

VIRUS BULLETIN

THE INTERNATIONAL PUBLICATION ON COMPUTER VIRUS PREVENTION, RECOGNITION AND REMOVAL

Editor: **Richard Ford**

Technical Editor: **Fridrik Skulason**

Consulting Editor: **Edward Wilding**,
Network Security Management, UK

IN THIS ISSUE:

- **New infection techniques.** The 3APA3A virus attacks the DOS system file IO.SYS in such a way that it cannot be detected by many current anti-virus programs. A full analysis of the technique and its implications is given on page 12.
- **How much of a problem?** The NCC survey on breaches of IT security has been collated and the parts relevant to virus attack extracted. What is the nature of the real virus problem, and how much does it cost? See p.14 for the results.
- **LZR on pre-formatted diskettes.** According to reports in Sweden and Finland, a large number of floppy disks have been distributed infected with the LZR virus. The full story is given on page 3.

CONTENTS

EDITORIAL	
The More the Merrier	2
VIRUS PREVALENCE TABLE	3
NEWS	
LZR Virus on Formatted Diskettes	3
Virus Mutation Toolkit?	3
IBM PC VIRUSES (UPDATE)	4
INSIGHT	
Auerbach: Viruses <i>Et Cetera</i>	7
VIRUS ANALYSES	
1. 3APA3A: The IO.SYS Hunter	9
2. The Peanut Vendor	11
TUTORIAL	
Virus Infection Techniques: Part 1	12
FEATURE	
IT Security Breaches: The 1994 NCC Survey	14
PRODUCT REVIEWS	
1. <i>Sweeping the Boards</i>	17
2. <i>EMD Armor Plus</i>	20
CONFERENCE REPORT	
<i>Compsec '94: Alive and Well</i>	23
END NOTES & NEWS	24

EDITORIAL

The More the Merrier

Virus Bulletin is in the process of moving offices - a job which requires much shifting of dusty boxes and reshuffling of paperwork. Like any move, the entire process has located many lost odds and ends which had dropped behind filing cabinets and printers (including several incriminating and amusing pictures, currently being held for a considerable ransom!).

Among the many finds has been the occasional diskette, somewhat dog-eared after a five-year sojourn in an inaccessible corner, but still readable. Such disks are a goldmine of information on the early years of *VB*, and contain such gems as unedited copies of articles written in the magazine's first year, information on the Aids Diskette, and early editions of *F-Prot*, *Sweep* and *Dr Solomon's Anti-Virus Toolkit*. All good fun, and an interesting glimpse into a little of the journal's history.

As the office is being sorted and packed, the number of diskettes has grown. Many have less-than-informative labels such as 'things' or 'Nov. 91', and more are devoid of identification. The total number of diskettes recovered has been something of a surprise (and an unpleasant one at that), and highlights one of the less obvious IT problems which can be brought about by a change of site.

Some of the software unearthed has not been used for years. Is it infected? What does it do? Nobody knows, although some of the filenames are intriguing. Fortunately, no matter how interested one might be in a disk's contents, *Virus Bulletin* has a very simple policy: untrusted diskettes are not put in 'clean' machines. No ifs, no exceptions. The way to examine the contents of the unknown diskettes is to write-protect them and cart them off to the quarantine machines used for virus checking and analysis.

Although this cache of mystery diskettes has presented no threat to the security of the *VB* move, one suspects that in many companies, whatever the policy, user inquisitiveness (especially during the chaos of a move) may overcome usual caution. It is very easy to imagine picking up a disk and quickly checking its contents - it is then only a small step to running software stored on it.

Above all else, such a scenario highlights the need to carry out a clean-up of a virus attack properly. This means scanning every diskette, old and new, after a virus outbreak, and doing one's best to ensure that all storage media are checked, not just those disks which 'happen to be around' during the clean-up operation. If this process is not completed thoroughly, the clean-up team will have plenty of opportunities to practice the procedure: the chances are that they will have to repeat the operation again and again until they finally do get it right. An infected diskette left mouldering in a desk drawer is a time bomb, waiting for an unwary user to trigger it.

Cleaning up a large-scale virus outbreak is easier said than done. In order to stand any chance of doing a thorough job, the co-operation of the entire user community is required. This means that it is absolutely vital that users should not be afraid to bring out personal disks which have been used in company machines, as well as disks which they believe may be infected. In a company which adopts the 'hang 'em and flog 'em' approach to computer security and IT use and abuse, carrying out a thorough clean-up operation could well be virtually impossible, as users may be afraid of submitting any non-company diskettes for checking. Should these diskettes be infected (as is likely in a major outbreak), the virus could be introduced unwittingly once the fuss had died down.

The gradual build-up of old diskettes is a continual problem for any IT-intensive organisation. Disks, unlabelled and unscanned, can be found in every office; checking one's own department (or even one's own desk) can be an eye-opening exercise. Such stray disks represent a large percentage of the time taken to check an office for viruses, and wherever possible should be eliminated. With most companies now using networks, there is little need for data to be transported around the office by foot - the network is quicker and cheaper. Users should be encouraged to use diskettes as infrequently as possible, as the more diskettes around, the more to scan... Obvious, but often only noticed when one has to check them all.

“stray disks represent a large percentage of the time taken to check an office for viruses”

NEWS

LZR Virus on Formatted Diskettes

According to reports received from both Finland and Sweden, a large shipment of pre-formatted diskettes has been found to be infected with the LZR virus. The diskettes, apparently imported from China, are unbranded, and have been found principally in the Nordic countries. The Finnish anti-virus company *Data Fellows* has obtained unopened boxes of the diskettes and confirmed that they are infected.

The shipment of 400,000 diskettes contains approximately 20,000 which are infected. Of the shipment, 15% was sent to a Finnish company, *PC Superstore*, for resale.

As soon as the virus was found on the diskettes, *PC Superstore* withdrew them from their shelves. The company has also placed a series of adverts in the largest Finnish newspapers, alerting buyers to the infection of the disks, and offering a special free version of *F-Prot* to anyone who can show proof of purchase of the infected media. Commenting on the virus, *PC SuperStore* Product Manager Ismo Viitamo said: 'We have done the best we can to notify everybody who bought the diskettes. They will be provided with a virus protection program which will detect and erase the virus, and all diskettes will be replaced if necessary. I am not sure whether we will sell pre-formatted disks in the future - we will only do so if we can come up with an extremely reliable method to guarantee their safety.'

Attack of the Data Diddler

LZR is a relatively simple boot-sector virus, which infects the Master Boot Sector (MBS) of the fixed disk, and the boot sector of floppy diskettes. The virus contains no stealth capabilities, and operates in a manner similar to most common MBS-infecting viruses.

When resident, the virus hooks Int 13h, and infects on any read or write to the first two floppy diskettes. On an infected fixed disk, a copy of the original MBS is located at Track 0, Head 0, Sector 2.

The virus has a particularly unpleasant trigger. When the virus intercepts any read or write to the disk, there is a one in 10000h chance (65,536) that the contents of the fixed disk will be overwritten. If this trigger routine is not called, the virus then enters a second trigger, which has a one in 256 chance of executing. This routine XORs a random byte in the read or write buffer with a random value, leading to gradual corruption of the data stored on the disk.

As yet, *Virus Bulletin* has been unable to contact the manufacturer of the disks. There is, however, a disturbing possibility that more than the single 400,000-diskette shipment is infected: it is claimed that one of the disk formatting machines used by the disk's manufacturer contained an infected disk image when it was purchased ■

Virus Prevalence Table - September 1994

Virus	Incidents	(%) Reports
Form	25	33.8%
AntiEXE.A	10	13.5%
Stoned	6	8.1%
Stoned.I	4	5.4%
AntiCMOS	3	4.1%
Parity_Boot	3	4.1%
Stealth2Boot	3	4.1%
NoInt	2	2.7%
One_Half	2	2.7%
SMEG:Pathogen	2	2.7%
Tequila	2	2.7%
Angelina	1	1.4%
Attack_Trojan	1	1.4%
Cascade.1704.G	1	1.4%
Jimi	1	1.4%
Junkie	1	1.4%
Keypress.1216	1	1.4%
NYB	1	1.4%
PrintScreen	1	1.4%
Quox	1	1.4%
Trackswap	1	1.4%
V-Sign	1	1.4%
Yankee.2C	1	1.4%
Total	74	100%

Virus Mutation Toolkit?

Among the binaries which showed up in the Technical Editor's E-mail in the last month was a collection of viruses, which seems to have been written by making slight changes to existing viruses using a virus-mutating tool.

According to sources in the virus 'underground', several such programs are now under development. The most effective of these is reported to be able to create 2000 new viruses per hour.

Although quite a few of the 177 viruses in the collection did not work properly, they may be just the first sign of what to expect in the near future.

The development of such a tool would be genuine cause for concern among the anti-virus software community. Every mutated virus, even one which did not appear to work, would have to be analysed individually; the cost of this process would doubtless be passed on to the end-user, either as an increased price or poorer scanner performance ■

IBM PC VIRUSES (UPDATE)

The following is a list of updates and amendments to the *Virus Bulletin Table of Known IBM PC Viruses* as of 21 October 1994. Each entry consists of the virus name, its aliases (if any) and the virus type. This is followed by a short description (if available) and a 24-byte hexadecimal search pattern to detect the presence of the virus with a disk utility or a dedicated scanner which contains a user-updatable pattern library.

Type Codes	
C	Infects COM files
D	Infects DOS Boot Sector (logical sector 0 on disk)
E	Infects EXE files
L	Link virus
M	Infects Master Boot Sector (Track 0, Head 0, Sector 1)
N	Not memory-resident
P	Companion virus
R	Memory-resident after infection

Anti_Pascal-II.407 CN: A minor, unremarkable variant, detected with the Anti-Pascal2 pattern.
Ash.743 CN: There are now ten new minor variants of this virus, which have been named Ash.743.B-K. All are detected with the Ash.743 pattern.

Australian_Parasite: CN, CR, CER: The Australian virus writer who calls himself 'Australian Parasite' has written a large number of viruses. The smallest are non-resident COM infectors. Most of the others are resident COM infectors, although some also infect EXE files. Some are encrypted, but they can all be detected with a simple searchstring. In one case a number of wildcards are necessary.

Austr_Para.118	B440 B176 B6F0 CD21 B800 4233 C933 D2CD 21B4 40B1 76FE C6CD
Austr_Para.122	B440 B17A 51BA 6AFF CD21 B800 4233 C933 D2CD 21B4 4059 BA00
Austr_Para.152	B440 B198 B601 CD21 B800 4233 D233 C9CD 21B4 40B6 01B1 04CD
Austr_Para.153B	B440 B199 B601 CD21 B800 4233 D233 C9CD 21B4 40B6 01B1 04CD
Austr_Para.155	B440 B19B B601 CD21 B800 4233 D233 C9CD 21B4 40B6 01B1 04CD
Austr_Para.187	B440 B9BE 00BA 0001 CD21 B800 4233 D233 C9CD 21B4 40B6 01B1
Austr_Para.215	B440 B9D7 00BA 0001 CD21 B800 4233 D233 C9CD 21B4 40B6 01B1
Austr_Para.217	B440 B1D9 CD21 B800 422B C92B D2CD 21B4 40B1 03B6 01CD 215A
Austr_Para.221	B440 B9DD 00CD 21B8 0042 2BC9 2BD2 CD21 B440 B103 B601 CD21
Austr_Para.229	B440 B9E5 00CD 21B8 0042 2BC9 2BD2 CD21 B440 B103 B601 CD21
Austr_Para.272	B440 B910 01CD 21B8 0042 33D2 33C9 CD21 B440 B903 00BA 0001
Austr_Para.306	B440 B932 01CD 21B8 0042 33D2 33C9 CD21 B440 B904 00BA 5E01
Austr_Para.338	B440 B952 01BA 0001 CD21 B800 4233 C933 D2CD 21B9 0300 BA00
Austr_Para.369	B440 B971 018D 9600 0152 CD21 B800 4233 C933 D2CD 21B4 40B9
Austr_Para.377	B440 B979 01BA 0000 CD21 B800 4233 C933 D2CD 21B9 0300 BA00
Austr_Para.440	BD?? ??B8 ??? 8D9E 1201 B9D6 0131 0743 E2FB
Austr_Para.482	B440 B9E2 01CD 21B8 0042 2BD2 2BC9 CD21 B440 B904 00BA 7901
Austr_Para.588	B440 BA00 01B9 4C02 CD21 B800 4233 D233 C9CD 21B4 40B9 0400
Austr_Para.591	B440 B94F 02BA 0001 CD21 B800 4233 C933 D2CD 210E 0E1F 07BA
Austr_Para.635	B440 B97B 02BA 0001 CD21 B800 4233 C933 D2CD 21B4 40B9 0400
Austr_Para.726	B440 B9D6 02CD 21B8 0042 33D2 33C9 CD21 B440 B903 00BA 0001
Austr_Para.762	B440 B9FA 02BA 0001 CD21 B800 4233 D233 C9CD 21B4 40B9 0400
Austr_Para.784	B440 B910 03BA 0001 CD21 B800 4233 C933 D2CD 21B9 0300 BA00
Austr_Para.1050	7D01 33C9 CD21 8BD8 B440 B9BC 028D 968B 01CD 21B4 3ECD 21C3
Austr_Para.1179	B440 B99B 04BA 0001 CD21 B800 4233 C933 D2CD 21C6 064A 025A
Austr_Para.AMSV	B440 B9BB 01BA 0001 CD21 B800 4233 C933 D2CD 21B4 40B9 0400
Austr_Para.Comic	B440 B92C 04BA 0001 CD21 B800 4233 C933 D2CD 21B4 40B9 0400
Austr_Para.Gotter	B440 B900 04BA 0001 CD21 B800 4233 C933 D2CD 21B4 40B9 0400
Austr_Para.Lipo	B440 B922 01BA 0001 CD21 B800 4233 C933 D2CD 21B4 40B9 0400
Austr_Para.VGA_Demo	7D01 33C9 CD21 8BD8 B440 B907 0E8D 9631 02CD 21B4 3ECD 21C3

Big_Bang CN: This virus has not been fully analysed, but seems to contain destructive code, as well as the text '[Big Bang] (c) 1993 Evil Avatar'.

Big_Bang B95A 018D 9603 01B4 40CD 21B8 0042 2BC9 99CD 21B9 0300 8D96

Black_Jec.247 CN: There are now four new minor variants of this virus, which have been named Black_Jec.247.B-E. They are all detected with the Black_Jec (Bljec) pattern.

- Bloody_Warrior** CER: A 1344-byte encrypted virus which contains a text message claiming it originated in Milan, Italy.
Bloody Warrior E800 005B FA2E 8177 3D?? ??90 0E1F 8177 2A?? ??90 8177 2E??
- Burger.382.C** CN: Detected with the VirDEM pattern.
- Civil_War.245** CN: Yet another variant from the person calling himself 'Dark Helmet'.
Civil_War.245 80E1 2F80 F901 5974 4A51 523E 8B9E F401 B43F B903 008D 96EE
- HLL0.4742** EN: As with other HLL viruses, no search pattern is provided.
- Hungarian.Kiss.1006** CER: Very similar to the 1015-byte variant originally reported as Kiss. Detected with the same pattern.
- Infector.847** CN: There are two variants of this virus, A and B. Both are detected with the pattern given below.
Infector.847 A200 01A0 8E03 2EA2 0101 A08F 032E A202 018C C8A3 3D03 B980
- Iron** CN: A non-remarkable, 271-byte virus, which contains the text 'Iron Butterfly V1.2'
Iron B90F 01B4 40CD 21B8 0042 9933 C9CD 218B 8640 022D 0300 8986
- IVP** CEN: There is one new IVP-generated virus this month: Sonic (CEN, 666).
- JD.158** CR: There are now fifteen new minor variants of this virus, which have been named JD.158.B-P. All are detected with the pattern below. As it will also detect the original variant (now renamed to JD.158.A), the original search pattern should be discarded.
JD.158 8EDB 833D 3D74 08B4 25CD 21B1 9E8E C30E 1FF3 A458 0E07 C3CD
- Jerusalem.1808.Rambo** CER: In addition to changing the self-recognition string to 'Rambo', the author has made several minor changes to this virus, which invalidate existing search strings.
Jer.Rambo 2638 05E0 F98B D783 C203 B800 4B06 1F0E 07BB 3500 061E 5350
- Jerusalem.AntiCad** CER: There are six new viruses in this group, 3012.F and 4096.E-I.
AntiCad.3012.F 33C0 8ED8 A017 041F 240C 3C0C 752E E460 247F 3C53 7526 2E81
AntiCad.4096 (gen) 33C0 8ED8 A017 041F 240C 3C0C 7534 E460 247F 3C53 752C 2EA1
- Lockjaw.887** P: This 887-byte variant contains the text 'KenSON V - Lobo/435 BF! :)'.
Lockjaw.887 9C06 1E50 5352 3D00 4B75 03E8 0E00 5A5B 581F 079D 2EFF 2E77
- Lyceum.958** CER: A Russian 'stealth' virus, which uses the '100-year' trick to mark infected files. It contains the text 'HELLO HACKERS FROM MIREA'. Another similar 930-byte variant has also been found.
Lyceum-gen 3D00 4B74 0F80 FC3D 740A 80FC 4374 0580 FC56 7508 E808 0075
- Murphy.Migram.1221** ER: There are now eleven new minor variants of this virus, which have been named Murphy.Migram.1221.B-L. They are all detected with the HIV pattern.
- Npox.963** CER: There are now ten new minor variants of this virus, which have been named Npox.963.C-L. All are detected with the Npox pattern.
- Sandy.1392** ER: An encrypted virus, containing the text 'sandy beaches. bridges sinking into the sea. beautiful confusion. you're a fading memory'.
Sandy.1392 0050 5A2E 310C 03FE 4650 5A46 8CDA 81FE 5F05 7EEF 505A 505A
- Screen+1.948.O** CEN: Detected with the Screen+1 (948) pattern.
- STSV** CN: The five new minor variants of this virus have been named ST SV.C-G. All are detected with the STSV (previously '200') pattern.
- Sundevil. 762** CR: There is not much to say about this one... or maybe it is just that after the first few thousand, all viruses start to look the same.
Sundevil.762 8B86 FC02 8EC0 33FF 8BF5 B9FA 03F3 A433 C08E D8B8 FD01 A384
- Taiwan.708.C** CN: Detected with the Taiwan pattern.
- Troi.C, Troi.D** CR: Similar to the A and B variants. Detected with the Troi pattern..
- Trivial.Banana** CN: The eleven new minor variants of this virus have been named Trivial.Banana.B-L. All are detected with the pattern below. As it will also detect the original variant (now renamed Trivial.Banana.A), the original search pattern should be discarded.
Trivial.Banana B801 43CD 21B4 4FEB B7C3 2042 414E 414E 412C 2063 6F64 6564
- Vacsina.TP.25.B** CER: A minor 1805 byte variant. Detected with the Vacsina-I pattern.
- VCL:** This month, there are two non-encrypted variants, detected with the pattern published for VCL.VoCo: 535 and Dial.600. There are also six VCL-generated 'companion' viruses: 337, 389, 405, Nomem, Pearl_Harbour.931 and Taboo. Finally, there are four variants which are almost identical to older variants: Code_Zero.654, Donatello.831, Earthday.799 and Kinison.809.

Vienna	CN: Several unremarkable variants have been found, but are not detected with patterns already published. They are: 660, 662, 700.C, Feliz, Gipsy, It.457, Parasite.861, Violator.779 and W-13.318. The Feliz variant contains the text 'Feliz Navidad! Feliz A o Nuevo!' Also, two new encrypted variants are known, 833.B and 1041.
Vienna.660	FC90 8BF2 83C6 0AB9 0400 BF00 01F3 A48B F206 B42F CD21 9089
Vienna.662	8BD7 2BF9 83C7 0205 0301 03C1 8905 8BFA B440 2BD1 B996 02CD
Vienna.700.C	FC90 8BF2 83C6 0A90 B904 00BF 0001 F3A4 8BF2 06B4 2FCD 2190
Vienna.833.B	5153 50BE ???? 2E8A 44FF 8BDE 81EB 5102 B98B 012E 3007 43E2
Vienna.1041	FC52 5E83 C60D 90B9 0001 515F B903 0057 F3A4 5F52 5EE8 1600
Vienna.Feliz	5D81 ED48 018D B646 03BF 0001 B903 00FC F3A4 06B4 2FCD 2189
Vienna.Gipsy	FCB9 0300 BF00 01F3 A48B FABA 1200 03D7 B41A CD21 32DB 83EA
Vienna.It.457	5D81 ED30 018D B6F1 02BF 0001 B903 00FC F3A4 06B4 2FCD 2189
Parasite.861	FC8B F283 C62A EB14 BA8B 0003 D6B4 1ACD 2106 568E 062C 00BF
Violator.779	FC8B F283 C668 BF00 01B9 0300 F3A4 8BF2 B80F FFCF 213D 0101
Vienna.W-13.318	2BF9 0504 0103 C189 05B9 3E01 905F 8BD7 81EA 3401 B440 CD21
Vienna.BNB	CN: The eight new minor variants of this virus have been named Vienna.BNB.C-J. All are detected with the pattern below. As this pattern will also detect the A and B variants, the original search pattern should be discarded.
Vienna.BNB	F3A4 8BF2 B824 35CD 2106 53B8 2425 BAB6 0003 D6CD 211E 0706
Vienna.561.B	CN: This non-remarkable variant is detected with the Vienna-2, Ghostballs and Vienna.1239 patterns. Those patterns all detect a number of Vienna variants, and should not be relied on for identification.
Vienna.709	CN: Detected with the Dr. Q pattern.
Vienna.680	CN: Detected with the Violator pattern.
Vienna.Black_Ice	CN: A 742-byte virus. Detected with the Violator pattern.
Vienna.Choinka.C	CN: Detected with the Vienna-4 pattern, just like the .A and .B variants.
Vienna.W-13.534.K	CN: Detected with the W-13 pattern.
Vienna.W-13.534.L	CN: Detected with the W-13 pattern.
Vienna.W-13.539	CN: Detected with the W-13 pattern.
Virdem.1336.Locked.B	CN: Detected with the Virdem pattern.
VLamiX	ER: This 1090-byte virus was distributed in a fake ARJ 3.0 package to BBS systems worldwide. Quite a few infections have been reported.
VLamiX	061E 8CC8 8ED8 BF28 00A1 5004 3105 83C7 02BA 5004 3BFA 72F4
XPH.1032	CER: Detected with the XPH.1029 pattern.
Yam.3596	CR: Detected with the Yam.3599 pattern.
Yankee_Doodle.2167	CER: Possibly related to the 'Login' group, but requiring further analysis.
Yankee_Doodle.2167	7503 0C01 C3F6 C208 75F8 80FE 0377 F332 C0C3 FC5B 81EB 2B00
YB	CN: Two new variants from the virus author known as Köhntark.
YB.425	B802 3DCD 2193 B905 008D 9475 01B4 3FCD 2172 218B 8498 0105
YB.426	B802 3DCD 2193 B905 008D 9476 01B4 3FCD 2172 218B 8499 0105
YB.2277	CN: This variant is much larger than the other YB viruses. Just like the previous virus, it contains the text 'YB-2 / Khöntark'.
YB.2277	B802 3D9C FF9C 6601 72E3 93B9 0500 8D94 5D01 B43F 9CFF 9C66
ZigZag.127	CEN: A 127-byte overwriting virus that contains the text '*ZZ* v 1.0'.
ZigZag.127	AACD 20B8 023D BA9E 00CD 2193 B43F B902 00BA 6D01 CD21 813E
Zombie	CR: 747 bytes, contains the text 'Zombie - Danish woodoo hackers (14AUG91)'.
Zombie	9C3D 004B 740F 3D69 4B74 069D 2EFF 2E84 008B D89D CF2E C706
Zulu	CR: 1390 bytes, contains the text 'ZULU-GULA by Dr Mengele and Rudolf Hess'.
Zulu	9C3D 004B 7403 EB53 902E 8C16 1A01 2E89 261C 010E 17BC 0302
ZX-X	ER: 600 bytes, contains the text 'ZX-X'.
ZX-X	9C80 FC3D 7412 80FC 4374 0D80 FC56 7408 3D00 4B74 03E9 3E01

INSIGHT

Auerbach: Viruses Et Cetera

Megan Palfrey

The man: Tjark Auerbach, German anti-virus researcher and software developer. The company: *H+BEDV*, little known outside Germany, but highly regarded and holding a considerable proportion of the German market. The product: *AVScan*, a consistent high-performer in *VB* comparative reviews. The series of *VB* interviews continues with an insight into the man, the work, and the product.

How it Began

Computing was not a real passion for Auerbach until after he left school. He was 20 years old, training to be a technical assistant in electronics, when he first came to grips with a *Commodore PET-2001*, which he describes as 'a lovely machine'. On completing that course, he went to a technical college to further his studies: 'I gave it up,' he said, 'after too many long nights in the computer room. One day I woke up and asked myself, "Is this really what you want to do?" It was not. So, I got a job at a computer company.' The date? 1984: the *IBM-PC* was just starting to appear in Germany.

At that company, where he spent four years, Auerbach repaired and assembled PCs; his programming experience did not begin until the 80x86 machines appeared. He then returned to college to complete his interrupted training as a Government-approved technician. During this period, he founded *H+BEDV*, the company which was to become one of Germany's best-known names in anti-virus software.

Virus Alert

H+BEDV did not begin as anti-virus software specialists, but as software importers, dealing in such programs as *386MAX*, *Super PC Quick*, and *PC Tools*. This was a route which helped Auerbach gain expertise with various types of soft- and hardware: in offering customers technical support, he could see the problems they were having, and learned more about users' demands and requirements.

Auerbach's first exposure to viruses was an accident: like many other users, his own system became infected. It was 1987; the virus was Jerusalem. He was directed to a 'friend of a friend' for help, and between the two, the first *AntiVir* program was born. Although not very well known elsewhere, it has been a huge German success, and is currently undergoing its fifth major revision.

Virus Authors

Auerbach has never written, nor been tempted to write, a virus - 'But I get very itchy fingers when I see the rubbish that some of the virus writers push out!' he commented. One

of his 'ambitions' is to find a good virus writer: 'I love well-written viruses; it's incredible to see a virus which works. Bad viruses waste my time. If I get inside a virus and find out that it doesn't infect, I have to spend a long time disassembling it to find out why.'

Fortunately, there are few 'good' virus authors. Even polymorphics pose only a limited threat to dedicated researchers: 'It took about two weeks to master the first MTe virus I encountered. When we met a TPE virus, it took us three days; now, an ordinary polymorphic will take about a day. They are no big deal.' Whoever programmed SMEG, in Auerbach's opinion, belongs to this category: he could have made the viruses generated more difficult to detect merely by putting more randomness in his further instructions.

*"the concessions... cannot
compensate for the benefits of
having a resident scanner"*

The real threat, for Auerbach, is the 'two-legged' virus. Forgetting backups, formatting the hard disk - often the person sitting in front of the computer creates the greatest problems. Fear, he asserted, has the potential to make a catastrophe out of a minor incident: it is time to make the developer's approach to the user more accessible.

Developmental Elements

His role in *H+BEDV* is now less of a researcher, more of a Quality Assurance controller: it is he who tries out new viruses as they come in; he who liaises with customers who have virus problems; he who ensures that their problems are solved with a maximum of expertise and a minimum of fuss.

'Our usual turnaround time, from the moment someone logs on to the company's mailbox to the time when analysis is complete and a solution returned to the customer, is two to four hours. This doesn't always work, but we do our best.'

The first step in this process is for Auerbach to try to make the virus replicate: if it does, it goes to one of his programmers. When it returns, Auerbach attempts to make it replicate onto 'real-world' files, files which would be on every user's machine: *COMMAND.COM*, *WIN.COM*, *DISKCOPY*, etc. Repairs are usually successful: 'Touch wood, we have only had one false repair; last year, with an incorrectly repaired Tremor virus.'

Outward Bound

H+BEDV's product is marketed only in Germany at present, but the company is already at work on a bilingual version, which is planned for the next major revision. There are some



Auerbach believes that anti-virus software must become more user-friendly: 'Have we seen the past only through our own glasses? Were we wearing our hats and not those of users?'

problems, he conceded, although none of them are insurmountable. Mostly, they concern the manual: 'It's very dry material, but I want my customers to be able to relate to the company and the product, so I've put my own personal touch in it; footnotes where I write down my impressions. I'm not sure it would be possible to translate those - I'm very anxious about it.'

The next major revision of *H+BEDV*'s product is due for March 1995, to coincide with the German *CeBit* exhibition, which is attended by people from all over the world.

The company plans to release an NLM by the end of 1994: like the stand-alone product, it will at first only be available in German, but it is being developed with an English-language module as well.

A Heuristic Future...

Auerbach, like many other anti-virus software developers, is exploring heuristics: he sees virus-non-specific detection as the road to the future. Indeed, *AVScan* already detects polymorphic viruses using generic detection. He has not, as yet, expended a great deal of effort to develop the heuristic side; this is planned for next year's major revision.

'Everything done by software,' says Auerbach, 'can be undone by software. From a developer's standpoint, however, you have to draw borders; you must stop emulating the virus sometime. Virus writers already defeat some emulation engines. Emulating and heuristics will stay around; they are part of the future.'

He is also concerned that developers have been, until now, concerned primarily with their own aims: 'Have we seen the past only through our own glasses? Were we wearing our hats and not those of users?' Anti-virus software in general, he opined, must become more user-friendly, despite researchers who prefer mile-long command lines for each task.

In common with some other developers, Auerbach has reservations about TSRs, believing that the concessions one must make in terms of memory utilisation cannot compen-

sate for the benefits of having a resident scanner. However, having seen *Sophos*' *InterCheck* (see p.18), he admits that TSRs can have a place in virus prevention.

H+BEDV has started work on a product which, though not memory-resident, will automatically scan every diskette inserted into the PC for boot sector viruses - these comprise about 85% of all viruses known in the wild. It will at first be virus-specific, but the generic element will eventually become more prominent. The company is also currently developing a program with a memory-resident component: 'Well, it can be effective in the right place,' said Auerbach.

Outside Viruses

The company also develops programs outside the anti-virus arena - amongst other projects, Auerbach has recently developed a card to re-boot a server: 'When you develop software, it is not rare for your server to crash. If this happens to me at home, I have to run down three floors, switch the server off, and switch it back on again. This little card is accompanied by an engine which re-triggers the server; so, if your server says "Guten Abend", it will re-boot fifteen seconds later.'

The company plans to start marketing this product this year - though their main task will remain anti-virus research, they intend to continue development in other areas.

Increasing and Augmenting

Despite the fact that *H+BEDV* is going from strength to strength, a corporate decision has been taken to limit growth to 20% per annum. Auerbach has a strongly paternalistic streak regarding the company: 'I like to greet everyone by name when I come in, to take an interest in each employee, to know when they're ill so I can send them home. If it's too hot, I like to say, "Come on, let's all go swimming!" at lunchtime, or to know that they'll just get up and go. You can't do these things in a larger company.'

He regards himself as 'paterfamilias' to his staff, an outlook which proved beneficial during the Michelangelo 'crisis': for some six weeks, work started at 7 am, finishing often as late as 8 pm, with business as usual on Saturdays, and even two Sundays: 'No-one asked about working overtime; they just did it,' reminisced Auerbach. 'So after it was over, I took them all to *Eurodisney* and Paris for three days.'

'This is the way I like it,' he said, though ruefully noting that his marketing manager had a slightly different outlook. 'I like to keep my employees happy: if they are happy, they do good work, and that's good for our customers.'

Life for this man is his family and his company, and he readily agrees that the one often merges with the other: 'I am a contented man - like the people around me, I enjoy what I do. At *H+BEDV*, we work together towards common goals: this is, has been, and will be, our aim. As long as we share these same aspirations, our success, I hope, will continue.'

VIRUS ANALYSIS 1

3APA3A: The IO.SYS Hunter

Eugene Kaspersky
KAMI Associates

MS-DOS has been a prime target of computer hackers for more than ten years now. They crack, hack and beat it, find new ways to infect it, and refine their infection algorithms. With all this effort, it is unsurprising that new types of viruses surface regularly, sometimes startling researchers with unusual characteristics. 3APA3A (pronounced Zaraza, meaning 'infection' in Russian), is just such a virus.

Overview

This virus was discovered in Moscow in mid-October 1994. Its encrypted text is in Russian, but anglicised so that it can be displayed using standard DOS display drivers. It is 1024 bytes long, with two 512-byte parts (sectors), the first of which contains the virus' installation code and the floppy disk infection routine. On a floppy, the boot sector is overwritten by virus code, and the original boot sector and another 512 bytes of virus code are stored in the last sectors of the floppy disk root directory.

The second part of the virus is code placed into the floppy disk boot sector, and holds the hard drive infection routine. On hard drive infection, the virus does not change either the Master Boot Sector (MBS) or the DOS Boot Sector, but replaces IO.SYS or its equivalent (e.g. IBMBIO.COM).

Thus, both the boot sector of floppies and the hard drive's IO.SYS file will be infected. Nothing else is affected, but this alone is sufficient to allow the virus to spread quickly.

Loading from an Infected Floppy

On loading from an infected floppy, the virus decrypts itself and passes control to the hard drive infection routine. This routine reads the first boot sector of the fixed disk (usually the C: drive) and checks its size. If it is less than 16MB (i.e. the disk has a 12-bit File Allocation Table), it will not be infected. The virus then calculates the address of the first sector of the root directory, reads it, and checks the attribute of the first entry in the root directory.

On a 'normal' DOS drive, the first entry in the root directory is the file IO.SYS. This is loaded at boot time by the operating system. If this file does not contain the VOLUME label, 3APA3A treats the file as if it were an uninfected copy of the IO.SYS file.

The virus copies the contents of the IO.SYS file into the last clusters of drive C, then moves each root directory entry, from the third to the 77th, down. The last entry is overwritten by the previous one, and the file occupying that space

will be lost. The virus then copies system data from the first entry to the third (date/time stamp, size, and pointer to first cluster of IO.SYS). This is shown below:

Root directory before moving	Root directory after moving
File1	File1
File2	File2
File3	IO.SYS
File4	File3
File5	File4
...	File5
	...

The virus then changes the third entry in the root directory to point to the first cluster of the *copy* of IO.SYS. Thus, there are two copies of IO.SYS in the system: the first root directory entry, which points to the original IO.SYS, and the third root directory entry, which points to the copy of IO.SYS created by the virus. 3APA3A then overwrites the original IO.SYS with its own code and sets the VOLUME attribute on this directory entry. This is shown below. Note that in (2), the first copy of IO.SYS contains the virus code, and is loaded at boot-time.

(1) Root directory before infection

		size	cluster	attributes
IO	SYS	40470	2	Arc R/O Sys Hid
MSDOS	SYS	38138	22	Arc R/O Sys Hid
COMMAND	COM	52928	41	Arc
DOS		0	67	DIR
AUTOEXEC	BAT	100	68	Arc
CONFIG	SYS	150	69	Arc

(2) Root directory after infection

		size	cluster	attributes
IO	SYS	40470	2	Arc R/O Sys Hid Vol
MSDOS	SYS	38138	22	Arc R/O Sys Hid
IO	SYS	40470	16108	Arc
COMMAND	COM	52928	41	Arc
DOS		0	67	DIR
AUTOEXEC	BAT	100	68	Arc
CONFIG	SYS	150	69	Arc

The virus uses only Int 13h calls and manipulates FAT sectors as well as IO.SYS sectors. It checks the FAT for free space before saving the original IO.SYS, and corrects all FAT copies. The virus also stores the absolute address (in Int 13h format) of the saved IO.SYS, as well as the address of the overwritten IO.SYS, with the virus copy.

The VOLUME attribute of the infected IO.SYS is the virus identifier (the virus does not infect the C: drive if the first entry has it set), as well as the virus' stealth feature: normal calls to DOS will not detect the presence of files marked as the volume label, as the operating system does not expect any such file to exist. Thus, without even being active in memory, 3APA3A can avoid detection by most checksummers (though this is not a difficult situation to change).

Loading from an Infected Hard Drive

On loading from the hard drive, the ROM BIOS loads and executes the MBS, which in turn loads the DBS. The procedure is identical to loading from a clean disk, until the boot sector code searches and executes the system disk files.

The standard loader searches for system files using their names; IO.SYS and MSDOS.SYS (or equivalents). It does not check file attributes and loads IO.SYS into memory even if the corresponding directory entry has a VOLUME label. Then, the system loader reads and executes the file to which the first directory entry points, i.e. it executes the virus. The virus installs itself into memory, reads the original IO.SYS (using the absolute addresses stored on infection) and passes control to it.

Floppy Disk Infection

On loading from an infected disk, control is passed to the installation routine, which is uninteresting: the virus stores the Int 13h address, decreases the size of system memory, copies itself into system memory, hooks Int 13h and passes control to the original system loader. On Int 13h calls, the virus checks the number of the accessed disk: if it is a floppy, infection commences.

On infection, the virus reads the boot sector, checks for its ID-byte (by comparing the byte at offset 21h with 2Eh), saves the original boot sector and first part (installer) of the virus to the last sectors of root directory (the tenth and eleventh logical sectors on a 360K floppy disk) and overwrites the boot sector with the virus' second part, the hard drive infection routine.

When overwriting the boot sector, the virus uses a primitive polymorphic algorithm to encrypt itself. Four different decryption commands are used (NOT, XOR, ADD and SUB): these use several different register variants and pad out the decryption routine with junk code.

Trigger Routine and Bugs

The virus checks the system date on loading from an infected disk. If it is August, the following message is decrypted and displayed in standard ASCII:

```
B BOOT CEKTOPE - 3APA3A
```

This means: 'There is an infection in the boot sector' ('V Boot sectore - zaraza').

The virus contains bugs, one of which is common in PC/XT viruses. In order to speed up code execution, Intel processors load in the next instruction while the current one is executing. This is known as the 'Pre-fetch Queue'. In the virus' decryption routine, an instruction which is already in the pre-fetch queue is decrypted and executed. This results in the *encrypted* instruction being executed on the i286-i486 machines. On XT and Pentium machines, the virus code functions correctly.

Detection and Disinfection

3APA3A is not, according to standard definition, a stealth virus, and does not substitute infected sectors for their originals on access. Detection and disinfection is trivial on floppies, but difficult when the hard drive is affected. It is impossible to access the infected IO.SYS by standard DOS utilities, to delete or rename it, or to change its name with the LABEL command. On entering LABEL, DOS reports the disk name as IO.SYS. If an attempt is made to change LABEL, DOS reports:

```
Cannot make directory entry
```

Drive C cannot be disinfected with the SYS command, as this will replace only the second copy of IO.SYS, not the virus. Moreover, the disk will not be bootable because the virus uses fixed addresses to load the original IO.SYS: these will be incorrect after the SYS command has been used. The only secure way to detect and disinfect this virus is to update scanners with a routine which checks files through absolute access (via Int 13h or Int 25h).

3APA3A	
Aliases:	Zaraza.
Type:	Memory-resident, and polymorphic. Infects boot sectors of floppy disks and IO.SYS (or equivalent) of C: drive.
Self-recognition on Hard Disk:	Checks the first root directory entry for VOLUME attribute.
Self-recognition on Floppy Disk:	Compares byte at offset 21h with the value 2Eh.
Self-recognition in Memory:	Does not check for itself in memory.
Hex Pattern on File:	0EE8 0000 5E83 EE04 5650 5351 521E 06B4 04CD 1A80 FE08 7512 No search pattern possible in sectors.
Intercepts:	Int 13h for infection.
Trigger:	When loading from an infected drive, during August of any year, a message is displayed (see text). Deletes the 77th entry in the root directory on infection.
Removal:	Removal of the virus is difficult, due to the unusual way in which infection takes place. Use of specialist software is recommended. Alternatively, use a sector editor to reverse the changes made by the virus.

VIRUS ANALYSIS 2

The Peanut Vendor

*Benjamin Sidle
Sophos Plc*

Peanut (a name for which there is no apparent reason), is a small but highly code-efficient multipartite virus. It has now been reported at least once in the wild in Manchester, England. This creature is a pure virus in that it exists only to propagate, and is capable of infecting the Master Boot Sector (MBS) of a hard disk, the boot sectors of floppy disks, and COM files, with only 444 bytes of code.

How Infection Begins

If a PC is booted from an infected floppy, the virus will check whether or not the MBS is already infected; if not, it will copy the MBS to sector 2, head 0, track 0 and then copy itself to the MBS.

It is at this point that Peanut plays its first trick: it does not run the original floppy boot sector like most boot sector viruses (it cannot, as the original boot sector on the floppy was not stored by Peanut). Instead, it loads the PC's original MBS of the fixed disk, making it look as if the PC was booted from the hard disk. The user may never even realise that he has left a diskette in the A: drive.

This trick saves the virus a lot of overhead in having to calculate where it would have stored the original boot sector and thus also helps to keep its size small. By this stage the virus has also installed its own Int 13h handler.

The Infection Process

When the user runs an EXE file, the virus plays its second trick. The file is loaded by reading sectors from a disk: if a sector begins with MZ, the marker indicating an EXE file, the virus installs its own Int 21h handler and remaps the original Int 21h to Int B9h.

The primary purpose of the virus' Int 21h function is to intercept function 4Bh, load and execute a file. Should the file begin with an M, it will not be targeted for infection and processing will continue normally. This prevents the virus trying to infect EXE files, although any such files which begin with the equally valid identifier 'ZM' will be infected and corrupted.

If the file does not start with an M, Peanut assumes it is dealing with a COM file. It then patches the beginning of the file with an M (this translates to the machine code PUSH BP, which has no detrimental effect on the running of the program) so as to mark it as infected, and then a jump to the end of the file. The first four bytes of the original COM file are stored for patching back later.

The virus then appends the rest of its code to the end of the file. This infection process also preserves the time and date of the host file and intercepts Int 24h, the critical error handler, for the duration of the infection process.

When an infected COM file is subsequently executed, it will attempt to infect the MBS of the hard disk. No interrupts are intercepted by the running of an infected COM file; this only happens when the boot sector or MBS version is run.

All floppy reads are intercepted by the virus' Int 13h handler. Regardless of whether or not the floppy is infected, its boot sector is overwritten by the virus. If the disk is write-protected, the critical error is not displayed, and the user will not be aware that there is anything amiss. As if all this were not enough, Peanut also has stealth characteristics: all reads to the MBS are intercepted and the original MBS returned; any writes to the MBS are thrown away.

Conclusion

As has already been indicated at the beginning, the virus carries no payload; its only function is replication. Although one would never wish to glorify the writing of viruses, it must be said that the virus is well written and well thought out. It's a pity that the author does not turn his/her hand to something more useful and profitable.

Peanut	
Aliases:	None known.
Type:	Multipartite boot sector virus with stealth capabilities.
Infection:	MBS of hard disk, boot sector of floppies, COM files.
Self-recognition on Disk:	First byte E8h (MBS) on hard disk; no self-recognition on floppies.
Self-recognition in Files:	File begins with the letter 'M'.
Hex Pattern:	2681 3F4D 5A75 2139 06E6 0275 1B8C C887 0686 00A3 E602 2EA3
Intercepts:	Int 13h, and Int 21h. Int 24h during infection of files.
Trigger:	None.
Removal:	Under clean system conditions, use the FDISK /MBR command. Infected files should be identified and replaced.

TUTORIAL

Virus Infection Techniques: Part 1

The simple boot sector and parasitic file infectors which once made up the sum total of all virus infection techniques on the IBM PC have now been outrun by the advances of technology. This article is the first in a series designed to give the reader an overview of infection mechanisms currently in use by PC viruses, and to discuss their implications in the context of detection and removal. The series is not aimed at virus experts - it is in response to public demand to collect information from several editions of *VB* to form a source of reference.

Overwriting File Viruses

Of all currently known virus types, the overwriting viruses use the most simple infection technique. When such a virus infects an executable file, the start of that file is overwritten with virus code. This means that the host file can no longer function correctly, as the beginning of its own code has been destroyed.

After an overwriting virus has finished executing, it will normally return control to the last DOS function which was accessed, sometimes displaying an error message. The obvious behaviour of these viruses makes it unlikely that they would spread widely without being quickly detected.

Overwriting viruses can be either memory-resident or non-memory-resident, but most are non-memory-resident. Typical examples are members of the Tiny virus family, which infect any executable, regardless of its structure.

The infection technique used by overwriting viruses makes them easy to detect by anti-virus software. However, it is usually impossible to disinfect files, as the start of the file has been destroyed. Some generic disinfection programs which store critical parts of protected files can circumvent this problem, as disinfection involves simply replacing the previously-stored file header.

Unfortunately, virus authors have realised that it is not necessary for an overwriting virus to overwrite the start of the host file: the virus code may be added at any point. In most cases, virus code which is not at the start of the file will never get executed, or will have control passed to its middle, leading to unreliable operation. However, on some files, this technique will work. This forces scanners and integrity checkers to examine the entire contents of a file, something which many developers refuse to do in default mode, as it would slow the scanner down.

Simple Boot Sector Viruses

The type of virus most frequently encountered in the wild is the boot sector virus. This, like the overwriting virus, uses infection techniques which are simple to understand.

When a PC is booted, execution starts at a fixed location in the ROM of the machine. This routine is known as Power On Self Test (POST): its main function is to test the integrity of system memory. It then initialises the hardware and collects information stored in the CMOS so that the time, date, passwords and peripheral settings can be incorporated into the configuration. The last operation of the POST routine is to set up the BIOS addresses in the interrupt vector table, so that software can communicate correctly with IO devices.

The next part of the boot sequence attempts to read from the A: drive (on some machines, the boot process can be configured in the CMOS to prevent this). If a diskette is present in the drive, its first sector is read into memory, and then executed.

If there is no diskette in the drive, a Disk I/O Error occurs. The machine then attempts to read the first physical sector of the fixed disk. This sector, known as the Master Boot Record (MBR) or Master Boot Sector (MBS), searches for a pointer to the Active Partition Boot Sector, which (on a computer set up to boot from *MS-DOS*) contains the DOS Boot Sector (DBS). Finally, the DBS loads the *MS-DOS* operating system into memory and passes control to it.

Replication and Infection

In the boot sequence, there are three obvious executable items which can be infected: the MBS and Active Partition Boot Sector (usually the DBS) on the fixed disk, and the boot sector (sector 0) of the floppy disk.

When a machine is booted from an infected diskette, the virus code in the boot sector is loaded into memory and executed. This code then copies itself to the MBS or DBS of the fixed disk, usually storing a copy of the original boot sector elsewhere on the disk. When the infection process is complete, the virus loads the original floppy disk boot sector and passes control to it. The user, therefore, will not notice any difference in the boot process.

All DOS-formatted diskettes have a valid boot sector, though not all diskettes are bootable. A non-system disk contains a short program in the boot sector which displays the familiar 'Non-system disk or disk error' message. Thus, **a boot sector virus can be carried on both system and non-system diskettes.**

When a machine with an infected fixed disk is booted, the virus code in the MBS or DBS is loaded into memory and executed. Once active in memory, the virus loads any remaining sectors which make up its code, and loads the original boot sector into memory.

Thus, an infected PC will appear to boot correctly, although the virus will be active. Depending on the virus' structure, it is possible for the virus code to monitor every disk read and write. Whenever an uninfected diskette is detected, infection occurs, allowing the virus to spread.

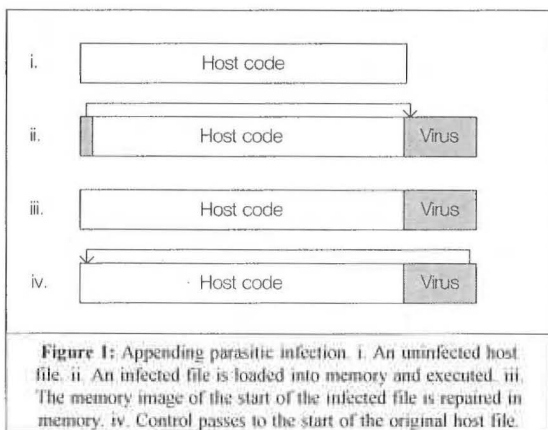
This discussion has been somewhat simplified; many boot sector viruses work in different ways. However, the infection algorithm generally remains consistent: the original boot sector is replaced by virus code which becomes resident on the host machine.

Boot sector viruses are not usually difficult to detect, as long as the virus is not active in memory when the software is run. For a simple boot sector virus, disinfection is trivial, as the original boot sector is usually stored on the disk.

Some boot sector viruses complicate matters by altering the disk partition table: if the virus is not active in memory (as with Empire.Monkey), the fixed disk becomes inaccessible. Others encrypt or alter the original boot sector, so that it cannot simply be copied back into place. Finally, some viruses do not store the original boot sector at all, but load the operating system themselves.

Appending Parasitic File Infectors

The most common infection technique employed by virus writers is to append code to an executable file and ensure that control is passed to it when the file is run. This process is easy to carry out, due to the simple way in which the operating system processes executable files. DOS handles two principal types of executable files, the COM file and the EXE file. Although these are denoted by file extension, DOS determines how they should be treated by considering



their structure. COM files are simply copied into memory at a fixed offset (0100h), and the control is then passed to the start of the memory image, with registers set to default values. The maximum length of a COM file cannot exceed 65,536 bytes.

EXE files are more complicated, and, unlike COM files, their size is limited only by the computer's memory. The entry point of an EXE file is not fixed, but depends on the values stored in the file's header. DOS requires EXE files to start with the hex word 4D5Ah (ASCII MZ) or 5A4Dh (ASCII MZ). If this marker is not present, the file is treated as a COM file, regardless of its extension.

Detection and Detection Evasion

It is very simple to design an infection algorithm for both of these file types. In the case of COM files, the main body of the virus code is copied to the end of the host file, and a short stub of code added to its beginning: this will pass control to the virus code. Once that code has been executed, the memory image of the COM file is restored and control passed to it. Thus, unless the virus code displays its presence in some way, there will be no observable difference when an infected file is executed.

EXE file infection can be carried out in a similar manner: code is appended to the file end, and the entry point of the file stored in the EXE header is altered to point to the virus code (e.g. as done by Frodo.Frodo.A and Peach). The appearance of a typical file before and after infection is shown in Figure 1.

Simple parasitic appending file infectors are easily detected by checksummers and scanners, but virus authors have attempted to complicate matters by adding self-modifying code to the virus. The technique, which is known as polymorphism, is discussed in *Virus Bulletin*, November 1993, page 13.

Another way to make an appending file virus more difficult to detect is to make the transfer of control to the virus code significantly more complex. A good example of this technique is One_Half: the main body of this virus is reached not by a single jump or a change in the EXE header, but by a number of small segments of code which are interspersed throughout the entire file.

There are many ways in which a virus can avoid detection by a checksummer. For a more detailed discussion, the reader is referred to Vesselin Bontchev's paper 'Possible Virus Attacks Against Integrity Programs and How to Prevent Them' (*VB Conference Proceedings 1992*).

Summary

The first article in this series has outlined the primary infection techniques used by computer viruses. Next month, we look at less frequently encountered techniques such as link viruses and companion viruses.

FEATURE

IT Security Breaches: The 1994 NCC Survey

Chris Hook,
NCC Services Limited

Earlier this year, the *National Computing Centre (NCC)*, with the support of the *DTI* and the *JCL*, carried out its second UK survey of IT security breaches and failures. Its aims were to provide information on a range of IT security issues, including attitudes to IT security, the incidence of security breaches, and case study examples.

The Participants

Responses were received from 832 organisations of all sizes, and from most industry sectors. Almost 60% were from organisations employing under 500 staff (see Table 1 below), an encouraging indication that smaller companies are beginning to take the problems of IT security seriously.

Over 80% of respondents reported at least one significant breach over the past two years. The most prevalent incidents reported were equipment failure (47%), power failure (47%), viruses (34%), network failure (31%) and theft (29%). In this article, I shall concentrate on those results affecting the logical security of PCs; in particular, those relating to viruses.

In the two years since the last survey, reported virus infections have increased by 250%. Two hundred and seventy-nine respondents (34%) reported a total of 1,029 incidents: in our first survey (published in 1992), 142 respondents (16%) reported 410 incidents.

Part of the reason for this increase may be, one hopes, due to a greater awareness of the disruption a virus infection can cause, although whether this increased awareness has been translated into positive action to prevent incidents from occurring is perhaps debatable.

No of employees	No of respondents	% of total
Under 100	197	23.7
100 to 499	301	36.2
500 to 999	114	13.7
1000 to 4999	151	18.1
5000 to 10000	24	2.9
Over 10000	35	4.2
Not known	10	1.2

Table 1: Responses by size of organisation. As can be seen, most respondents' companies have fewer than 500 employees.

Policy Implementation

An analysis of the survey showed that formal security policy for IT systems was being implemented by 57% of those who responded, and 51% had formal procedures for PC security. A further 37% had ad hoc PC security procedures.

There were notable variations within different industry sectors: two-thirds of the Finance sector had formal procedures for PC security, compared to only 38% of the Education and Research establishments, although 54% of these claimed to have ad hoc procedures.

Similarly, 66% of organisations employing over 10,000 staff had formal PC policies, whilst under 50% of those employing less than 500 had such measures in place. Controls and procedures implemented by various companies are described in the table below.

Control	Automated Procedure	Formal Instruction	Guidelines & training	Monitoring	Any of These
Backup	36%	35%	0%	16%	94%
Anti-Virus	39%	39%	27%	23%	88%
Authorised Software	13%	59%	24%	24%	91%

Table 2: Controls implemented for PC users (percentage of respondents).

The majority of respondents addressed the areas of data backup, anti-virus procedures and installation of only authorised software in some way, though few of them monitored compliance with the procedures.

Even in the very largest companies (those employing over 10,000 staff), the figures for those monitoring compliance with procedures for backup, anti-virus and installation of authorised software were surprisingly low; only 26%, 31% and 40% respectively.

Respondents were asked to indicate which types of incident, in their opinion, presented the greatest threats to their IT systems. Alongside equipment failure, virus infections were rated by all classes of respondent to be the most notable threat, with 31% rating them as a major threat and 53% a minor threat. This view was particularly prevalent amongst larger organisations. Despite this, only a minority of respondents (including the largest) had issued formal guidelines on virus protection, or included anti-virus procedures in their staff training.

It would seem that whilst there is an awareness within organisations of the threat from virus infections, many have failed to take sufficient steps to counter it (the 'It won't

happen to me' syndrome). Ultimately, over one-third of respondents had suffered disruption to their systems due, in many instances, to repeated virus infections (average nearly four per respondent).

Virus Incidents

Further information was given by some respondents about 109 viruses, two-thirds of which occurred during 1993, often at a cost of many thousands of pounds per incident. Networked PCs were affected in 52% of incidents and standalone PCs in 64%. Personal systems were mainly affected in 62% of incidents. Fifty percent of departmental systems were also affected, in comparison with only 9% of corporate systems.

In half of the cases, more than one day was required to recover all facilities fully, with 14% taking more than a week. The majority of respondents' overall assessment of the impact was that it was minimal or easily absorbed, but in 11% of incidents it was considered significant.

In a number of cases, it was found that backup copies would not restore when needed, or were not up to date. For the greater part, investigation costs greatly exceeded the cost of actual damage caused: this is one reason why every virus, no matter how innocuous its payload, must be treated seriously.

Cost	Major	Minor	Not significant
Investigation/checking	22%	70%	8%
Long-term remedies	11%	40%	49%
Reconstruct software	4%	52%	44%
Loss of business	4%	11%	85%
Reconstruct data	2%	42%	56%

Table 3: Major and minor costs arising from virus infections (percentage of incidents).

Only 20% of respondents who reported a virus infection in detail had costed the incident, but of those, investigation costs of between £10,000 and £50,000 per incident were not uncommon. The highest reported cost was £100,000 to investigate and remove a virus which had affected 200 PCs on three networks. The source was an infected anti-virus software disk (presumably not write-protected)!

Where Infections Originate

The sources of infection were many and varied, and by no means confined to end-users loading illicit software from their own disks, although this was common. We found that viruses infiltrated systems in many different ways: for example, a distribution company strongly suspected a small PC interface company, contracted to carry out some work for them, of introducing the Joshi virus. In a construction

company, a virus was found installed on a new Spanish-built PC. In a government department, CMOS1 was brought in on a portable PC used by an officer on a visit to Spain, and a disk brought into a training course by a delegate was infected with Form. A PC in a manufacturing company, after being checked by an outside engineer, became contaminated with Michelangelo.

The tales are endless, but others worth recounting include that of an organisation which had made 34 of its IT staff redundant. It was subsequently discovered that a number of blank disks had been infected with Form and replaced in their boxes. Nine PCs out of 40 on a network were affected when the disks were used and the company estimated that the cost of lost business, investigation and disruption over a twelve-day period amounted to £50,000.

"probably more than with any other threat to IT, protection from viruses lies first and foremost in the hands of the end user"

A major retailer was continually re-infected over a four month period by Form. The source of the infection was eventually traced to a software house which was supplying the company with a bespoke system. In total, about 70 'man-days' of effort were expended in investigating the outbreaks at the company's computer centre and at its head office, and a disk scanner had to be hired. The cost was put at £10,000, in addition to the considerable ill feeling amongst staff who were blaming each other for the continual re-infection. The company now has a mandatory disk authorisation system installed.

In a similar case, a virus was introduced to several machines in the area office of an insurance company from a master floppy disk purchased from a software supplier. The virus was then transferred from the original machines to several other machines via floppy disks. Several thousand of these disks then had to be checked. Whilst most people could start work again within a day, some users had to wait up to a week before being able to use their systems. The incident cost the company between £1000 and £5000. The software supplier's only concession was a letter of apology.

Overall Costing

Respondents who had not costed an incident when it occurred were asked to estimate its likely cost (under £1000, £1000-£4999, £5000-£9999, £10,000-£50,000, or more than £50,000). The majority of incidents were estimated to have cost under £1000, but 23% were thought to be between £1000 and £4999, and 2% at over £50,000, although costs had not been itemised at the time. In addition to immediate costs of disruption and investigation, the long term costs (e.g. installing anti-virus software and training users) were estimated at over £5000 in nearly 20% of all incidents.

Based on details of the actual and estimated immediate costs supplied, most virus infections (approximately 70%) cost less than £1000. However, 20% of incidents detailed cost between £1000 and £5000, and 10%, over £5000. The average immediate cost was £3922. If the results of respondents are representative of the United Kingdom as a whole, we estimate that the annual cost of virus infections could be circa £128 million.

Standards and Procedures

What, then, is going wrong? We asked those respondents who had reported details of virus infections whether or not they had relevant standards and procedures in place: 87% said they had, and 76% said that their staff were adequately trained in their use. However, when asked if these staff were adhering to the standards when the infection occurred, 75% said they were not.

This seems to be a common problem with all types of logical breaches of security. End users appear to be able to grasp the physical dangers to their PCs (fire, theft, equipment failure for example), but are totally unable to understand the concept of these invisible things called 'data' or 'software' and the threats which endanger them. This is reflected in the type of incidents encountered.

"investigation costs (for virus infections) of between £10,000 and £50,000 per incident were not uncommon"

In one case, a company made regular backups of its free-standing PCs by taking a portable tape drive to each workstation in turn. The backup software was loaded from a floppy disk each time. The disk was not write-protected and became infected with a boot sector virus which was present on the PC of a senior manager, whose cavalier attitude to virus protection was well known. Over the following three-day period, other PCs became infected, as the backup software was loaded on each in turn. Once the infection was discovered, it took a further three days to clear the infection from all PCs in the company.

In another example, Form infected a disk fax and was unwittingly distributed to multiple sites of a government department. It was picked up by one site immediately, but at another, a machine was infected as the user booted up his PC with the disk in the drive. That user then posted disks to eight other sites before going on leave for three days; no one else knew to which other sites disks had been posted! Two of the sites were infected. The major impact of this incident was growing acrimony amongst staff.

A common source of infection is illustrated by the member of staff of an educational establishment who was studying at a local college. He brought back a disk infected with

Cascade, which he loaded without permission onto a stand-alone PC used for maintaining accounts at the establishment where he worked. It took two days for the local authority's technical support team to clean up the infection, during which time it was not possible to deal with account and telephone enquiries. A clerk then had to work overtime to catch up on the backlog of work. The total cost of the incident was estimated at £2000.

According to two-thirds of the virus infection reports, standards were revised following the incident. Disciplinary measures were taken in 15% of cases.

Which Solution?

What can be done to counter the problem? Probably more than with any other threat to IT, protection from viruses lies first and foremost in the hands of the end user. Unless mandatory control systems are implemented, or floppy disk drives are locked or removed, with all software and data downloaded from a file server, ignorant or careless actions by staff will increase the risk of infection.

Every employee must receive proper IT security awareness training. This should cover all aspects of PC security (i.e. backups, theft, data protection, copyright and viruses). Such training should be designed so that staff understand fully the impact which all breaches may have on the operation of the business for which they work.

In particular, it should cover how viruses can be introduced, what to do if an infection is suspected, and how every individual must be responsible for the protection of their PCs from the threat which viruses present.

Unless measures such as those described above are consistently introduced and implemented, the security breaches and virus attacks which are being experienced in businesses throughout the UK (and indeed internationally) will not only remain but increase.

Chris Hook, MCBS, ACIB

Chris Hook is a Managing Consultant with the *NCC Business Technology Group*, and has particular responsibility for IT security consultancy assignments for clients, and for presenting security awareness seminars to IT end users.

Prior to joining the *NCC*, he was Computer Auditor at *Rochdale MBC* and was Chairman of the *Greater Manchester Local Government Computer Audit Group*. He is a member of the *British Computer Society* and an associate of the *Chartered Institute of Bankers*.

The National Computer Centre (NCC)

The *NCC* is an independent provider of advice on every aspect of IT, a role it has held for nearly 30 years. Its consultancy service provides risk analysis of IT security, contingency planning for IT systems, network security and audit, general IT security reviews, and assistance with corporate IT security awareness programmes.

The *1994 IT Security Breaches Survey* is available from the *NCC* at £145 + £4.00 p&p. For further information, contact Jayne Howell on +44 (0)161 228 6333.

PRODUCT REVIEW 1

Sweeping the Boards

Jonathan Burchell

Sweep for NetWare is a package from *Sophos Plc*, an Oxford-based company specialising in PC virus detection and data security products, and sister company to *Virus Bulletin*. *Sophos* has an enviable reputation for quality products and is well-known for its innovations in virus detection, in addition to their other data security products. In view of this, I was very interested to see how the *Sweep* NLM measured up.

On Offer...

What's in the box? Everything bar the kitchen sink and a T-shirt! The package includes *Sweep for NetWare* (supporting versions 3.x, 4.x and SFTIII) and a full copy of *Sweep for DOS*. The software comes in 3.5- and 5.25-inch formats, with product manuals and quick reference cards which cover installing and using the products. Two sorts of sticky label are also in the package, some stating 'Virus-checked with *Sweep* on {date}', and others, bright yellow, labelled 'Virus' (these are great fun to put on other people's computers or disks or to identify your records at parties).

The documentation is extremely well produced: an A5, ring-bound manual with a table of contents even more comprehensive than the index! A troubleshooting guide, a glossary, and a section on *NetWare* viruses make up the appendices. A copy of *Sophos' Data Security Reference Guide* is also included in the package: it is a mine of information on virus technology, protection and removal, and covers the other *Sophos* products in considerable depth.

Installation and Administration

No specific install program is provided for the *NetWare* component; however, as installation consists of copying a single NLM to the server, it is reasonable to have to copy it manually. If later versions of the product include more files, I would expect to see a proper installation routine.

Sweep is a little different from most *NetWare* anti-virus products in that the NLM provides background and immediate scanning for files on the server, whilst real-time checking of files is performed by co-operation between a workstation component (*InterCheck*) and the NLM. It seems logical to divide the review into these components, so the real-time issues are dealt with separately below.

Sweep is started on the server by loading a single NLM, which brings up the combined administration and reporting screen. This is the only way to administer and view the status of *Sweep*. It is of course accessible from a workstation by using RCONSOLE; however, neither DOS nor *Windows*

clients are provided for monitoring or administering the NLM. Additionally, *Sweep* is a single server product, without the ability to organise groups of servers into a single logical administrative domain. Multi-server sites must administer and update each server individually.

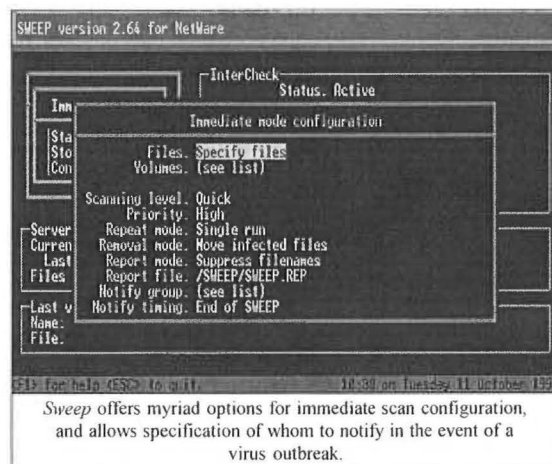
Immediate Scanning

The immediate mode option allows the starting and stopping of an immediate scan, in conjunction with scan configuration. Files, volumes, scanning level, priority, repeat mode and removal mode are individual options which can all be tailored in a scan.

The Files option allows selection between 'All Executables' (defined as files with extensions EXE, COM, SYS, and OV?), 'All Files', or a user-defined list of files and directories. It is also possible to specify which *NetWare* volumes are to be scanned. The Files option generates a single global list and is not organised on a 'per volume' basis, making it difficult to tailor the file extensions scanned in this manner.

Scanning level can be either quick or full: a full scan inspects an entire file for virus code, whilst quick scanning checks only the most likely parts. *Sophos* claims that the quick scan is five to ten times faster than the full. High or low scanning priority may be selected: low priority allows performance degradation due to scanning to be minimised if the server is heavily loaded or under-powered.

The repeat mode option selects either a single execution of the defined scan or continuous repetitions. The removal mode offers actions to take on discovering an infected file. These are no action, renaming the file, moving the file to a quarantine directory, deleting the file, or deleting and purging the file (this makes it impossible to recover the file using *NetWare's* 'undelete' facilities).



Reporting and Notification

Virus incidents are logged to a user-specified file: either all filenames, or infected file names only, may be recorded. There are no facilities for viewing, filtering or reporting on the log file. Further, as no documentation on possible entries in the log file is provided, writing your own reporting software may require considerable trial and effort.

When a virus is detected, *Sweep* sends a *Novell* broadcast message to everyone in the designated *NetWare* user group(s), and any users not logged in when the incident occurs are informed when they next log in. Choices are available for timing of notifications: End of Scan, on the First Infected File, or on Every Infected File. No facilities to customise the infection message are provided.

Background or Scheduled Scanning

Scheduled scanning is organised into a list of jobs. Up to 32 separate jobs may be defined, each of which is given a name and a set of conditions which apply when it runs. The time to execute any job(s) may be specified as a combination of time of day (multiple times may be specified) and days of the week. By defining multiple jobs, it should be possible to accommodate any imaginable pattern of weekly server scanning. The other options for scheduling scanning are identical to background scanning (but unique to each job) and allow specification of files and volumes to be scanned, along with the reporting and notification details.

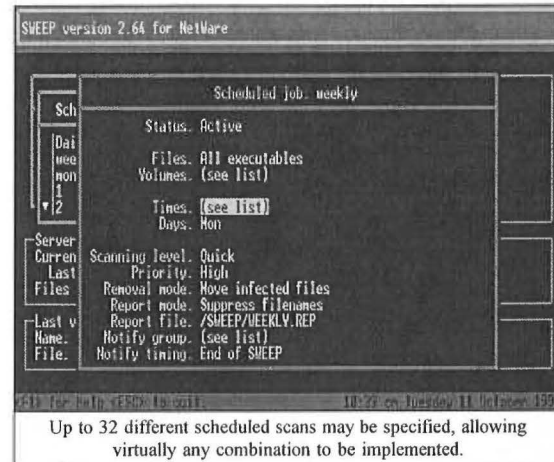
The virus signature database is part of the NLM; a menu option displays the names of all viruses in the current database. If a new virus is discovered, it is possible to get a recognition pattern from *Sophos* and extend the database by including this pattern in an extra file. These 'extra' signatures include a checksum: presumably the NLM itself is protected against tampering.

Real-time Scanning: *InterCheck*

InterCheck is a small TSR (23K) which loads automatically into high memory and can be used on or off the network. It provides workstation protection by checking every executable at the point of access against a list of authorised files stored on the local hard disk. If the file is in the authorised list, execution continues normally. If not, the file is shipped off to the server for scanning. Providing all is well, execution continues; otherwise, file access is prevented and an alert/notification generated.

This adoption of a client-server technique is subtle: with the workstation as the checkpoint, all executables, regardless of source (local, network, or floppy), are spotted, making file inspection happen on the server. This keeps the TSR size constant. Real-time detection obviously uses the same software and algorithms as the background scanner.

Workstation installation involves loading *InterCheck* from a local drive in AUTOEXEC.BAT. *InterCheck* will continue to function even if the workstation is not logged in. If an



unauthorised file is executed, the user will be prevented from using it until it has been scanned. This can be achieved by logging onto the network or by running *Sweep* for DOS.

On the file server, installation involves loading *InterCheck* onto the server and modifying login scripts to load that component automatically from the file server on login. A modified login utility (provided) automates *InterCheck* loading; loading to individual workstations is unnecessary.

InterCheck works both at the DOS prompt and in *Windows*: an optional pop-up box informs the user that a file has not been authorised and is being transferred to the server for verification. When *Windows* is loaded, this magically becomes a *Windows*-based dialogue without any specific *Windows* installation having to be performed. This is a great idea, particularly for users of server-based installations. Installation of *InterCheck* involves simply running either the server or workstation installation batch files.

TSR configuration options are specified in a configuration file. TSR operation is very flexible: it is possible to control what is checked when, what messages and text are presented to the user, and many aspects of general system operation. Like many features in *InterCheck*, the configuration is hand-driven; not a sight of a GUI or menuing-based system to allow 'standard options' to be specified quickly and easily.

NLM configuration for *InterCheck* consists of enabling or disabling *InterCheck* operation, selecting between full or partial file scanning and specifying the notification group(s) and removal methods. Removal can be specified as 'None', or the file may be copied to the server quarantine directory.

DOS Software

The workstation software (*Sweep*) can be installed from the DOS prompt or within *Windows*. The *Windows* 'version' is in fact an icon which launches the DOS version of *Sweep* in a DOS box. *Sweep* can be run from a command line or via an interactive shell, SW. This provides a user-friendly interface, via character-based menuing, allowing fine-tuning

of *Sweep* operation, with access to a good on-line virus encyclopædia and a list of viruses known to the current signature database.

A copy of *SU (Sophos Utilities)*, a disk editor which allows inspection and editing of just about any part of a disk or filing system, is also included. Its operation is not limited to working via the filing system and it can access logical and physical sectors directly. *SU* is provided to help experienced users investigate and repair viral damage at low level.

The *Sweep/InterCheck* system is also available to protect DOS files on servers using *OpenVMS* and will shortly be available for *OS/2* and *Windows NT*.

Conclusions

The only infections missed by the NLM or the DOS scanner (in full mode) were 25 Cruncher samples: this was because neither scanner looks inside compressed files. However, I understand that this feature is due to be added in next month's release. The fact that the DOS scanner missed an additional four polymorphic infections under quick scan illustrates the need to run products at their full capability.

Sweep's detection ratios, considering real-time and background scanning, are the highest we have thus far tested. The facilities provided in terms of user interface and administration are, however, only just above basic. In the hands of a skilled user, there is almost nothing which could not be achieved: the only serious shortcoming is the lack of ability to administer groups of servers as a single entity, which could be tiresome at large sites. Such sites might also be troubled by the minimal logging, reporting and notification facilities.

I suspect that, in the 'real-world' network, many administrators rate ease and sophistication of user interface on a par with detection ability. This is understandable. An administrator must be able to train relatively junior staff to look after the network; they need to impress their superiors (which may include being able to produce snazzy reports) and claim that if anything goes wrong the software will automatically contact them by pager. This attitude may be right or wrong, but I suspect that it is a fact of life.

For the moment, *Sophos* can claim access to high ground on the basis of the outstanding detection ratios and product reliability. The threat comes, however, not just from scanners with equal or better detection ratios but also from products with better administrator interfaces - even those which detect slightly fewer viruses.

Notwithstanding, *Sweep* works extremely well and as far as the user (rather than the administrator) is concerned provides virtually transparent operation and almost 100% detection. If the December release does solve the problem with compressed files, the product's detection ratio will improve commensurately: one hopes that in view of this, *Sophos* will begin to direct its energies more towards the issues of user interface and administration.

Sweep for NetWare

Detection Results:

NLM Scanner (quick and full scan)

Standard Test-Set ^[1]	229/229	100.0%
In the Wild Test-Set ^[2]	109/109	100.0%
Polymorphic Test-Set ^[3]	575/600	95.8%

DOS scanner

Full scan: as NLM

Quick scan:

Standard Test-Set ^[1]	229/229	100.0%
In the Wild Test-Set ^[2]	109/109	100.0%
Polymorphic Test-Set ^[3]	571/600	95.2%

Technical Details

Product: *Sweep for NetWare*

Version: 2.64

Developer: *Sophos Plc*, 21 The Quadrant, Abingdon Science Park, Abingdon, Oxfordshire OX14 3YS, UK.

Telephone: +44 (0)1235 559933; fax +44 (0)1235 559935

Price: File server licence - up to 25 PCs, £495.00; 25+ PCs, £895.00. Site licence for 200+ users - from £12.48 per user. This includes NLM and DOS executables, *InterCheck*, and monthly updates for one year.

Hardware used: Client machine - 33 MHz 486, 200 Mbyte IDE drive, 16 Mbytes RAM. File server - 33 MHz 486, EISA bus, 32-bit caching disk controller, *NetWare 3.11*, 16 Mbytes RAM.

Each test-set contains genuine infections (in both COM and EXE format where appropriate) of the following viruses:

^[1] **Standard Test-Set:** As printed in *VB*, February 1994, p.23 (file infectors only).

^[2] **In the Wild Test-Set:** 4K (Frodo.Frodo.A), Barrotes.1310.A, BFD-451, Butterfly, Captain_Trips, Cascade.1701, Cascade.1704, CMOS1-T1, CMOS1-T2, Coffeeshop, Dark_Avenger.1800.A, Dark_Avenger.2100.D1.A, Dark_Avenger.Father, Datalock.920.A, Dir-II.A, DOSHunter, Eddie-2.A, Fax_Free.Topo, Fichv.2.1, Flip.2153.E, Green_Caterpillar.1575.A, Halloecheen.A, Helloween.1376, Hidenowt, HLLC.Even_Beeper.A, Jerusalem.1808.Standard, Jerusalem.Anticad, Jerusalem.PcVrsDs, Jerusalem.ZeroTime.Australian.A, Keypress.1232.A, Liberty.2857.D, Maltese_Amoeba, Necros, No_Frills.843, No_Frills.Dudley, Nomenklatura, Nothing, Nov_17th.855.A, Npox.963.A, Old_Yankee.1, Old_Yankee.2, Pitch, Piter.A, Power_Pump.1, Revenge, Screaming_Fist.II.696, Satanbug, SBC, Sibel_Sheep, Spanish_Telecom, Spantz, Starship, SVC.3103.A, Syslock.Macho, Tequila, Todor, Tremor (5), Vaccina.Penza.700, Vaccina.TP.5.A, Vienna.627.A, Vienna.648.A, Vienna.W-13.534.A, Vienna.W-13.507.B, Virдем.1336.English, Warrior, Whale, XPEH.4928

^[3] **Polymorphic Test-Set:** The test-set consists of 600 genuine samples of: Coffeeshop (250), Groove (250), Cruncher (25), Uruguay.4 (75).

PRODUCT REVIEW 2

EMD Armor Plus

Dr Keith Jackson

EMD Armor is different from most anti-virus products I have reviewed, with virus-specific features as well as general security components, including secure logons and passwords for multiple users, a screen blanker, and a hot key facility for activating/altering the security features. Although the product works in *Windows*, it does not function under *Windows NT*. *EMD Armor* requires an ISA slot into which a small plug-in card is installed, and software on this card becomes an extension of the PC's BIOS.

Documentation

The 94-page long A5 manual provides a reasonable explanation of installation and use; however, the lack of an index hinders location of specific information. Instructions on how to install the card are particularly well written. Other sections are not: for instance, being told that 'Virus Protection protects your system from viruses' is hardly useful.

As well as inserting the card in the PC, the *EMD Armor* software must be installed from floppy disk. The manual states twice: 'ALWAYS KEEP YOUR INSTALLATION DISKETTE WRITE-PROTECTED'. This is excellent advice, but it was a shame that the developers did not follow it: neither of the floppy disks provided were write-protected.

Installation

The hardware card is simple to install. The PC is turned off, and the jumpers on the card are set so that the ROM space and control port do not clash with other items of hardware. Before power is reapplied, the card is inserted, but remains inactive until its software is installed and executed: this was provided on two 3.5-inch floppy disks (720 KB, 1.44 MB).

Two installation methods are available: 'Quick Install' sets defaults for available features, and 'Detail Install' customises features. By default, the virus protection checks COM, EXE, OVL, SYS and BIN files. Also by default, password protection and the screen blanker are disabled, and automatic repair of hard disk data errors is enabled.

The software will install only on drive C. After installation, execution of a program called SETARMOR will activate the *EMD Armor* card's security features. Any available memory manager must be removed before the card can be activated, which means altering CONFIG.SYS, rebooting, activating the card, changing CONFIG.SYS again, and re-rebooting.

The software is not happy when used with 4DOS (a substitute for COMMAND.COM). When my test PC was booted with 4DOS in use, the message 'Expanded memory not

available or unusable' was displayed, and the PC then hung. Using COMMAND.COM in place of 4DOS removed this error. *EMD Armor* can be installed under *Windows*, and a problem-free installation program is provided for this purpose. After installation, the scanner/disinfector and SETARMOR programs can be executed in a DOS box under *Windows*, along with a *Windows*-specific version of the memory-resident monitor program.

Hardware Detection

The *EMD Armor* card can prevent actions it thinks may be caused by a virus. When such an event is detected, the software pops up a box requiring the user to select one of several actions. The product can detect when a boot sector virus is trying to take control or to hide in memory, when a protected file is being changed, when the CMOS settings are being altered, and when a virus is writing or formatting 'vital sections of your hard disk'.

It is very difficult to test such behaviour-monitor features thoroughly, but those I tried did respond as claimed, and did not suffer too much from false positive reactions. However, whilst writing to a disk, CHKDSK can induce an error which states that DOS is being bypassed. The disinfection part of *EMD Armor*'s CLINIC program can also be persuaded to invoke the same error, which demonstrates a lack of consistency on the developers' part. Surprisingly, although *EMD Armor* thought altering CONFIG.SYS was an 'illegal write', altering the companion file AUTOEXEC.BAT was permitted.

Detection of a boot sector virus taking over before DOS boots is only possible because of the addition of hardware which operates before DOS boots. This is one of the greatest advantages of using add-on security hardware.

Scanning

CLINIC can be used to scan disks conventionally, but the program will not execute unless the *EMD Armor* card is inserted in the PC bus and activated. CLINIC scans various parts of memory when execution commences: if no problems are detected, a simple menu is displayed which offers to scan, clean, immunize, or view a report.

When a scan is activated, a starting path name must be entered, and the subdirectory tree is scanned recursively from this point downwards. All CLINIC reports are kept in a separate subdirectory and labelled numerically so that they are available for future perusal.

The program took 50 seconds to scan the hard disk of my test PC, rising to 1 minute 56 seconds when all files on the hard disk were scanned. Under *Windows*, a scan is carried out at approximately the same speed as under DOS, a

notable achievement. In comparison, *Dr. Solomon's AVTK* took 46 seconds to scan the same hard disk. *Sophos' Sweep* took 54 seconds in 'Quick' mode, and 2 minutes 19 seconds in 'Full' mode. CLINIC's scan times are very good.

Accuracy

The scanner detected only 131 of the 248 virus-infected test samples described in the *Technical Details*, a detection rate of 53%. None of the 500 Mutation Engine-infected test samples were detected. The scanner also detected many of the test samples as infected by several viruses: in fact, they are not. This often seemed to point towards close variants of a particular virus, but that was not always true.

The list of viruses not detected is far too long to quote in full, but the detection rate of the more recently discovered viruses was even worse than the 53% quoted above - none of the viruses contained in the last two upgrades to the test-set were detected. Of the boot sector viruses, Monkey, Quox, Form and V-Sign went undetected.

Memory-resident Monitor

The memory-resident monitor, AUTOSCAN, which occupies 28 Kbytes of memory, allows itself to be multiply-installed: I executed the program five times, and it didn't bat an eyelid. Unsurprisingly, when it was multiply loaded, strange errors appeared when programs were executed. AUTOSCAN introduced a 19% overhead to the time needed to copy a multitude of small files from one subdirectory to another; however, when executing a single program, I doubt users would notice an increase in the program's load time.

I tested AUTOSCAN's detection capabilities by copying a set of 148 virus sample test files (one of each of those listed in the *Technical Details*) from one disk drive to another: the monitor detected 107. The scanner detected less, which probably makes this the only product I have ever reviewed where the memory-resident monitor program is better at

detecting viruses than its companion stand-alone scanner. Like the scanner, the memory-resident monitor failed to detect any of the more recently discovered viruses.

It did, however, detect 23 unique viruses which the scanner missed; vice versa, the scanner thought that five test samples (one each of Aids, Keypress, Necropolis; two of Liberty) were infected when missed by the memory-resident monitor.

After I had finished all this copying, the hardware card forced me to confirm the deletion of each test file. *EMD Armor* was happy for me to copy the viruses, but then did its level best to prevent me deleting them! Isn't life grand when things get quite as perverse as this! I could have replied to the requested confirmation so that it would allow all deletions, but that defeats the point of having the card in the first place. All this happened because the card reacted to the deletion of a file *per se*, not to the fact that the file contained a virus, wonderfully illustrating the problems which products trying to behave as behaviour-blockers can produce.

Disinfection

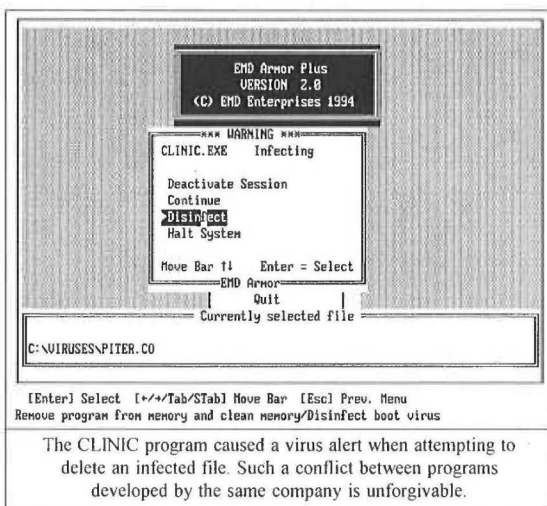
I do not usually test how well a product can remove viruses from infected files; however, the recently published comparative review on disinfection (*VB*, September 1994) tempted me into testing *EMD Armor's* performance.

CLINIC was asked to disinfect a subdirectory containing 148 unique viruses. When an infected file was found, the disinfection option only offered to 'Clean' the file if it thought disinfection was possible; otherwise it offered to delete the file, or did nothing. I selected 'Clean' where offered; however, it often failed and caused another request for action to appear when Clean was no longer on offer.

EMD Armor succeeded in cleaning only five of the files tested: 696, Taiwan, Diskjeb, Datacrime and 1575. The most interesting entry on that list is Datacrime: the disinfection program's scanner failed to detect that file as infected! Another option is a 'Generic Clean', which is defined as a 'proprietary cleaning algorithm for unknown viruses': this means users are not being told how it works. When used, exactly the same five virus-infected files were disinfected.

The disinfection part of CLINIC locked up when it tried to disinfect files infected with Cascade, Virus-90 or 1260. The PC then required a reboot, and CHKDSK later showed that this had left report files scattered around my hard disk which were not properly linked into the File Allocation Table (FAT). Whenever a file infected by either Dec_24th or the Piter virus was encountered, disinfection appeared to have worked, only for the program to retest the same file immediately, say it was infected, and once again offer to disinfect it. Unless the Delete option was selected, this game could be continued *ad infinitum*.

All this is particularly galling when the manual claims that the product 'cleans all known viruses and also cleans unknown viruses by the unique algorithm of *EMD Armor*'. *EMD Armor* is in fact almost useless at disinfection.



Hard Disk Security Features

The product includes a feature called Diskguard, which claims to be a 'Hard Disk Auto Repair' feature. It is meant to repair hard disk data errors 'before they become irreversible due to hard disk aging' (sic). Apart from physically damaging my hard disk, I am at a loss as to how to test this.

Programs are included to add an offset to hard disk storage locations, and to 'lock' the hard disk with a password. I ducked out of testing these features - the manual contains warnings that all information on the hard disk can be lost if it is locked and the password forgotten, to the extent that a low-level format may be required. The hard disk offset feature seems to ensure that if a virus gets past the protection offered by the card, and makes assumptions about the sector structure of the hard disk, then chaos is almost guaranteed.

Other Points

EMD Armor includes features providing user accounting in terms of a logon ID, passwords and a supervisor ID. Each user can be set up to have access to a specific set of hardware features such as floppy disk, printer, serial port and various subdirectories.

The product offers an immunization feature where executable files are modified, and extra code is added to the executable, which checks that a file has not been altered before it is executed. This acts in exactly the same way as viruses themselves, and can seriously interfere with program execution. Only the software developers can implement such features safely. The manual claims that immunization is 'a unique feature of *EMD Armor*'. This is untrue: many products offer this; all should be avoided.

Once the card is activated, it takes control of a long list of interrupt vectors. Communication with a high speed modem was not possible, as a plethora of errors was detected. When the card was deactivated, that problem disappeared. The 'Hot-key' feature provided with *EMD Armor* simply did not work: no matter what I did, no key combination would activate anything. The screen blanker feature is set by default to 60 seconds, but after five minutes of waiting, the screen steadfastly refused to go blank. Any time setting I tried gave the same result. Myriad other problems were encountered, but space restrictions prevent a full description.

Conclusions

The first, and almost blindingly obvious, conclusion must be that *EMD Armor* cannot be used with a laptop, as it requires an ISA slot for its security card. Few laptops have such space. However, the fact that the *EMD Armor* card takes control before any part of the operating system is loaded from disk is a definite advantage.

The scanner is fast enough, but its detection capability is frankly risible; one of the worst I have ever tested. No doubt the marketing people will stress that the capability of the *EMD Armor* card to notice/prevent virus-induced activity is

Copyright (C) EMD Enterprises 1994 Protection Installed				
C:\EMDARMOR <14:14:33>				
Allocated Memory Map - by TurboPower Software - Version 2.0				
PSP	blks	bytes	owner	command line

0000	2	83200	config	
00AD	1	6200	N/A	US,,C:\DOS\DOS\KE...
1837	2	4832	N/A	22 24 2E
19B5	1	15136	N/A	08 10 33
1D68	1	36240	N/A	/D:TSLCD /H:10 /... 2F
2642	2	3184	NDOSEDIT	N/A
26FD	2	23808	GRAB	08 09 13 28
2CC1	2	28736	AUTOSCAN	
33B9	2	28736	AUTOSCAN	
3AB1	2	28736	AUTOSCAN	
41B7	2	28736	AUTOSCAN	
48BD	2	28736	AUTOSCAN	21
4FC3	2	328656	free	
C:\EMDARMOR <14:14:44>				
EMD Armor's TSR component, AUTOSCAN, allowed itself to be installed several times in memory, leading to highly unpredictable results.				

the main line of defence. Even if I agreed (which I do not: behaviour blockers can never prevent all virus activity), it misses the main point - the scanner is pathetic.

The general security features offered by *EMD Armor* may be useful in a situation where several users share a single PC, but this is becoming increasingly rare. Apart from the virus-specific bits, the product has an old-fashioned look and feel. Page 1 of its manual states that '*EMD Armor* is "The State of the Art" in computer security'. It is not.

The disinfection results are one more piece of evidence to reinforce the argument that sensible users do not indulge in such idiocies. Being able to disinfect 3% of viruses will impress nobody, and failing to disinfect the most common in-the-wild virus (Form) is ludicrous. Avoid at all costs.

Technical Details

Product: *EMD Armor*, version 2.0.

Vendor/Developer: *EMD Enterprises*, 606 Baltimore Ave., Towson, MD 21204, USA. Tel. +1 410 583 1575, fax +1 410 583 1637.

Availability: Any PC/XT/AT/386/486 or Pentium computer with one 8-bit ISA (or EISA) slot for a plug-in security card. DOS 3 or higher is required, *Windows 3.1* is optional. 32 KB ROM space is required starting at memory address C800, D000 or D800 (all figures in hex). A control port must be available at one of the following locations - 0220, 0240, 0320 or 0340 (all figures in hex).

Serial number: None visible.

Price: £119.99 with no updates.

Hardware used: A 33 MHz 486 clone with 4 Mbytes of RAM, one 3.5-inch (1.4 Mbyte) floppy disk drive, one 5.25-inch (1.2 Mbyte) floppy disk drive, and a 120 Mbyte hard disk, running under *MS-DOS v5.00*.

Viruses used for testing purposes: A suite of 158 unique viruses (according to the *VB* virus naming convention), spread across 247 individual virus samples, is the current standard test-set. For details, see *VB* February 1994 p.23. A specific test is also made against 500 viruses generated by the Mutation Engine (which are particularly difficult to detect with certainty).

CONFERENCE REPORT

Compsec '94: Alive and Well

This year's *Compsec* was held, in accordance with tradition, in London's *Queen Elizabeth II* conference centre. Spending on security (and the consequent size of marketing budgets) seems to follow the well-being of the economy closely. The corollary to this is that things must be getting better: over 300 delegates attended, and the exhibition, with well over a dozen participants, returned this year to *Compsec*'s agenda.

The conference was divided into four simultaneous streams, which covered a wide and comprehensive range of security issues (see below). It was often difficult to choose which presentation to attend, as the quality of most speakers and presentations was high.

Conference Content

The conference began with a speech by Chris Hook, from the *National Computing Centre*. Using the statistics gathered from over 800 companies as a backdrop (see article on the *NCC* survey, p.14), Hook highlighted the need for IT security to be viewed as a business problem, not just as an IT problem.

He concluded that the costs of security breaches were high, and that delegates should take advantage of the opportunities presented by *Compsec '94* to ensure that they too did not become a statistic in the next *NCC* survey.

After the keynote address, the conference split into four streams, each addressing different security aspects. Day One dealt with Management Issues, Virus/Telecommunications, Financial/Legislation, and EDP Audit. The second day concerned itself with Network Security, Technology/Encryption, Disaster Recovery, and (again) EDP Audit. Open Systems Security, System/Application Specific, Industrial Espionage/Hacking, and (yet again!) EDP Audit comprised the last day's sessions.

The only problem with such a full program was that there were more talks than could actually be attended by one person. Frustrating, but encouraging at the same time!

Virus Information and Education

As *Compsec* is a general security conference, only two talks addressed the virus problem directly, although much of the general security advice given could be used to good effect. Dr Jan Hruska of *Sophos Plc* opened the first day's virus stream with a discussion of viruses on networks, examining the threats and debating how the problem might be solved. Hruska's talk concentrated on *Novell NetWare*, which he believes can be protected from virus attack by the right choice of software.

The next paper, given by Bernard Zajac, discussed the issue of cost-effectiveness of anti-virus software. Zajac concluded that a company must examine whether or not a policy is cost-effective before a solution can be recommended - there is no such thing as a universal panacea.

Internet, Encryption and Disaster Recovery

Day Two devoted some time to networks and the *Internet*: the latter phenomenon seems unstoppable in its growth. Many of the organisations interested in the *Internet* are rightly worried about the potential exposure due to ease of access, both to internal information from the outside and external information from the inside. Brian Neale of *DEC* described his company's approach to the problem and tried to convince the delegates that it is hacker-proof. One remains sceptical. The state of the art in biometrics, as well as future trends, was also discussed in this stream.

The sessions on encryption began with a summary of those techniques, which was presented by Karl Meyer, a well-known data encryption expert. Michael Ganley presented a very interesting case study of the application of digital signatures in a practical banking situation in an ex-Eastern bloc country.

One of the most interesting presentations of the disaster recovery stream was the talk on vulnerability analysis of different building designs to bomb attack given by Bob Ince, of *B Ae Defence*. A number of case studies were presented where very large buildings were described and analysed by a computer program developed by *B Ae Defence*.

Social Studies

The gala dinner took place in the Geological Museum. Unfortunately, it was not included in the conference fee, and as a consequence, not many delegates attended. It was nevertheless a worthwhile opportunity to meet speakers and delegates socially, as well as to have a private viewing of the museum's precious stones. One of the delegates pointed out that he '... was lucky not to have brought his wife along'. Could have been an expensive evening!

Summary

Compsec continues to be the most important international general security conference of the year. The presentations obviously cannot go deep into the matter of each subject, but the event is nevertheless valuable for any security manager needing updates on new security developments.

The exhibitors would doubtless have been happier had the delegates had easier access to the exhibition, but exhibitors are often difficult to please. All in all, congratulations to the organisers - well done *Elsevier*!

ADVISORY BOARD:

David M. Chess, IBM Research, USA
 Phil Crewe, Ziff-Davis, UK
 David Ferbrache, Defence Research Agency, UK
 Ray Glath, RG Software Inc., USA
 Hans Gliss, Datenschutz Berater, West Germany
 Igor Grebert, McAfee Associates, USA
 Ross M. Greenberg, Software Concepts Design, USA
 Dr. Harold Joseph Highland, Comulit Microcomputer Security Evaluation Laboratory, USA
 Dr. Jan Hruska, Sophos Plc, UK
 Dr. Keith Jackson, Walsham Contracts, UK
 Owen Keane, Barrister, UK
 John Laws, Defence Research Agency, UK
 Dr. Tony Pitt, Digital Equipment Corporation, UK
 Yisrael Radai, Hebrew University of Jerusalem, Israel
 Roger Riordan, Cybec Pty, Australia
 Martin Samociuk, Network Security Management, UK
 Eli Shapira, Central Point Software Inc, USA
 John Sherwood, Sherwood Associates, UK
 Prof. Eugene Spafford, Purdue University, USA
 Dr. Peter Tippet, Symantec Corporation, USA
 Joseph Wells, Symantec Corporation, USA
 Dr. Steve R. White, IBM Research, USA
 Dr. Ken Wong, PA Consulting Group, UK
 Ken van Wyk, DISA ASSIST, USA

No responsibility is assumed by the Publisher for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions or ideas contained in the material herein.

SUBSCRIPTION RATES

Subscription price for 1 year (12 issues) including first-class/airmail delivery:

UK £195, Europe £225, International £245 (US\$395)

Editorial enquiries, subscription enquiries, orders and payments:

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

Tel. 01235 555139, International Tel. +44 1235 555139

Fax 01235 559935, International Fax +44 1235 559935

Email virusbtn@vax.ox.ac.uk

CompuServe 100070,1340@compuserve.com

US subscriptions only:

June Jordan, Virus Bulletin, 590 Danbury Road, Ridgefield, CT 06877, USA

Tel. +1 203 431 8720, Fax +1 203 431 8165



This publication has been registered with the Copyright Clearance Centre Ltd. Consent is given for copying of articles for personal or internal use, or for personal use of specific clients. The consent is given on the condition that the copier pays through the Centre the per-copy fee stated on each page.

END NOTES AND NEWS

The proceedings of the fourth annual *Virus Bulletin Conference, VB 94*, are now available. Price is £50 + p&p (£7 in the UK, £17 in Europe, and £25 elsewhere in the world). Tel. +44 (0)1235 555139, fax +44 (0)1235 559935.

Sophos Plc is holding one of its regular **Computer Virus Workshops** at its premises in Abingdon. The two-day event takes place on 25 and 26 November: day one is the Introductory session; day two, the Advanced. Further details available from Karen Richardson on Tel. +44 (0)1235 555139.

The *Tenth Annual Computer Security Applications Conference* will take place in Orlando, Florida, December 5-9. For further information contact Vince Reed of the *Mitre Corporation*, 1500 Perimeter Parkway, Suite 310, Huntsville AL 35806 USA. Tel. +1 205 830 2606, fax +1 205 830 2608.

S&S International is holding a series of **Anti-Virus Workshops both in Germany and the UK**: in Hamburg on 8/9 November, Munich on 29/30 November, and Hertfordshire (UK) on 5/6 December. For the German workshops, call +49 40 491 0041; for those in the UK call S&S on +44 (0)1296 318700.

VSUM listings for September 1994: DOS-based scanning products (figures in brackets indicate when that version of the product was first reviewed in VSUM): 1. *Command Software's F-Prot Professional 2.13*, 96.7% (9406), 2. *McAfee Associates VirusScan v116*, 95.0% (9408), 3. *Dr Solomon's AVTK v 6.64*, 92.9% (9406), 4. *IBM Anti-Virus for DOS v1.05*, 85.5% (9406), 5. *Norton Anti-Virus v3.0*, 76.3% (9406). **NLMs**: 1. *McAfee NetShield 1.6v116*, 93.9% (9408), 2. *Dr Solomon's AVTK v6.64*, 92.3% (9406), 3. *Command Software's Net-Prot v1.25*, 82.5% (9406), 4. *Norton Anti-Virus NLM v1.0*, 75.0% (9406), 5. *Central Point Anti-Virus NLM v2.0*, 59.7% (9403).

Secure Computing Corp has announced the launch of *Sidewinder*, an *Internet* firewall device promising active defence, content-based message filtering, easy *Internet* service access, and one-time passwords. Details from Kevin Sorenson at *Secure Computing Corporation*, 2675 Long Lake Road, Roseville MN55113; fax +1 612 628 2701, Email sidewinder@sctc.com.

The 1994 *EICAR (European Institute for Computer Anti-Virus Research) Conference* will take place in Hertfordshire, UK, from 23-25 November. Presentations include the 'cult' of anti-virus testing, the pursuit and prosecution of virus authors, and the history of viruses. Further information is available by contacting Steve Warren at *S&S International*. Tel. +44 (0)1296 318700; fax +44 (0)1296 318755.

A report in the UK magazine *Computer Weekly* states that **hackers have managed to gain access to British Telecom's 'unhackable' CD-ROM** telephone directory, and that pirated versions are being sold at car boot sales for as little as £35.00 (*BT* sell them for £299.00).

The *Wall Street Journal* reports that teenage hackers in Britain are retrieving large amounts of PC software and pornography after circumventing billing systems for calls to the USA. The hackers have been downloading games, etc, valued at millions of dollars. *BT* claims no knowledge of the incidents, and plans to investigate.

The contact number as given in *VB* (October 1994, End Notes and News) for *Sea Change Corporation Europe*, which produces the *Janus Firewall Server*, was incorrect, being the number for press as opposed to sales enquiries. Those who would like further information on the product should contact the company directly, on +44 (0)1753 581800.

Another dump of viruses has been made to the FidoNet Virus_Info echo. Such occurrences are now happening with monotonous regularity, severely disabling the functioning of the group.