchu" is typed. All messages are encrypted. (*VB July 89*)

```
Fu Manchu       FCB4 E1CD 2180 FCE1 7316 80FC 0472 11B4 ; Offset 1EE, 2086 bytes COM, 2080 bytes EXE
```

GhostBalls - CN: A strain of Vienna virus. Seconds field changed to 62, as in Vienna. Infective length is 2351 bytes and the virus attaches itself to the end of the file. When run, it will infect other COM files and try to place a modified copy of the Italian virus into boot sector of drive A. This copy of the Italian runs on 286 machines but is non-infective. Virus contains text "GhostBalls, Product of Iceland".

```
GhostBalls      AE75 EDE2 FA5E 0789 BC16 008B FE81 C71F ; Offset 051
```

Hallochen - CER: A virus which reputedly originated in West Germany. It contains two text strings (o in Hallochen is character code 148 decimal):

```
Hallochen !!!!!!, Here I'm..
Acrivate Level 1..
```

The virus will not infect "old" files. If the value of the month or year fields in the time stamp is different from the current date, the file will not be infected. The virus will only infect files longer than 5000 bytes, increasing their length by 2011 bytes.

```
Hallochen       EB8C C903 D98E D3BC DB08 53BB 2E00 53CB ; Offset 01E, 2011 bytes
```

Icelandic, Saratoga - ER: The virus attaches itself at the end of an EXE file and after becoming memory-resident, it will infect only one in ten (one in two for the Icelandic (2) mutation) programs executed. When a program is infected, the disk is examined and if it has more than 20 MBytes, one cluster is marked as bad in the first copy of the FAT. There is a mutation which does not flag clusters. Version (1) will not infect the system unless INT 13H segment is 0700H or F000H, thus avoiding detection by anti-virus programs which hook into this interrupt. Version (3) does not flag clusters and bypasses all interrupt-checking programs.

```
Icelandic (1)   2EC6 0687 020A 9050 5351 5256 1E8B DA43 ; Offset 0C6, 656 bytes
Icelandic (2)   2EC6 0679 0202 9050 5351 5256 1E8B DA43 ; Offset 0B8, 642 bytes
Icelandic (3)   2EC6 066F 020A 9050 5351 5256 1E8B DA43 ; Offset 106, 632 bytes
```

Italian, Pingpong, Turin, Bouncing Ball, Vera Cruz - DR: The virus consists of a boot sector and one cluster (2 sectors) marked as bad in the first copy of the FAT. The first sector contains the rest of the virus while the second contains the original boot sector. It infects all disks which have at least two sectors per cluster and occupies 2K of RAM. It displays a single character "bouncing ball" if there is a disk access during the one-second interval in any multiple of 30 minutes on the system clock. The original version will hang when run on an 80286 or 80386 machine, but a new version has been reported which runs normally. If a warm boot is performed after the machine hangs, an uninfected disk will still become infected. (*VB Nov 89*)

```
Italian-Gen     B106 D3E0 2DC0 078E C0BE 007C 8BFE B900 ; Offset 030
Italian         32E4 CD1A F6C6 7F75 0AF6 C2F0 7505 52E8 ; Offset 0F0
```

Jerusalem, PLO, Friday the 13th, Israeli - CER: The virus attaches itself to the beginning of a COM file or at the end of an EXE file. When an infected file is executed, the virus becomes memory-resident and will infect any COM or EXE program run, except COMMAND.COM. COM files are infected only once, while EXE files are re-infected every time that they are run. Infective length is 1813 bytes (COM) and 1808 bytes (EXE). The virus finds the end of EXE files from the information in the file header, and if this is less than the actual file length, the virus will overwrite part of the file. After the system has been infected for 30 minutes, row 5 column 5 to row 16 column 16 on the screen are scrolled up two lines, creating a "black window". The system then slows down, due to a time-wasting loop installed on each timer interrupt. If the system is infected when the date is set to 13th of any month which is also a Friday, every program run will be deleted. (*VB July 89*). Jerusalem mutations matching the search pattern:

Anarkia: Virus signature is changed from 'sURIV' to 'ANARKIA'.

Anarkia-B: Minor mutation of Anarkia.

Mendoza: Another minor mutation of Anarkia.

PSQR: Mutation with the signature changed to 'PSQR'. The infective length is 1715 (COM) and 1720 bytes (EXE).

```
Jerusalem       03F7 2E8B 8D11 00CD 218C C805 1000 8ED0 ; Offset 0AC, 1813 bytes COM, 1808 bytes EXE
Jerusalem-USA   FCB4 E0CD 2180 FCE0 7316 80FC 0372 11B4 ; Offset 095
PSQR            FCB8 0FFF CD21 3D01 0174 3B06 B8F1 35CD ; Offset 071
```

Jo-Jo - CR: This is a non-encrypted version of Cascade with the encryption code patched out and a few other changes made.

```
Jo-Jo           B800 F08E C0BF 08E0 813D 434F 751B 817D ; Offset 0D2
```

July 13th - ER: This encrypted virus will activate on 13th July, but its exact effects have not yet been determined. It is 1201 bytes long.

```
July 13th       2EA0 1200 3490 BE12 00B9 B104 2E30 0446 ; Offset variable
```

Kennedy - CN: A simple COM infecting virus, probably originating from Sweden. When an infected file is run, it will infect a single COM file in the current directory, expanding it by 333 bytes at the end. The virus activates on three dates: 6th June, 18th November and 22nd November and displays the message

```
        Kennedy er dod - lange leve "The Dead Kennedys"
```

```
Kennedy          E817 0072 04B4 4FEB F38B C505 0301 FFE0 ; Offset 035
```

Korea - DR: A simple boot sector virus with no side-effects. It may cause damage to data, as the original boot sector is always written to sector 11.

```
Korea            31C0 8ED8 8ED0 BCF0 FFFB BB13 048B 0748 ; Offset 008
```

Lehigh - CR: The virus only infects COMMAND.COM. It is 555 bytes long and becomes memory-resident when the infected copy is run. If a disk is accessed which contains an uninfected COMMAND.COM, the copy is infected. A count of infection generation is kept inside the virus, and when it reaches 4 (or 10 in a mutated version), the current disk is trashed each time a disk is infected, provided that (a) the current disk is either in the A drive or B drive, (b) the disk just infected is either the A drive or B drive and (c) the disk just infected is not the current one. The trashing is done by overwriting the first 32 sectors following the boot sector. Infection changes the date and time of COMMAND.COM.

```
Lehigh           8B54 FC8B 44FE 8ED8 B844 25CD 2106 1F33 ; Offset 1EF
```

Liberty - CER: A virus from Indonesia with an infective length of 2873 bytes. No harmful effects have been reported, but the virus is awaiting disassembly.

```
Liberty          0174 031F 595B 5053 5152 1E06 1E0E 1FE8 ; Offset 080
```

Macho - CEN: Swaps every string "MicroSoft" with "MachoSoft" on the hard disk. Searches 20 sectors at a time, storing the last sector searched in IBMNETIO.SYS which is marked hidden and system. After searching the last sector it starts again. This will only happen after 1st January 1985 and if the environment variable VIRUS is not set to OFF. Infective length is 3550 to 3560 bytes. Random directory search for uninfected files. Infects COMMAND.COM. This virus is closely related to Syslock.

```
Macho            5051 56BE 5900 B926 0890 D1E9 8AE1 8AC1 ; Offset ?
```

Mistake, Typoboot, Typo - DR: Exchanges letters for phonetically similar ones (for example "C" and "K") while they are being output to the printer. Reportedly written in Israel. A mutation of the Italian virus with about 35 % of the code rewritten. The boot sector is almost identical to the Italian.

```
Mistake          32E4 CD1A 80FE 0376 0A90 9090 9090 52E8 ; Offset 0F0
```

MIX1 - ER: The virus infects only EXE files, attaching itself to the end. When an infected program is run, the virus will copy itself to the top of the free memory. Some programs may overwrite this area, causing the machine to crash. The virus traps printer and asynch interrupts and corrupts traffic by substituting characters. 50 minutes after infection, the virus alters Num Lock and Caps Lock keyboard settings. 60 minutes after infection, a display similar to the Italian virus (bouncing ball) will be produced. The virus will infect every tenth program run. Infected files always end in "MIX1" and the infective length of MIX1 is 1618 to 1633 bytes and MIX1-2 1636 to 1651 bytes. (*VB Dec 89*)

```
MIX1             B800 008E C026 803E 3C03 7775 095F 5E59 ; Offset 02E
MIX1-2           B800 008E C0BE 7103 268B 3E84 0083 C70A ; Offset 02A
```

Murphy - CER: Two versions exist. One produces a click from the loudspeaker when any DOS functions are called while the other may produce the bouncing-ball effect when the user enters ROM BASIC. The virus will only activate between 10:00 and 11:00 a.m.

```
Murphy           B44A CD21 8CC0 488E D8C7 0601 0008 00E8 ; Offset variable
```

New Zealand, Stoned, Marijuana - MR: The virus consists of a boot sector only. It infects all disks and occupies 2K of RAM. On floppy disks, sector 0 is infected, while on the hard disks the physical sector 0 (Master boot sector) is infected. The original boot sector is stored in track 0 head 1 sector 3 on a floppy disk and track 0 head 0 sector 2 on a hard disk. The boot sector contains two character strings: "Your PC is now Stoned!" and "LEGALISE MARIJUANA" but only the former one is displayed, once in eight times, and only if booted from floppy disk. The version (2) stores the original boot sector at track 0 head 0 sector 7 on a hard disk. The second string is not transferred when a hard disk is infected. One version displays the message ''Your PC is now Sanded!''. A mutation has been reported in Australia which also displays "LEGALISE MARIJUANA". (*VB May 90*)

```
New Zealand (1)  0400 B801 020E 07BB 0002 B901 0033 D29C ; Offset 043
New Zealand (2)  0400 B801 020E 07BB 0002 33C9 8BD1 419C ; Offset 041
```

Number of the Beast, 666, V512 - CR: An advanced virus from Bulgaria, only 512 bytes long. The length of the file does not appear to increase since the virus overwrites the first 512 bytes of the programs it infects with itself, storing the original 512 bytes in the unused space of a disk cluster, after the logical end of file. A mutation has now appeared. (*VB May 90, June 90*)

```
Number of Beast  5A52 0E07 0E1F 1EB0 5050 B43F CBCD 2172 ; Offset 0A3
No. of Beast 1   B800 3DCD 2193 5A52 0E1F 1E07 B102 B43F ; Offset variable
```

Ohio - DR: Boot sector virus, probably an older version of Den Zuk.

```
Ohio            FAFA 8CC8 8ED8 8ED0 BC00 F0FB E845 0073 ; Offset 02B
```

Old Yankee - EN: This is the first of the viruses which play the "Yankee Doodle Dandy". It only infects EXE files, increasing their length by 1961 bytes. When an infected program is run, it will infect a new file and then play the melody. (*VB June 90*)

```
Old Yankee      03F3 8CC0 8904 0E07 53B8 002F CD21 8BCB ; Offset 009
```

Oropax, Music virus - CR: The length of infected files increases between 2756 and 2806 bytes and their length becomes divisible by 51. 5 minutes after the infection, the virus plays three different tunes at 7-minute intervals. Does not infect COMMAND.COM.

```
Oropax          06B8 E033 CD21 3CFF 7423 8CCE 8EC6 8B36
```

Pentagon - DR: The virus consists of a boot sector and two files. The sample obtained does not work, but it contains the code which would survive a warm boot (Ctrl-Alt-Del). It could only infect 360K floppy disks, and will look for and remove Brain from any disk it infects. It occupies 5K of RAM.

```
Pentagon        8CC8 8ED0 BC00 F08E D8FB BD44 7C81 7606 ; Offset 037
```

Perfume - CR: The infected program will sometimes ask the user a question and not run unless the answer is 4711 (name of a perfume). The virus will look for COMMAND.COM and infect it. Infective length is 765 bytes.

```
Perfume         FCBF 0000 F3A4 81EC 0004 06BF BA00 57CB ; Offset 0AA
```

Pixel - CN: The Pixel viruses are practically identical to the Amstrad virus, although they are shorter: 345 and 299 bytes. No side-effects are noticeable until the 5th generation is reached, at which stage there is a 50 % chance that the following message will appear when an infected program is executed:

```
Program sick error: Call doctor or buy PIXEL
for cure description
```

(*VB June 90*)

```
Pixel (1)       0E1F 2501 0074 4CBA D801 B409 CD21 CD20 ; Offset 0C8, 354 bytes
Pixel (2)       BA9E 00B8 023D CD21 8BD8 061F BA2B 01B9 ; Offset 033, 299 bytes
Pixel (3)       0001 0001 2E8C 1E02 018B C32E FF2E 0001
```

Pretoria, June 16th - CN: Overwrites the first 879 bytes of infected files with a copy of itself, and stores the original 879 bytes at the end of the file. When an infected program is executed, the virus searches the entire current drive for COM files to infect. On 16th June the execution of an infected file will cause all entries in the root directory to be changed to 'ZAPPED'. The virus is encrypted.

```
Pretoria        AC34 A5AA 4B75 F9C3 A11F 0150 A11D 01A3
```

PrintScreen - DR: Occasionally performs a Print Screen (PrtSc) operation.

```
Printscreen     FA33 C08E D0BC 00F0 1E16 1FA1 1304 2D02 ; Offset 023
```

Prudents - EN: Infective length is 1205 bytes and the virus will destroy the last 32 bytes of any infected file. Activates during the first four days of May of every year, turning every write operation into a verify operation, which results in the loss of data.

```
Prudents        0E07 BE4F 04B9 2300 5651 E87E 0359 5EE8 ; Offset 055
```

Shake - CR: A primitive 476 byte virus which reinfects already infected files. Infected programs sometimes reboot when executed. Occasionally, infected programs display the text "Shake well before use !" when executed.

```
Shake           B803 42CD 213D 3412 7503 EB48 90B4 4ABB
```

South African, Friday the 13th, Miami, Munich, Virus-B - CN: Infective length is 419 bytes, but some reports suggest mutations with an infective length between 415 and 544 bytes. Does not infect files with Read-Only flag set. Virus-B is a non-destructive mutation containing South African 2 pattern. COMMAND.COM is not infected. Every file run on a Friday 13th will be deleted.

```
South African 1 1E8B ECC7 4610 0001 E800 0058 2DD7 00B1 ; Offset 158
South African 2 1E8B ECC7 4610 0001 E800 0058 2D63 00B1 ; Offset 158
```

Stealth, 1260 - CN: Virus infects COM files, adding 1260 bytes to them. The first 39 bytes contain code used to decrypt the rest of the virus. A variable number of short (irrelevant) instructions are added between the decoding instructions at random in an attempt to prevent virus scanners from using identification strings. **No search pattern is possible**. (*VB Mar 90*)

Sunday - CER: Variation of Jerusalem. Infective length is 1631 bytes (EXE) and 1636 (COM). Activates on Sunday and displays message "Today is SunDay! Why do you work so hard? All work and no play make you a dull boy.". There are unconfirmed reports of FAT damage on infected systems.

```
Sunday          FCB4 FFCD 2180 FCFF 7315 80FC 0472 10B4 ; Offset 095
```

Suomi - CN: A 1008 byte virus from Finland, which uses self-modifying encryption, like the Stealth virus. The virus seems to disinfect already infected files under certain conditions, but COMMAND.COM seems to remain permanently infected. No harmful side-effects have been reported, but the virus is awaiting disassembly. **No search pattern is possible**.

Suriv 1.01, April 1st COM - CR: A precursor to Jerusalem infecting only COM files with the virus positioned at the beginning of the file. Infective length is 897 bytes. If the date is 1st April, the virus will display "APRIL 1ST HA HA HA YOU HAVE A VIRUS" and the machine will lock. If the date is after 1st April 1988, the virus produces the message "YOU HAVE A VIRUS !!!" but the machine will not lock. The virus is memory resident and will not infect COMMAND.COM. (*VB Aug 89*)

```
Suriv 1.01          0E1F B42A CD21 81F9 C407 721B 81FA 0104 ; Offset 304, 897 bytes
```

Suriv 2.01, April 1st EXE - ER: A precursor to Jerusalem infecting only EXE files with the virus positioned at the beginning of the file. Infective length is 1488 bytes. If the date is 1st April, the virus will display "APRIL 1ST HA HA HA YOU HAVE A VIRUS". If the year is 1980 (DOS default) or the day is Wednesday after 1st April 1988, the machine will lock one hour after infection. (*VB Aug 89*)

```
Suriv 2.01          81F9 C407 7228 81FA 0104 7222 3C03 751E ; Offset 05E, 1488 bytes
```

Suriv 3.00, Israeli - CER: An earlier version of Jerusalem infecting COM and EXE files and displaying the side-effects 30 seconds after infection instead of 30 minutes. Infective length is 1813 bytes (COM) and 1808 bytes (EXE). Program delete does not work. (*VB Aug 89*)

```
Suriv 3.00          03F7 2E8B 8D15 00CD 218C C805 1000 8ED0 ; Offset 0B0, 1813 COM, 1808 EXE
```

Svir - EN: A simple 512 byte virus with no side effects. Svir means "music" in Bulgarian.

```
Svir                33F6 4626 8B0C E302 EBF8 8BD6 83C2 04E8 ; Offset 049
```

Swap - DR: Does not infect until ten minutes after boot. One bad cluster on track 39, sectors 6 & 7 (head unspecified). Uses 2K of RAM. Infects floppy disks only. Does not store the original boot sector anywhere. Virus creates a display similar to Cascade, but is transmitted via boot sector.

```
Swap                31C0 CD13 B802 02B9 0627 BA00 01BB 0020 ; Offset ?
```

Sylvia - CN: The virus displays messages including "This program is infected by a HARMLESS Text-Virus V2.1", "You might get an ANTIVIRUS program....." when an infected program is executed, but if the above text is tampered with, the (encrypted) messages "FUCK YOU LAMER !!!!", "system halted....$" will be displayed. The victim is told to send a 'funny postcard' to a genuine address of a Dutch woman called Sylvia. When an infected program is run, the virus will look for 5 COM files on drive C and the current drive. COMMAND.COM, IBMBIO.COM and IBMDOS.COM are not infected. The virus adds 1301 bytes to the beginning of the infected files and 31 bytes at the end.

```
Sylvia              CD21 EBFE C3A1 7002 A378 0233 C0A3 9E02 ; Offset 229
```

Syslock - CEN: This encrypted virus attaches itself to the end of a COM or an EXE file. Infective length is 3551 bytes. It infects a program one in four times when executed. Will not infect if the environment contains SYSLOCK=@.

```
Syslock             8AE1 8AC1 3306 1400 3104 4646 E2F2 5E59 ; Offset 0, 3551 bytes
```

Taiwan - CN: The virus activates on the 8th day of every month and overwrites the FAT and the root directory of drives C and D. Two variants are known with different infection lengths: 708 and 743 bytes.

```
Taiwan              07E4 210C 02E6 21FB B980 0033 F6BB 8000 ; Offset 0A0
Taiwan (2)          07E4 210C 02E6 21FB B980 00BE 0000 BB80 ; Offset 065
```

Tenbyte, Valert - CER: This virus was by accident posted to the V-ALERT electronic mail list recently. Adds 1554 bytes to infected files. Activates on 1st September corrupting data written to disk.

```
Tenbyte             1E0E 1F8D 36F7 04BF 0001 B920 00F3 A42E ; Offset 0
```

Tiny - CN: A mutation of the Kennedy virus only 163 bytes long. It has no side effects other than replication.

```
Tiny                408D 94AB 01B9 0200 CD21 B43E CD21 FFE5 ; Offset 088
```

Traceback, Spanish - CER: This virus attaches itself to the end of a COM or an EXE file. Infective length is 3066 bytes. It becomes memory-resident when the first infected program is run and will infect any program run. If the date is 5th December or later, the virus will look for and infect one COM or EXE file either in the current directory or the first one found starting with the root directory. If the date is 28th December 1988 or later, the virus produces a display similar to Cascade one hour after infection. If nothing is typed, the screen restores itself after one minute. Display will repeat every hour. Spanish is an earlier version with a reported infective length of 2930 or 3031 bytes. (*VB Sept 89*)

```
Traceback           B419 CD21 89B4 5101 8184 5101 8408 8C8C ; Offset 104, 3066 bytes
Spanish             E829 06E8 E005 B419 CD21 8884 E300 E8CE ; Offset ?
```

Typo, Typo COM, Fumble - CR: Infects all COM files in the subdirectory on odd days of every month. If typing fast, substitutes keys with the ones adjacent on the keyboard. Infective length is 867 bytes. (*VB Apr 90*)

```
Typo                5351 521E 0656 0E1F E800 005E 83EE 24FF ; Offset 01D, 867 bytes
```

Vacsina - CER: Infective length 1206 to 1221 bytes (COM) and 1338 to 1353 bytes (EXE). After a successful infection of a COM file, a bell is sounded. Infects any file loaded via INT 21 function 4B (load and execute), i.e. COM, EXE, OVL and APP (GEM) files. Checks version number of itself (current is 5) and replaces with newer code. A member of the "Bulgarian 50" (see Yankee). (*VB June 90*)

```
Vacsina (1)     8CC8 8ED8 8EC0 8ED0 83C4 02B8 0000 502E ; Offset variable
Vacsina (2)     E800 005B 2E89 47FB B800 008E C026 A1C5 ; Offset variable
```

Vcomm - ER: This virus first increases the length of infected programs so that it becomes a multiple of 512 bytes. Then it adds 637 bytes to the end of the file. The resident part will intercept any disk write and change it into a disk read.

```
Vcomm           80FC 0375 04B4 02EB 0780 FC0B 7502 B40A ; Offset 261
```

Victor - CEN: A 2442 byte virus from the USSR awaiting disassembly. The only known damaging effect is the corruption of the FAT.

```
Victor          8CC8 8BD8 B104 D3EE 03C6 50B8 D800 50CB ; Offset 0C8
```

Vienna, Austrian, Unesco, DOS62, Lisbon - CN: The virus infects the end of COM files. Infective length is 648 bytes. It looks through the current directory and the directories in the PATH for an uninfected COM file. One file in eight becomes overwritten. Seconds stamp of an infected file is set to 62. A number of mutations, shorter than the original, but functionally equivalent, have been reported in Bulgaria.

```
Vienna (1)      8BF2 83C6 0A90 BF00 01B9 ; Offset 005, 648 bytes
Vienna (2)      FC8B F281 C60A 00BF 0001 B903 00F3 A48B ; Offset 004, 648 bytes
Vienna (3)      FC89 D683 C60A 90BF 0001 B903 00F3 A489 ; Offset 004
Vienna (4)      FC8B F283 C60A BF00 01B9 0300 F3A4 8BF2 ; Offset 004, 623 bytes
Vienna (5)      CD21 0E1F B41A BA80 00CD 2158 C3AC 3C3B ; Offset variable
Vienna (6)      8E1E 2C00 AC3C 3B74 093C 0074 03AA EBF4 ; Offset variable
```

Virdem - CN: This virus was published in the R. Burger book "*Computer Viruses - A High Tech Disease*". Originally intended as a demonstration virus, but now also found in the wild. Infective length is 1336 bytes. Two versions are known to exist with texts in English and German. (*VB July 90*)

```
Virdem          BE80 008D 3EBF 03B9 2000 F3A4 B800 0026 ; Offset 011
Virdem-1        BE80 008D 3ED7 03B9 2000 F3A4 B800 0026 ; Offset 011
Virdem-Gen      434B 7409 B44F CD21 72AC 4B75 F7B4 2FCD ; Offset 098
```

Virus-90 - CN: The author of this virus is Patrick A. Toulme. He uploaded the virus to a number of Bulletin Boards, stating that the source was available for $20. When an infected program is run it will display the message "Infected", infect a COM file in drive A and display the message "Done". Infective length is 857 bytes.

```
Virus-90        558B 2E01 0181 C503 0133 C033 DBB9 0900 ; Offset 01E
```

Virus-101 - CER: This virus has been written by the same author as Virus-90. The virus is encrypted. Awaiting disassembly.

Virus-B - CN: 'Test virus' which was available as a restricted access file from the *Interpath Corporation* BBS in the USA. It is a mutation of the South African, with the destructive code of the original disabled. The identification pattern is the same as for the South African virus.

VP - CN: Contains a variable number (1 to 15) of NOPs at the beginning followed by 909 bytes of virus code. When an infected program is run, the virus may attempt to locate, infect and execute another program.

```
VP              0001 FCBF 0001 B910 00F2 A4B8 0001 FFE0 ; Offset variable
```

W13 - CN: A primitive group of viruses from Poland, based on the Vienna virus. They have no known side-effects and there are two versions, 534 and 507 bytes long. The version with 507 bytes has some bugs corrected.

```
W13             8BD7 2BF9 83C7 0205 0301 03C1 8905 B440 ; Offset variable
```

XA1 - CN: The XA1 virus overwrites the first 1539 bytes of infected COM files with a copy of itself and stores the original code at the end of the file. On 1st April the boot sector will be overwritten, causing the computer to 'hang' on the next boot. The virus will also activate on 21st December and stay active until the end of the year. It will then display a Christmas tree and the text:

```
        Und er lebt doch noch: Der Tannenbaum!
        Frohe Weihnachten
XA1             B02C 8846 FF8B 7E00 884E FE8A 4EFF 000D ; Offset 01E
```

Yale, Alameda, Merritt - DR: This virus consists of a boot sector and infects floppies in drive A only. It becomes memory-resident and occupies 1K of RAM. The original boot sector is held in track 39 head 0 sector 8. The machine will hang if the virus is run on an 80286 or 80386 machine. If a warm boot is performed after the machine hangs, an uninfected disk will still become infected. It has not been assembled using MASM and contains code to format track 39 head 0, but this is not accessed. Survives a warm boot.

```
Yale            BB40 008E DBA1 1300 F7E3 2DE0 078E C00E ; Offset 009
```

Yankee - CER: This is a member of the "Bulgarian 50" group of viruses, which consists of some 50 related versions, all written by the same person. Vacsina viruses belong to the same group. All the viruses in the group will remove infections by older versions, and the size varies from 1200 to 3500 bytes. The Yankee viruses will play the tune "Yankee Doodle Dandy", either at 5:00 p.m. or when Ctrl-Alt-Del is pressed.

```
Yankee          0000 7402 B603 520E 5143 CFE8 0000 5B81 ; Offset variable
```

Zero Bug, Palette - CR: Infective length is 1536 bytes and the virus attaches itself to the beginning of COM files. The virus modifies the number of seconds to 62 (like Vienna). If the virus is active in memory and the DIR command is issued, the displayed length of infected files will be identical to that before the infection. When the virus activates, a "smiley" (IBM ASCII character 1) may appear on the screen, and "eat" all zeros found.

```
Zero Bug        81C9 1F00 CD21 B43E CD21 5A1F 59B4 43B0 ; Offset 100
```

REPORTED VIRUSES

1702 - CR: A new mutation of the Cascade virus. Some doubts if it exists.

Advent - CEN: Reported to be related to Macho and Syslock.

Century A - CER: As Jerusalem-C, but activation date is 1st January 2000. Destroys FAT.

Century B - CER: As Jerusalem-C, but produces a wait during the execution of BACKUP.COM.

Chaos - DR: A new and changed mutation of Brain.

Itavir - EN: A 3880 byte virus from Italy. Reported to activate when the computer has been running for more than 24 hours, and corrupt the boot sector. It also displays a message in Italian and outputs random data to I/O ports.

Jerusalem-A - CER: Does not display black-hole in the screen.

Jerusalem-B - CER: EXE re-infection bug removed.

Jerusalem-C - CER: No slow-down effect.

Jerusalem-D - CER: Destroys FAT in 1990.

Jerusalem-E - CER: Destroys FAT in 1992.

Missouri - D: Some doubt if it exists.

Nichols - D: Some doubt if it exists.

Novell - CER: A variant of Jerusalem, reported to attack Novell networks. According to some reports, it is identical to Jerusalem, but a virus, reported to be Netware specific, is currently under examination in the USA.

Poem - ?

Screen - CR: Infects all COM files in current directory, including any already infected, before becoming memory resident. Every few minutes it transposes two digits in any block of four on the screen.

Slow - CER: A 1701 byte virus reported to slow down the system.

Sublimal - CR: Adds 1496 bytes to COM files. Flashes "LOVE, REMEMBER?" for a fraction of a second on the screen. Reported to be an early version of the Dyslexia virus.

TROJAN HORSES

AIDS Information Diskette: Widely distributed disk which is an extortion attempt. Installs multiple hidden directories and files, as well as AIDS.EXE in the main directory and REM$.EXE in a hidden subdirectory ($ is the non-printing character FF Hexadecimal). *(VB Jan 90)*

```
REM$.EXE        4D5A 0C01 1E01 0515 6005 0D03 FFFF 3D21 ; Offset 0
AIDS.EXE        4D5A 1200 5201 411B E006 780C FFFF 992F ; Offset 0
```

FOR PROGRAMMERS

Fridrik Skulason

The Structure of Virus Infections - Part II: EXE Files

In the first part of this series of articles, the structure of virus-infected COM files was analysed. This article will deal with another type of executable files, generally called EXE files.

COM Versus EXE

COM files just contain binary code - ready to be loaded directly into memory and executed without modifications, but EXE files are more complex. They start with a header containing various information about the file.

This information includes:

1) The length of the file to be loaded, which is normally equal to the length of the file. If it is less than the true length, only a part of the file is loaded into memory, when it is executed. If it is greater, DOS produces the message "Error in EXE file" if an attempt is made to execute the file.

2) The initial execution address. As the value of the CS register depends on where in memory the file is loaded, the header only stores a value which locates the starting segment of the file in memory. The initial value of the IP is also stored.

3) The value to be given to the stack pointer (SS:SP registers). This is stored in the same form as the CS:IP combination.

4) A 16-bit checksum. DOS ignores this number and it is just set to 0 by many compilers.

5) The number of addresses needing relocation (see below).

Following the header is a relocation table. It contains a list of locations within the file which need relocation after loading. Unlike COM files, an EXE file may contain instructions like "MOV AX, SEG DATA", where the value of SEG DATA depends on where the file is loaded into memory. The relocation process involves adding the starting memory address to all the locations listed in the relocation table.

The final part of the EXE file is the load module, containing the code and data of the program. Execution may begin anywhere within the load module, not only at offset 0 as in the case of COM files.

Structure of Infections

Infected COM files could be divided into several groups, depending on their structure, but almost all infections of EXE files are caused by a similar method. The virus code is appended to the file, and the EXE file header modified, so the virus will gain control when the program is executed. The original CS:IP information is then stored somewhere within the virus.

There are some minor, yet significant differences between the various viruses, but before they are described, the two viruses which do not follow the general method must be mentioned.

sURIV 2.01

This virus, also known as "April 1st - EXE", is unique because it does not append the virus code to the files it infects. Instead, the virus code is inserted into the file, between the relocation table and the load module.

This introduces some unnecessary complications, as the the virus must update all the addresses in the relocation table, as the load module has been moved.

This virus was the first EXE infecting virus written, and the author seems to have realised that the infection process could be simplified considerably. At least, the Jerusalem virus, which was probably written by the same author, uses the "standard" method. It is unlikely that any future virus will use this method.

Disinfecting a file infected with the April 1st virus is a three step process. First the original header is restored and written to a file. Then the relocation table is read and the length of the file (1488) is subtracted from each entry. After skipping over the virus code, the rest of the file is written to the output file.

Vacsina

The Vacsina virus is actually not able to infect EXE files directly. Instead, the first time it attempts to infect an EXE file, the file is changed into a COM-structured file, by overwriting the first bytes with a JMP to the end of the file, where a small relocation program is located. This program is clearly derived from the program used in FORMAT.COM and other programs in some versions of DOS. An EXE file changed by Vacsina in this way is not infected by a virus, but the next time the virus encounters it, the program looks like a standard COM file and is infected as such. It must be

noted that this method limits the infection to files less than 64K in size.

The Standard Method

The rest of the known EXE infecting viruses are structurally similar, but there are some minor differences.

Storage of the Original Header

Some viruses store the entire original header somewhere within the virus code. In those cases disinfection is simply a matter of locating the header and writing it back to the beginning of the file. The virus code then becomes inactive, and may be removed by shortening the file. The viruses using this method are:

Traceback/Spanish
Hallochen
Yankee (all variants)
4K
Fish 6
July 13th
Datacrime II

Some extra complications arise in the case of the last two, as the virus code, including the original header is encrypted and before it can be restored, it must be decrypted.

The other viruses only store various pieces of information from the header. They must all store the original execution address, in order to be able to transfer control back to the original program. The following viruses store no additional information:

Icelandic (all variants)
Alabama
Vcomm
Amoeba
Fellowship
Murphy
Virus-101
Svir

In some cases the viruses change the location of the stack. This is done because the stack is often located right after the load module, and may therefore overwrite the virus, which is appended to the program, as described before. This may cause the infected program to crash, when executed, which draws attention to the virus. To prevent this, the following viruses switch to an internal stack while executing and restore the original stack pointer when they transfer control to the original program.

Jerusalem (all variants)
Fu Manchu
Dark Avenger
Syslock
Tenbyte
Durban
8 Tunes
Prudents
Liberty
Flip
Old Yankee
Eddie II
Victor
5120

A disinfection program must be able to locate the original CS:IP values and possibly SS:SP as well, possibly decrypting them, writing the values back to the header and shortening the file as necessary.

Program Length

The number of bytes added to a file when it is infected needs not be constant for any given virus. The Jerusalem virus, as well as its relatives, uses the information in the program header to determine where the virus is to be written, instead of using the true length of the file. Normally this causes no problems, but if the header information is incorrect, the virus will overwrite a part of the file, making full recovery impossible.

Many EXE infecting viruses first pad the programs they infect with a few bytes, so their length becomes a multiple of 16 bytes, before appending the virus code. This is done so the virus code can begin at a paragraph boundary. A disinfection program can remove the virus code, but may not be able to determine the original length of the file. The extra 1-15 bytes at the end of the disinfected file do not affect its execution in any way.

The Checksum

Some viruses change the checksum in the header of infected files, often in order to mark the file as infected. If the original value is stored within the virus, it can be restored by the disinfection program, or just set to 0. Leaving it unchanged does no harm, however, and may even make the program immune to a later infection by the same virus.

Next month: *The structure of boot sector infection.*

VIRUS DISSECTION

Jim Bates

Datacrime II - Refined Hatred

Many users regard the computer virus problem as little more than an irritation which does not warrant serious attention. The infantile messages and non-fatal effects produced by many viruses seem to support this view ("Your PC is now Stoned!", "The world will hear from me again.", bouncing balls, falling letters etc.). However, an increasing number of viruses contain deliberately vicious trigger routines. The most pernicious of them all is the latest in the Datacrime series.

Background

The original Datacrime virus caused a great deal of concern in continental Europe during the latter part of 1989, particularly in Holland where even the police distributed anti-virus software in an attempt to counter the threat. The first in the series (*VB, August 89*) attacked COM files only (excluding COMMAND.COM), added 1168 bytes to infected files, was not encrypted and displayed the message:

```
DATACRIME VIRUS
RELEASED: 1 MARCH 1989
```

The virus contained several mistakes in the coding and was swiftly followed by a minor update which was essentially similar but now added 1280 bytes to infected files. Both are referred to as the Datacrime virus.

Datacrime II

A substantial rewrite then made its appearance as Datacrime II. In this version, the virus code was encrypted with a variable key, leaving only 42 bytes of recognisable code for signature scanning programs to use. "Improvements" included widening the infection capability to include EXE files (and COMMAND.COM), changing the trigger message to

```
DATACRIME II VIRUS
```

and correcting still more errors in the code - the infection length becoming 1480 bytes.

Now we have another "update" which has been refined still further and is discussed in detail below. These latest two versions are referred to as **Datacrime II** and **Datacrime IIB** respectively.

The exact origin of this virus is unknown but the furore in Holland during October/November suggest that its source may be local to that area. Whatever the origin, someone, somewhere, fuelled by pure malice, is actively engaged in refining this evil piece of code.

The Trigger Routine

The common factor across all four versions is the trigger routine. This has some errors in the earlier releases which prevent it operating as intended on some AT and PS/2 type machines but these have been corrected in the latest version.

The intention is to low-level format track zero (heads 0 to 9) of the first physical hard drive thus destroying the contents of the Master Boot Sector, the first copy of the File Allocation Table and most of the second copy of the FAT. An incidental effect on many later model machines is to destroy the drive specific "signature" used to identify and categorise the drive characteristics.

The effect after triggering is that the drive is no longer recognised by the machine's POST (Power On Self Test) routines and effectively "disappears" from the machine configuration when it is rebooted. **Restoration of this drive signature is an involved process which will probably be beyond the technical capabilities of even the best customer support departments and might even entail the drive being returned to the manufacturer for reconfiguration**. Bearing in mind also that many of these machines will have relatively large capacity drives, probably configured into several partitions, the destruction of the MBR will result in the disappearance of all the partitions of that disk.

During examination and analysis of the code I inadvertently ran the trigger routine and instantly lost 44 MBytes of disk capacity on my first hard disk (partitioned over two drives) with virtually no hope of recovery. **Restoring backups at this point is impossible since the drive is no longer accessible by the system!** Restoring the drive is a major undertaking involving searching the ROM based Disk Parameter Table for the appropriate entry and then reformatting it for re-writing to the correct area of the disk.

Virus Analysis

Datacrime IIB is a non-resident, parasitic virus with an infective length of 1514 bytes. It attacks EXE and COM files including COMMAND.COM. Files with the second letter set to 'B' are exempt from infection. As a result IBMBIO.COM and IBMDOS.COM remain uninfected.

The virus code is encrypted with a key that changes with each infection and the initial unencrypted code (including the decryption routine) is 56 bytes long. This is sufficient to extract a usable recognition signature which is noted at the end of this report.

Since the code does not become resident, the virus only executes when an infected program is run.

The trigger date remains the same as in previous versions (ie: later than 12th October of any year) but with an added check such that if the weekday is a Sunday and the machine has no hard disk, or if the weekday is a Monday and the machine **does** have a hard disk - the trigger routine is **not** executed. This seems somewhat convoluted since on a machine with no hard disk, the trigger routine simply displays

```
* DATACRIME II *
```

(note the asterisks in this version) and hangs the system.

No damage is done to floppy diskettes and hard disk destruction is limited to the first physical drive on the system. There have also been some modifications to the infection indication mechanism.

Recognition Method

With this version, infected files are marked by having the seconds field of the file's directory entry set equal to the lower three bits of the minutes field plus 5 (the plus 5 is the modification). This identification system means that occasionally the virus will identify a target file as infected when in fact it is not. It also means that this version could re-infect files infected by earlier versions.

Infection

File infection takes place by appending the encrypted code to the end of the target file and then either modifying the initial instructions (in the case of COM files) or changing the relevant information within the file header (for EXE files). Thus it is theoretically possible to repair and disinfect infected files.

The code also contains a childish attempt to confuse certain types of disassembly/debugging routines by the introduction of self-modifying code within the decryption section.

Although the virus has no other deliberate effects, this version still has an error in the INT 24H restoration routine which will produce unpredictable effects when the first critical I/O error occurs after the virus has executed.

Virus Operation

The code is generally untidy with many minor errors and shows evidence of unfamiliarity with assembly programming techniques. Operation begins by executing a self-location sequence which provides a global index for data and code access throughout the remainder of the program. The data decryption key is then collected and used to decrypt the whole of the remainder of the code. This version does **not** encrypt the virus message separately but it does change the encryption key at each pass such that the next infection will produce a substantially different sequence of 1514 bytes.

The decrypted program code then proceeds to initialise four buffers beyond the end of the code. It then examines the INT 41H vector segment address to determine whether the host machine has a fixed disk and sets a flag to indicate this. A test is then made to see if the host program length was zero, which is obviously to facilitate the initial introduction of the newly written virus code.

A check is then made to determine whether the host is an EXE file or a COM type and although a flag is set to indicate this, the program manipulates various sections of the stored program header in either case.

The system date is then examined to see if the trigger date has been reached, this is the same as in other versions - later then 12th October in any year. A further routine tests the day of the week against the hard disk indicator flag which prevents the trigger routine from executing on Sundays for floppy only machines and on Mondays for hard disk machines. (*The virus writer possibly incorporated these 'non-triggering' criteria to test the virus' infection routine safely. Ed.*).

The trigger routine has been extensively modified from erroneous earlier versions in that after displaying the message, the code now generates its own format buffer and address fields before issuing a function 5 call to INT 13H. **This is the low-level track format function and is configured to affect the first physical drive only, track zero, heads 0 to 9.** On drives with less than 9 heads the resulting error is trapped and the routine simply aborts normally to sound the PC speaker and then goes into an infinite loop. This hangs the PC.

The construction of the format buffer makes no attempt to retain the original Sectors per Track setting and this will invariably require reformatting (in addition to the configuration signature problems discussed above) before the drive can be returned to normal use. If the trigger date has not been reached, processing branches to the start of the file search routine.

This begins by collecting and storing the current drive and directory settings and installing a temporary handler routine for critical I/O errors to prevent any interruptions if empty floppy drives are accessed.

The code then executes a primitive sequencing routine which handles routing of the file search into drives C: and A: (in that order). **There is also evidence here of plans for an alternative trigger point since a check is made of the hard disk flag which, if successful, would branch back into the trigger routine.** However, within the present code this check cannot succeed and so processing continues with the search for suitable files.

The search looks first for EXE and then for COM files within the current directory on drive C:. If no uninfected files are found, the search then continues with a similar sequence in the subdirectories from that point and eventually reverts to a similar sequence on drive A:. Once a suitable uninfected file is found, it is infected and the search routine aborts. Thus within any specific subdirectory, the EXE files will be infected first and COM files only when there are no more uninfected EXE files. **It should also be noted that the search sequence does not search upwards from the current sub-directory, and the floppy drive is only accessed if no suitable files are found on the hard drive.**

The pre-infection check routine rejects any file with a second letter of 'B' in the file name and also any file where the directory entry time field indicates that it is already infected.

Once a file passes this check, the date and attribute fields are collected and stored (and modified if necessary) before the file is opened for write access. The first 28 bytes of the file are then read into memory and tested for the 'MZ' header indicator used to identify EXE files. Thus renamed files are treated correctly regardless of their new name.

If the target file is an EXE type, then the original program entry point is stored, before the header is modified to include the addition of the virus code. COM files are simply checked for their length so that a jump offset can be calculated into the virus code.

With COM files, no attempt is made to ensure that there will be room for the 1514 bytes of virus code and this will cause "Insufficient Memory" problems for COM files greater than 63,765 bytes in length. The first 28 bytes of the original program are stored in either case for restoration when the program is executed. The final phase of infection involves first copying a special section of the virus code to a higher position in memory whence it can be invoked to encrypt the virus program.

A part of this section of code actually functions as an encryption 'toggle' whereby it switches the target code between encrypted and decrypted states. This routine is first called to encrypt the virus, the encrypted code is written to append to the end of the file and finally the routine is invoked again to decrypt the virus before processing returns into the main code for the completion and tidying up routines.

Tidying up involves the usual process of writing out the modified header to the target file and restoring the date and attribute fields to their original values and the time field to the 'infected' condition. The drive and directory defaults are then also restored and an attempt is made to de-install the critical error handler. However, as mentioned above and as in previous versions, this routine contains an error which means that subsequent calls to the INT 24H vector will produce unpredictable results.

Finally, the virus restores the original program header and passes processing to the host program.

Detection

With the exception of the actual trigger routine, this virus makes universal use of DOS service functions for file access and can thus be easily detected by resident DOS monitor programs of the Flushot+ type.

The original decryption routine is sufficiently large and individual for the extraction of a reliable hexadecimal search pattern and this is noted here (*also noted in VB March 1990*):

```
2E8A 072E C605 2232 C2D0 CA2E 8807 432E
```

This will be found at offset 22H of the virus code.

Conclusions

There is no doubt that Datacrime IIB is just the latest in a series of viruses emanating from a **single** author. The motivation behind them is not clear, suffice to say that these programs are the creations of a deranged mind.

The singularly destructive nature of the Datacrime series places these viruses in a unique category. Recovery in the event of a Datacrime virus triggering is a complex and involved procedure requiring considerable technical knowledge.

As a matter of urgency, anyone who knows the author of these viruses should expose his identity in order that this vandalism can be curtailed.

PRODUCT EVALUATION

Dr. Keith Jackson

Eliminator: Virus Detection & Removal

Eliminator is a PC based software package that claims to'... *not only recognise all known PC viruses, but can remove them and even prevent infected programs from executing'*. Eliminator is provided on both 3.5 inch and 5.25 inch floppy disks, so most types of user are catered for.

The manual that accompanies Eliminator is a 38 page A5 booklet. It explains very clearly how virus names are derived, and provides a list of 47 short virus descriptions. Each description includes such details as the length of the virus, any pseudonyms by which the virus is known, the method by which the virus infects, and a description of how the virus works. Many viruses are known to have both different code variants, and different names. These points are well covered in the manual.

The Programs

Eliminator has two component programs,**Virus Clean** and **Virus Monitor**. Virus Clean will detect any infected files or disks, and can be instructed to either remove viruses that are detected, or to delete the infected programs. Virus Monitor is a memory resident (TSR) program that will test every program that is executed to make sure that it is not virus infected. Eliminator is updated quarterly as new viruses appear.

Speed of Execution

I tested Virus Clean by letting it scan the whole of the hard disk on my test computer for viruses. Virus Clean reported that it had searched through 55 directories, containing 543 files, and 131 files had been tested for viruses (by default only COM and EXE files are searched). Virus Clean completed this test in the stunningly fast time of 42.3 seconds. Given that the files occupy 11.6 Mbytes on the hard disk, this corresponds to a search rate of about 4 seconds for every Megabyte of disk storage. Impressive for a 4.77 MHz 8088 processor.

For comparison, I scanned the same hard disk with two other well known virus-specific search programs: SCAN (from McAfee Associates in the USA) which took 4 minutes 55 seconds to complete this task, and SWEEP (from Sophos Ltd.) which took 7 minutes and 27 seconds. Therefore Virus Clean searches for viruses 7 times faster than SCAN, and 10 times faster than SWEEP, both of which are no slouches in terms of virus 'scanner' type programs.

This huge speed advantage can only come from one source, Virus Clean must only be checking the parts of COM and EXE files that definitely can be infected. This requires knowledge of how every virus infects. Such information can only be used when a virus has a)*been identified* b) *been disassembled* c) *been understood,* and d) *the method and location of infection has been determined.* The amount of work hidden behind Virus Clean's raw speed is prodigious.

Detection

Given the impressive speed offered by Virus Clean, I decided to test how well it can detect viruses. Floppy disks were scanned containing 49 unique viruses, with variants on the viruses taking the number of test samples used up to 101. The specific viruses used for testing are explained in the *Technical Details* section below. This test set has recently been doubled in size from the previously used set of 26 viruses, and will form the basis of all future testing of virus-specific products in *VB*.

Out of the 99 samples of parasitic viruses tested, Eliminator found a virus to be present in 66 files. Both boot sector virus samples were correctly detected. Eliminator did not spot samples of 18 different viruses: Amstrad, Anarkia, December 24th, Devils Dance, Hallochen, Kennedy, Number of the Beast, Perfume, Prudents, PSQR, Taiwan, Valert, Vcomm, Virdem, Virus-90, Virus-B, VP, and XA-1. The documentation does not appear to mention these viruses - Eliminator has not yet been programmed to detect them. PC Security says that the program is currently being upgraded.

By comparison, SCAN detected 86 files containing viruses, and from the set of viruses for testing used it had no knowledge of only the Eddie-2, Lehigh (curious given the longevity of this virus), Virdem, VP and W13 viruses. SCAN also had no knowledge of some of the many variants of the Vienna virus.

Not surprisingly, SWEEP which incorporates *Virus Bulletin* search patterns, located all the viruses used for testing, and found 114 matches in only 99 files. This super-abundance of viruses detected is accounted for by SWEEP sometimes finding the patterns of two individual viruses in one file, and finding the pattern for a particular virus more than once in other files. SWEEP's search patterns are not specific enough; the Devils Dance test pattern was found no less than four times within one test file.

Eliminator's lack of knowledge about 18 out of 49 viruses (over one third of the test sample) was disappointing. Virus Clean also reported that it had found a 'new' virus (or new variant of an existing virus) for six of the 99 parasitic viruses (and variants) tested. Virus Clean checksums such suspected virus variants to ascertain their true identity. If unknown, an on-screen message states that the virus cannot be removed, and requests that a copy of the virus is kept for future analysis by the author of Virus Clean.

Virus Specifity

We can argue about virus variants, and naming convention problems, as an explanation of this lack of knowledge. It is also possible that Eliminator searches for some viruses not contained in the *VB* test suite. However, the plain fact is that the amount of effort required from the author of Eliminator to be able to obtain the extremely fast scanning speed offered by Virus Clean requires that each virus is thoroughly disassembled, and its salient points of infection found.

Even though I admire the Herculean effort that has gone into the development of Virus Clean, the fact remains that it had no knowledge of over one third of the parasitic viruses used for testing. It is not a coincidence that out of these 18 viruses, 11 have been added to the list of known viruses since the last time that the complete list of PC viruses was published in the March issue of *VB*. They are of recent origin, and samples have only reached virus researchers relatively recently.

This neatly illustrates the major weakness of Virus Clean: it is virus-specific in the *extreme* - it needs intimate knowledge of how each virus operates. In the long run, such virus-specific measures are doomed. As the number of viruses proliferates it will become more and more difficult to keep pace. Indeed I would contend that by only being able to detect about two thirds of the test viruses used, Virus Clean is demonstrating this point quite succinctly.

As a tool to scan disks for outbreaks of a virus known to be detectable by the Virus Clean software it far surpasses anything on the market. For virus researchers, and for diagnostic purposes, it should prove invaluable. For generic anti-virus use, it poses the very real problem: *how do you know that the next virus attack will be one of those that Virus Clean knows about?*

A nail in the coffin for anti-virus programs that just search for search patterns, is the recent developments of viruses such as Stealth (formerly 1260), Suomi and Flip which encrypt the body of the virus with a random encryption key, and vary the small amount of code that is left unencrypted with each virus replication. It is noticeable that the test sample of the Stealth virus was not detected by Virus Clean.* Mutating viruses will surely follow to drive this point home further: general anti-virus measures must not be virus specific. Such methods are inadequate protection in the face of new and more sophisticated malicious programs. *(There are now three computer viruses which use self-modifying encryption routines, all of which require scanners to incorporate a virus identity, as opposed to searching for a hexadecimal recognition pattern. A fourth sample, Virus-101 is currently being analysed. See VB Table of Known Viruses, pp 9 - 16, Ed.)*

Virus Monitor and Memory Residence

The documentation provided with Eliminator states that the memory resident program known as Virus Monitor should be the first line of the AUTOEXEC.BAT file (a batch file that is always executed when a PC commences operation). Beyond requiring 3.5 Kbytes of available memory, no other details are given of how Virus Monitor operates.

This is unsatisfactory. If I am expected to have a program residing in memory while I use my computer, then I wish to know what other TSR software is compatible with the program and which memory resident programs conflict with Virus Monitor.

Two examples illustrate this quite graphically. If it is required that Virus Monitor is installed by being present as the first line of AUTOEXEC.BAT (see above), then it is absolutely required that the executable Virus Monitor file VM.COM is present in the root directory of the disk from which the system is booted. Nothing in the documentation mentions this fact, it merely says (very imprecisely): 'copy VM.COM on to your hard disk'. For the more naive users, an explanation of what AUTOEXEC.BAT actually is, where to find it, and how to modify it, would not go amiss.

From my own testing Virus Monitor appears to capture MS-DOS interrupts 8, 13, and 21 (in hex notation), but nothing in the manual tells me this even though this makes Virus Monitor vulnerable to contention from other memory resident programs which take the timer tick (interrupt 8H) for their own purposes.

Not providing **any** details about how Virus Monitor operates is a serious omission which should be rectified. Until this is the case, I would not recommend the use of Virus Monitor. Memory resident software is the greatest single source of incompatibility amongst PC programs.

Conclusion

Leaving Virus Monitor aside, I found the Virus Clean part of Eliminator to be the fastest program that I have ever tested for searching for known viruses, by some appreciable margin. However, it relies upon intimate virus knowledge for its blistering speed. It is probably the best tool that I have come across for virus investigators, but its extremely virus-specific nature makes it (like Scan and Sweep and all search programs) quite unsuitable for generic virus defence. The documentation omission is eminently rectifiable, the virus-specific nature of the lightning fast search speed is inherent.

Technical Details

Product: Eliminator

Vendor: PC Security Ltd., The Old Court House, Trinity Road, Marlow, Bucks. SL7 3AN, Tel: 0628 890390, Fax: 0628 890116.

Developer (and Copyright holder): Joe Hirst, Brighton, Sussex, UK.

Availability: IBM PC/XT/AT, PS/2, or compatible running MS-DOS, compatible operating system versions are not stated in the Eliminator documentation.

Version Evaluated: 1.15 6-A

Serial Number: None visible

Price: £79.00 per release. An upgrade service is available.

Hardware Used: ITT XTRA (a PC compatible) with a 4.77MHz 8088 processor, one 3.5 inch (720K) drive, two 5.25 inch (360K) drives, and a 30 Mbyte Western Digital Hardcard, running under MS-DOS v3.30.

Viruses used for testing purposes. This list of viruses has recently been expanded, and a suite of 49 unique viruses (according to the virus naming convention employed by *VB*), spread across 101 individual virus samples is now the standard *VB* test set. This comprises two boot viruses (Brain and Italian) and 99 parasitic viruses. There is more than one example of many of the viruses, ranging up to 10 different variants in the case of the Cascade and Vienna viruses. The actual viruses used for testing are listed below. Where more than one variant of a virus is available, the number of examples of each virus is shown in brackets.

Stealth, 405 (2), 4K (2), AIDS, Alabama, Amstrad (2), Anarkia, Brain, Cascade (10), Dark Avenger (2), Datacrime (3), dBASE, December 24th, Devils Dance, Eddie (2), Fu Manchu (3), GhostBalls, Hallochen, Icelandic (2), Italian, Jerusalem (6), Kennedy, Lehigh, Macho-Soft, MIX1 (2), Number of the Beast, Oropax, Perfume, Prudents, PSQR, South African (2), Suriv (8), Sylvia, Syslock (2), Taiwan, Traceback (4), Typo, Vacsina, Valert, Vcomm, Vienna (10), Virdem, Virus-90, Virus-B (2), VP, W13 (2), XA-1, Yankee (5), Zero Bug.

** Editor's note: Stealth is, according to the product vendor, detectable by Eliminator as Vienna 5. Our test version of Stealth was not detected.*

EDITORIAL POLICY

The Dreaded Table

The *VB Table of Known IBM PC Viruses* appears in this month's edition in its entirity. The current table occupies nine full pages despite attempts to diminish its sprawl through the use of small point sizes and decreased leading.

VB, having just entered its second volume, has reported a thirteen-fold increase in the number of such viruses since the first edition appeared fourteen months ago. We started publishing the table with every intention of its appearing in each and every edition. However, the proliferation of virus samples has caused a change of policy.

Monthly updates to the table will be published while the full table will be maintained and updated as a database. Additional information is continually added as samples are analysed and reveal their characteristics. The table will be published in full three times a year. Where possible, hexadecimal search patterns will be published. However, the appearance of viruses which render the extraction of such patterns impossible will be analysed and a short 'identity' or series of characteristics exclusive to each such sample will appear.

The explosion in the numbers of viruses appearing must inevitably bear on the anti-virus software industry. The demise of traditional virus-specific detection methods (as opposed to integrity checking methods) as a routine way of preventing viruses can already be perceived. No such package can ever claim to be comprehensive - a fact consistently highlighted in technical evaluations which *VB* has conducted of this type of software. Quite simply these search programs cannot guarantee protection against all viruses. Search programs consists of an engine and a series of patterns and/or identities to look for. As the number of viruses increases so will the time taken to scan disks for their presence - problems will eventually be encountered with scanning 'run-times'. Such programs also are inherently limited by the number of patterns which can be incorporated and searched for efficiently.

A hexadecimal search pattern can be extracted and incorporate to a search library in a matter of minutes. However, the work involved in disassembling and understanding encrypting self-modifying viruses in order to devise a reliable virus 'identity' is very time consuming. The number of samples attempting to evade detection by self-modifying encryption is is increasing and this trend looks set to continue.

Scanning programs will continue as a means of **diagnosis** - checking disks in environments infected by **known** viruses, screening software for common viruses and so on. However, the belief that such methods can provide generic defence is illogical and has been disproven conclusively.

END-NOTES & NEWS

In March of this year *VB* examined a disk called 'Ten of the Best' which was distributed by *Database Publications,* also the publisher of *PC Today (*see Stop Press, page 2*)*. The disk was sent out as a 'freebie' with *Personal Computing* (*Vol. 3 No. 1*). The examined disk contained a hacked but fully functioning version of the New Zealand virus which contained the message "Your PC is now Sanded!". Ian Sharpe of *Database Publications* was made aware of this incident following the investigation. (The titles *PC Today* and *Personal Computing* refer only to magazines published by *Database Publications Ltd*, Europa House, Adlington Park, Adlington, Macclesfield SK10 5NP, UK).

A Macintosh Trojan horse which vandalises PostScript RIPS and printers by resetting their passwords to an unknown value has been reported across the USA. Information from *DesktopTo Press*, USA. Tel 617 527 1899.

A Cascade (1704) contaminated demonstration disk containing CAS software (Concurrent Authoring System) has been received by a number of UK companies. Infected disk was sent out by *ICL Interactive Learning Systems,* Beaumont, Old Windsor, Berkshire SL4 2JP. It would appear that the disk became infected prior to duplication.

A Cascade (1701) infected file called WAITKEY.COM was uploaded by accident to the UK CIX *Computer Information Exchange.* The file was contained in a compressed archive REALITY.ZIP. Warnings have been posted and users who downloaded the file informed of the incident.

Products and Events

RG Software Systems, USA, has released the **Spanish version of Vi-Spy** (*VB, May 1990*). *The Instituto Interamericano para la Transferencia de Tecnologia*, San Jose, Costa Rica has been appointed sole distributor of Vi-Spy in South America. The institute, which runs courses in virus prevention and detection, has undertaken software and document translation.

S & S International will run a **Data Recovery Seminar** (October 4-5th) and a **Virus Threat Seminar** (November 8th - 9th). Both events will be held at Great Missenden, Bucks, UK. Tel 0490 791900.

Sophos, UK, continue a series of **computer virus workshops.** The next available dates are 11/12th September at the Rembrandt Hotel, London. Management and technical streams are available. Tel 0235 559933.

Microlease plc, UK, have announced a guaranteed virus-free PC rental service. All machines are thoroughly inspected for virus infection prior to delivery to hire customers. Tel 081 427 8822.

VIRUS BULLETIN

Subscription price for 1 year (12 issues) including delivery:

US$ for USA (first class airmail) $350, Rest of the World (first class airmail) £195

Editorial enquiries, subscription enquiries, orders and payments:

Virus Bulletin Ltd, 21 The Quadrant, Abingdon Science Park, Abingdon, OX14 3YS, England

Tel (0235) 555139, International Tel (+44) 235 555139
Fax (0235) 559935, International Fax (+44) 235 559935

US subscriptions only:

June Jordan, Virus Bulletin, 590 Danbury Road, Ridgefield, CT 06877, USA
Tel 203 431 8720, Fax 203 431 8165