August 1993 ISSN 0956-9979

VIRUS BULLETIN

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

Editor: **Richard Ford** Technical Editor: **Fridrik Skulason**

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

CONTENTS

EDITORIAL

NEWS

INSIGHT

VIRUS ANALYSES

ROGUES' GALLERY

PRODUCT REVIEW

COMPARATIVE REVIEW

EDITORIAL

Crime and Punishment

It is impossible to sit in a court and not instantly be subjected to the feeling of being called into the headmaster's office to have one's wrists slapped. The occasion is gilded with appropriate ceremony and solemnity, and, for most 'computer types', is the last place that they expect their nocturnal tinkering to lead.

With the advent of the *Computer Misuse Act* however, the law enforcers are now able to catch up with the hackers and (hopefully) the virus authors - businesses now have at least some recourse under the law to defend themselves. Unfortunately, simply *having* a law is not enough. Various problems arise when trying to enforce it.

How should one try a complex technical case? Take, for example, the trial of Alfred Whitaker, a computer programmer charged with an offence under Sections 3 and 17 of the UK *Computer Misuse Act.*

The details of the case were relatively simple. Whitaker had been commissioned to write stock control software for an agricultural company, *Protech*. At some time during the development of the software, *Protech* refused to pay a bill because they felt that 'the results to date had been disappointing'. It was at this point that Whitaker modified the software in such a way as to render it unusable after a certain date, unless he was paid.

The trial was scheduled for the 21st of July at Scunthorpe Magistrates court, and was expected to last for one day. However, at the pre-trial review, the Prosecution and Defence had provisionally accepted 10 statements of agreed fact. At the trial Alistair Kelman (quickly becoming the *de facto* standard defence counsel in *CMA* cases) withdrew these points: 'I cannot agree to these points, your Honour', blustered Kelman. 'If you will just examine point number one...'. From this moment onwards the character of the rest of the trial was set.

The legal wrangling in cases like this seems to centre around technicalities. During the cross-examination of Detective Constable White, Kelman suggested that the work of the Humberside Police had been 'slipshod' in that they had not impounded *Protech's* machine. White replied that this was not the case, and went on to explain why. If the police *had* been at fault in their approach to the case, the result might have been very different. The increasing occurrence of computers in crime (either directly or indirectly) increases the problems which the Police will have to overcome.

As cases become more technologically complex, this need for an in-depth knowledge of IT is becoming harder to address in an increasingly underfunded and overloaded force. What life will be like in a 'cost effective' system does not bear thinking about... and what do we do when the criminals start using serious encryption to cover their tracks?

The technical issues within the case were fortunately not too complicated - the defence was far more concerned with the copyright issues. The 'expert' witnesses necessary at such a trial [*What does constitute an expert? Ed.*] were Jim Bates for the prosecution, and, for the defence, a Mr Dilloway from the *British Academy of Experts.*

The report compiled bythe expert witnesses provided cause for some amusement. According to Bates, Dilloway's report was full of pernickety statements, to the extent that at one point Dilloway questioned Bates' use of the term 'floppy disk' when applied to the distinctly unmalleable 3.5-inch media. Between them, the two expert witnesses' reports were large enough to be responsible for the demise of a reasonable sized tree, only to show that at the end of the day, they could not even agree on fundamentals.

The testimony of the expert witnesses followed in a similar vein, taking in such nailbiting points as what constitutes a computer, and what the difference is between a program and data. With neither expert prepared to agree with the other, it was left to the magistrate to intervene, explaining that defining these technical points too precisely would be useless, as he would not be able to follow the argument.

The arguments given above highlight the difficulties of sorting the wheat from the chaff when considering the testimony of expert witnesses, although it is something judges have had to do many times. The danger is that in a trial by jury, it is all too easy to confuse the jurors - if they do not understand the technical issues, what chance do they have of reaching a sensible conclusion?

Notwithstanding Kelman's bellicose rumblings, the stipendiary magistrate, Mr Neville White, did not retire after hearing the barrister's summing up. He found the defendant guilty as charged, although he thought Whitaker was unlikely to re-offend. Kelman argued that Whitaker's business had suffered greatly due to the trial, and that he was virtually unemployed. Taking this into account, Whitaker was given a conditional discharge for a period of two years.

This is an excellent result for all those aficionados of the *Computer Misuse Act*, and the Police who worked on the case. Most importantly, given the untried nature of the act, this result sets an important legal precedent. However, the case also highlights some of the problems of trying complex technical issues. This may have been a step in the right direction, but there are many more to be made.

NEWS

Storing Up Trouble

Rumours continue to circulate that the computer underground is storing up a large number of new viruses in an attempt to flood anti-virus companies in September.

This rumour is nothing new, and has been discussed within the industry for a number of months. However the public has since been made aware of the possibility due to an article in *The Australian*, an Australian newspaper.

According to the article (which opens rather luridly, claiming that the industry is on worldwide alert), the new viruses which are appearing at the moment are either of a technically poor standard or have been produced by one of the virus construction toolkits which are available.

Whether the rumours are true or not is immaterial: there is no need for users to panic. The chances of this level of cooperation among the different virus writing groups is possible, but even if such a worldwide virus dump does occur it will only cause a few months of confusion until the *status quo* is restored ❑

40Hex Print DAME Source code...

The latest issue of the computer underground publication 40Hex contains source code for the polymorphic encryption engine DAME... no, not the Dark Avenger Mutation Engine, but Dark Angel's Multiple Encryptor.

A sample virus which uses DAME is also given in the magazine, both as a hex dump and as source code. As DAME is not very advanced, the only threat which it poses to the community is that the initials may confuse users.

Also in this edition of *40Hex* is an editorial by self-styled electronic freedom fighter, DecimatoR (*sic*) [*It appears to be* de rigeur *in the computer underground to have a silly name. Ed.*]. The editorial attempts to sell the classic line that knowledge equals freedom, and that therefore it should be completely reasonable to publish virus source code on the Internet. The editorial also objects to the way in which the 'self-appointed experts' try to put pressure on those who place this information in the public domain.

While these points have been heard before, it should be noted that part of the reason for the small number of different viruses in the wild is that the majority of specimens are not generally available. Restricting access to information is not necessarily the same as doing harm, and these arguments should be seen for what they are❑

Virus Prevalence Table - June 1993

Viruses reported to *VB* during June 1993.

Virus	Incidents	(%) Reports
Form	17	27.4%
New Zealand 2	9	14.5%
Spanish Telecom	8	12.9%
Tequila	5	8.1%
Maltese Amoeba	5	8.1%
V-Sign	4	6.5%
NoInt	3	4.8%
Eddie	2	3.2%
Joshi	2	3.2%
Parity Boot	2	3.2%
AntiCad	1	1.6%
Flip	1	1.6%
Invisible Man	1	1.6%
Italian	1	1.6%
Keypress	1	1.6%
Total	62	100.0%

Hackers Sentenced to Jail

Elias Ladopoulos (aka Acid Phreak) and Paul Stira (aka Scorpion) have been sentenced to six months in prison and six months home detention by a US Southern District federal court for conspiracy to commit computer related crimes.

Ladopoulos and Stira were indicted with three other computer hackers, (including Mark Abene, better known as Phiber Optik) on conspiracy charges. The five were all members of the group 'the Masters of Destruction/Deception'. All five have since lodged pleas of guilty.

US Attorney Mary White told the court how the MOD group had infiltrated systems from New York to California, including those operated by phone companies, banks, credit reporting services and educational institutions.

According to reports, Stira commented 'I realise that I broke the law. My intent was never to hurt anyone or to make money. I did what I did from intellctual curiosity. I hope that your honour will give me the chance to prove that I have something to give.' These sentences add strength to the increasingly firm message that the US commercial institutions will not tolerate hackers and phreaks❑

IBM PC VIRUSES (UPDATE)

Updates and amendments to the*Virus Bulletin Table of Known IBM PC Viruses* as of 24th July 1993. 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 preferably a dedicated scanner which contains a user-updatable pattern library.

Type Codes

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

Known Viruses

_185 (temporary name) - CR: A 185 byte virus which does nothing but replicate.

```
_185            5350 593D 004B 755C 561E 5053 5152 B802 3DCD B372 4993 B43F
```

_894 - CER: This 894 byte encrypted virus is probably of Italian origin. It contains code that may slightly corrupt data which is written to disk when the virus is memory-resident.

```
_894            5E50 B9B3 0156 2E81 04?? ??AD E2F8 5805 E002 FFE0
```

ARCV.Alpha - CN: A 743 byte virus. Awaiting analysis.

```
ARCV.Alpha      5351 5250 E86D FD2E 8384 C303 01E8 22FF E8D8 FF58 5A59 5BCD
```

Amt - ER: Two variants of this unremarkable virus are known, 3000 and 4000 bytes in length.

```
Amt.3000        813F 4D5A 7403 E953 01E8 C9FC C41E 4800 268B 470F B10C D3E0
Amt.4000        813F 4D5A 740C E801 032B C05E 5F8B E55D C390 E827 03B1 04C4
```

Ash - CN: Two new, encrypted variants, 817 and 1602 bytes in length. Apart from the encryption, they seem most closely related to the 743 byte variant.

```
Ash.817         E802 00EB 213E 8A86 3604 8DB6 3601 B9FE 0230 04D2 C046 E2F9
Ash.1602        E802 00EB 213E 8A86 4607 8DB6 3501 B90F 0630 04D2 C046 E2F9
```

Australian Parasite - CN, CR: This a group of viruses which seem to be written by the same author. The smallest viruses (142, 147, 150 and 153 byte variants) are non-resident, and are located at the beginning of infected files. The longest variants (550 and 615 byte variants) are resident and located at the end of infected COM files. Normally this would mean that the viruses would be divided into two families, with the smaller variants classified as 'Australian Tiny'. However, the 162 byte variant joins the two groups, being non-resident and located at the end of files, as well as sharing substantial code with variants from both of the other groups.

```
Austr.Para.142  B802 3DBA 9E00 CD21 8BD8 B43F BA56 FFB9 8E00 CD21 803E 56FF
Austr.Para.147  B802 3DBA 9E00 CD21 8BD8 BA4F FFB9 9300 B43F CD21 803E 4FFF
Austr.Para.150  B802 3DBA 9E00 CD21 8BD8 BA3A FFB9 9600 B43F CD21 803E 3AFF
Austr.Para.153  B802 3DBA 9E00 CD21 8BD8 BA37 FFB9 9900 B43F CD21 803E 37FF
Austr.Para.162  B802 3DBA 9E00 CD21 8BD8 B905 008D 9621 01B4 3FCD 2189 D6AD
Austr.Para.550  B802 3D33 C9BA 4402 CD21 725C 8BD8 BA45 01B9 0500 B43F CD21
Austr.Para.615  B802 3DCD 2172 618B D80E 0E07 1FB4 3FB9 0400 BA72 01CD 2189
```

Butterfly - CN: A simple 302 byte virus, which contains the text 'Goddamn Butterflies', possibly borrowed from an old Donald Duck story. The Butterfly virus was distributed worldwide in June as a part of*Telemate 4.11*, where two files, 37VESA.COM and 67VESA.COM, were infected.

```
Butterfly       B43F B904 008D 9604 01CD 218B 8655 023D 4E44 749F 80BE 0701
```

Cascade.1704.P - CN: A very minor variant, with some small differences within the related encrypted part. Detected with the standard Cascade pattern, and should be detected by all anti-virus programs which detect Cascade.1704.

Civil War.282 - CN: An unremarkable 282 byte virus, also known as Navigator.

```
Civil War.282   B802 3DBA 1EFE CD21 3E89 86EA 0193 B800 57CD 213E 8996 EC01
```

Civil War.561 - CN: An encrypted, 561 byte virus. The classification of the viruses in this family, as well as the Proto-T family requires further work, and re-classification may occur in the near future. This variant has also been reported as ‘Anti-DAF’.

```
Civil War.561   E800 005D 81ED 0901 8DB6 2301 8BFE B914 028A 2605 01FE CC
```

CyberTech.664 - CN: This virus has not been fully analysed, but it does not seem to be significantly different from the other known CyberTech variants.

```
CyberTech.664   E800 005D 83ED 0750 8DB6 1B00 89F7 B981 02AC 34?? AAE2 FA
```

Dead - CR: A Russian virus, 790 bytes long. The name of the virus is derived from the method it uses to check whether it is already resident. It issues an INT 21H call with AX=DEAD, which returns AX=DEAF if the virus is already active.

```
Dead            3D00 4B74 5280 FC3D 7507 E812 0072 07EB 463D ADDE 7405 2EFF
```

Denied - ER: A 1056 byte Russian virus. Awaiting analysis.

```
Denied          5053 5152 5657 1E06 3D00 4B74 03E9 7301 2E83 3EA9 0301 750A
```

E-Riluttanza - CN: A 689 byte Italian virus. Awaiting analysis.

```
E-Riluttanza    0001 5033 C033 DB33 C933 D233 F633 FF33 EDC3 B409 8BD7 CD21
```

End_of.788 - CR: This virus is detected by the ‘End_of’ pattern, but is 5 bytes longer than the original variant, which has now been renamed to End_of.783

Explosion - CER: An unremarkable 1000 byte virus.

```
Explosion       9C2E 803E AA00 0075 0580 FC4B 7403 E9DE 01FC 5156 5053 5506
```

Flagyll - ER: The code of this virus is 318 byte long, but it overwrites the first 512 bytes of EXE files, as they are executed. The virus may be related to the Proto-T and Civil War viruses - perhaps written by the same author.

```
Flagyll         9C06 1E50 5352 3D00 4B75 03E8 0B00 5A5B 581F 079D 2EFF 2E3E
```

Horns - Special: The Horns virus derives its name from a string ‘Horns of Jericho’ which it contains. What makes this virus unique is that it infects AVR files, which were used by a Dutch anti-virus package as an external virus detection module. Fortunately, the current version of the anti-virus program is not vulnerable to this virus. The virus appends 624 bytes to the AVR file, and re-calculates its internal checksum, making it appear unmodified. When the code in the AVR file is executed, the virus becomes memory-resident, and infects other AVR files as they are opened.

```
Horns           3DA0 4475 0298 CF80 FC3D 756A A803 7572 2E83 3E2D 02FF 756A
```

Ilja - CR: An encrypted virus, 1704 bytes in length. Awaiting analysis.

```
Ilja            1FBB ???? B910 0680 37?? 83C3 0173 078C D805 0010 8ED8 E2EF
```

Jerusalem.Sunday.Unam - CER: A 1631/1636 byte variant, which is detected with the Jeru-1735 pattern.

Kot - CN: A 900 byte Russian virus. Awaiting analysis.

```
Kot             3D00 4B74 069D EA?? ???? ??06 1E50 5351 5256 33DB E86D 018B
```

Kudepsta - CN: A 357 byte virus, probably of Russian origin. Awaiting analysis.

```
Kudepsta        837E FE00 75C5 817E FCE8 0372 BE81 7EFC 50C3 77B7 FC8B FDB0
```

Lesson I.263 - CN: A new variant of this primitive virus, previously (Aug. 92) called ‘Virus Lesson’.

```
Lesson I.263    03D6 CD21 7241 80BC FF00 4D74 35B8 0242 33C9 33D2 CD21 2D04
```

Nanite - CN: A 332 byte overwriting virus.

```
Nanite          B801 3DCD 2172 3B8B D8B9 4C01 BA00 01B4 40CD 212E 8B1E 2901
```

Nazgul.318 - CN: Longer than the original version, but detected with the same pattern.

Paramon - ER: A 917 byte virus. Awaiting analysis.

```
Paramon         3D99 9975 038C C8CF 3D00 4B74 052E FF2E 8202 FA2E 8C16 E602
```

PDP - C(E)R: Three members of this family are known. The smallest one is 822 bytes, and only infects COM files, but the two longer variants (1477 and 1564 bytes) can also infect EXE files.

```
PDP.822         9C2E 803E 1301 0075 381E 0650 5351 5256 572E C606 1301 012E
PDP.1477        9C1E 0650 5351 5256 572E 803E 1601 0075 282E C606 1601 012E
PDP.1564        558B EC1E 0650 5351 5256 572E 803E 1201 0075 352E C606 1301
```

Proto-T.599,901 - CR: These two viruses contain text identifying them as Civil War variants.

Proto-T.599	80FC A075 05B8 0100 9DCF 1E06 5756 5053 5152 80FC 3D74 133D
Proto-T.901	80FC A075 05B8 0300 9DCF 5053 5152 1E06 5754 5580 FC3D 7414

Proto-T.Number 6 - CR: A 631 byte virus. As with other Proto-T related viruses, the exact classification of this virus is subject to review, and the group may be reclassified in the near future.

Number 6	80FC 3D75 03EB 1990 3D00 4B75 03EB 1190 5D5C 5A59 5B58 5E5F

Puke - CER: An unremarkable 393 byte virus.

Puke	3D99 9975 0333 C0CF 3D00 4B75 03E8 1800 2EFF 2E8B 02B8 0242

Radyum.860 - CN: This virus is more polymorphic than earlier variants, and cannot be detected reliably with a single search string.

Rape_II.1639 - CER: This is a 1639 byte somewhat polymorphic semi-stealth virus, which cannot be detected reliably with a single search pattern.

Requires.959 - CER: This variant is quite similar to the other known variant, which was originally called Joe's Demise, but later renamed to Requires.953, and is detected by the pattern published for that variant.

Screeen+1.939 - CER: A minor variant, slightly shorter than the original virus, but detected by the same search pattern. The original variant was first called '948', but is now correctly named Screen+1.948.

SillyC.71 - CN: This 71 byte virus does nothing execpet replicate.

SillyC.71	B802 3DCD 2193 A19A 00BA 4701 8BC8 0547 0050 B43F CD21 5880

Tankard.493 - CR: A 493 byte virus which does nothing except infect files when they are opened or executed.

Tankard.493	80FC FF74 0F80 FC3D 740E 3D00 4B74 092E FF2E 6E00 B834 12CF

Trivial - CN: Several new small, overwriting viruses have been found recently:

Trivial.30.E	B802 3DBA 9E00 CD21 93BA 0001 B440 B11E
Trivial.32	B43D CD21 93B4 40BA 0001 B120 CD21 C32A 2E43 4F
Trivial.34	B43D B29E CD21 93B4 40BA 0001 B122 CD21 C32A 2E43 4F
Trivial.68	B801 3DBA 9E00 CD21 93B4 40B1 4490 90BA 0001 CD21 B43E CD21
Trivial.84	B802 3DCD 218B D8B4 3FB1 54B2 A051 CD21 722D B800 4233 C933

Turn - CR: A 571 byte virus.

Turn	9C50 5351 5256 571E 0655 8BEC 3D00 4B75 731E 078B FAB9 5000

Ugur - CER: A 1297 byte virus. Awaiting analysis.

Ugur	9C3D 4343 7505 B834 349D CF3D 004B 7436 80FC 3B75 0AE9 D302

Ultimation - EN: This is a 23802 byte overwriting virus, probably written in C. The following search string should be used with care.

Ultimation	5845 0063 6F70 7920 0020 0020 3E20 4E55 4C00 0A49 276D 2062

Ungame.770: Very similar to the virus originally reported as Ungame, and detected with the same pattern. The original version has been renamed to Ungame.766.

V3000 - CN: A 3000 byte virus. Awaiting analysis.

V3000	B8C2 3DCD 218C C68B D8C6 444F 00B8 0042 B900 00BA 0000 CD21

Wanderer - CR: This 400 byte virus contains the text 'As wolfs among sheep we have wandered'.

Wanderer	80FC 4B75 03E9 6300 80FC 4E74 2F80 FC4F 742A E9CF 0020 4173

Wave.373 - CR: This 373 byte virus is probably of Russian origin. It does not appear to do anything interesting.

Wave.373	80FC FF75 04B8 CDAB CF80 FC4B 7523 538B DA80 3F00 7403 43EB

Willow - ER: This 1870 byte virus has not been fully analysed, but one interesting feature has been observed - different samples of the virus have different entry points, perhaps in order to confuse certain anti-virus programs. The main effect of the virus is to delete COM files when they are executed.

Willow	B442 CDFD 7204 5B59 5DC3 BAFF FFB8 FFFF EBF4 558B EC1E 5657

Yam.3599 - CR: A fairly complex, semi-stealth virus from the YAM group. Awaiting analysis.

Yam.3599	502E 8A24 80F4 AA2E 8824 46E2 F458 C3B8 42F2 CD21 81FB 2F24

INSIGHT

Getting to the Point

Central Point Software is now firmly established as one of the biggest players in the anti-virus industry. *MS-DOS 6*, which includes a copy of *Central Point's* anti-virus product with the operating system, will help to make *CPAV* one of the most widely used virus scanners in the world, making Jim Horsburgh, Managing Director of *Central Point International*, an important figure in the anti-virus world.

Vive La Différence!

Has Horsburgh found the anti-virus industry very different from *Central Point's* more usual business areas? 'Yes. If you go back two and a half years to when we first began to get involved in the anti-virus market, it was quite a big surprise for us. We spend most of our time addressing customer needs - that's what drives products like our classic product, *PC Tools*. The anti-virus business seems to be not at all customer driven... we're trying to change that.'

Central Point's opening steps seemed less than certain. Horsburgh believes that it has now found its niche. 'When we talk to customers we find it very easy to deal with our product, but in those early days it was a very new experience for us. I think we wasted some time in the first six months trying to get to grips with that, and realised that there was a mismatch - we weren't like that part of the anti-virus community. What we focused on was trying produce our product well, take it to the market, consistently say the right thing... build a good business out of it - make no mistake, it is a business that we are in to make a profit out of.'

A Piece of the Action

Even as Managing Director, Horsburgh does not have precise figures of *Central Point's* market share. 'The data is absolute rubbish in this area. We can get *SPA* type of software data, but a lot of the other vendors are not in the *SPA*. A lot of the business is done into large accounts, and is not reported at all. The kind of data you *can* get is the *Dataquest* data, which I think indicates that in the US, *Central Point* is the number one or the number two with *Symantec* - that's on product selling now. Installed user base is a different issue - I think *McAfee* is still number one, as basically they gave away the product much earlier on.'

The UK market is very different however: 'If you talk about the UK it is really quite interesting. If you take distribution of products we still actually do pretty well.I would genuinely say that in the UK, *S&S* has a bigger market share than us. In terms of sales through the distribution channels the figures tell us that we are top, but market share is very different. I have no idea how *Symantec* is doing in any depth. For companies like *Sophos*, we would have to do a Dun and Bradstreet on them, and to be honest they are so small - I don't mean to be rude about that - that we have not got down to tracking that level of business. Companies like *S&S*, which does well in the UK, aren't encountered in our other territories at all. If you can't be one of the top few in the US, you can't be one of the big ones in the world.'

Truly Safe Six?

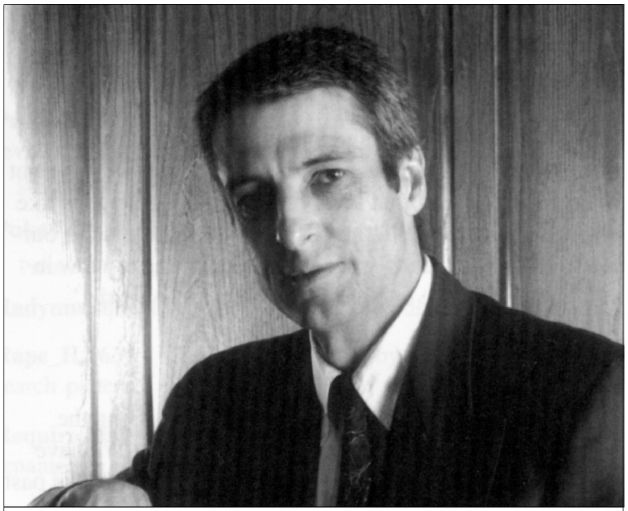
The first question on anyone's lips when discussing the *Central Point-Microsoft* deal is obvious. Why? 'We have had a good experience of working with *Microsoft* in the past - DOS 5 had some basic utilities in it that were supplied by *Central Point*. That was good for our relationship with *Microsoft* and also introduced our utilities to a wider base of people. Given that *Microsoft* were going to carry on down the path of putting some more basic utilities in with the operating system, it was a good place to be.'

> *"if you don't compete in the biggest and most competitive marketplace in the world, you won't have the bucks to go all the way"*

Horsburgh believes that this deal was beneficial for a number of reasons. 'If they are going to use somebody's technology, it is very nice for it to be yours, and the *MSAV* product that is in there - while being quite a basic anti-virus utility - still does for those end users who are only going to be exposed to the handful of viruses actually in the wild.'

'The other reason is that it does get us involved in the upgrade business, which in the first instance is not a highly profitable part of our business - it's a very low cost update that is offered through the *MSAV* operation, but that is something that gets us closer to other companies, and of course we can offer them *Central Point Anti-Virus*. Somebody who is using *MSAV* now could take those signature updates and move in that direction. They could also consider moving to the significantly better technology - and I have no apologies for saying the better technology - of *CPAV 2.0*.'

'But from a corporate point of view, *MSAV* has been very very good for us. Whether you like it or not, *Microsoft* is extremely important to large accounts - even those large accounts which sometimes have an ambivalent attitude



Horsburgh: ‘The anti-virus business seems to be not at all customer-driven... we’re trying to change that.’

towards *Microsoft* - all of those corporates will agree that *Microsoft* is very significant to them, and we have already had a tremendous amount of customers from them.’

A Foot In the Door

DOS 6 let *Central Point* include some features which, Horsburgh believes, will make it easier to sell *CPAV* to the customer. ‘*Central Talk* is a communications protocol which allows workstation anti-virus products to talk to a server product (*CPAV* for *NetWare*). Strategically, it is very important for us. Anti-virus protection for servers is one of the most critically important areas for us - if you are going into corporate accounts in the future, it is not so much about workstations, as about the file server. DOS 6 provides, if you like, a Trojan horse for getting our anti-virus products in there with *Central Talk* and *Central Alert*, which makes it very much easier for people to adopt our network strategy.’

‘One of the big issues is still that *MSAV* is a product from *Microsoft* - quite specifically’, Horsburgh stresses. ‘It is based upon code which we shipped to them some considerable time ago, but it is *Microsoft’s*, not *Central Point’s*. This is not a bitch, I hasten to say, against *Microsoft*, but there is a confusion in people’s minds sometimes, and that’s something which came over in *Virus Bulletin*.’

Horsburgh makes no apologies for the age of the anti-virus software included within DOS 6. ‘In simple terms, an operating system goes through a much longer cycle of development and testing than a utility product - if nothing else, look at all the different bits which are in the product. We agreed with *Microsoft* the specification of what they were to receive and shipped it to them a long time ago - since then the business has changed a lot. We knew that, and they knew that. But what they wanted to do was deliver something different from what I think the more advanced anti-virus community would have expected - they were very clear about what they were doing, which was to provide some fundamental basic protection against viruses.’

Living By Numbers

The whole of the anti-virus world seems to revolve around numbers. Horsburgh believes that finally the truth will out. ‘Anyone who has any sense knows that you cannot have this scanner approach in a world where there are 30,000 viruses. However, the customer is king. I hate to bring up the sordid business of the customer, but I believe that the customer will eventually see things as they are.’

‘Everyone is starting to agree that the number of viruses we detect is starting to become an ‘Emperor’s clothes’ type of issue. There is no industry which is based on fooling the customer for a long period of time, and I think that the user will see the truth.’ What is Horsburgh’s plan? ‘I think that companies which continually bring out good strong products and which address the market needs will win, and clearly splashing across the front of the box ‘‘Now detects x thousand viruses’’ will soon be seen to be wrong.’

The Road Ahead

Realistically, does Horsburgh think that a large non-specialist company can provide the sort of care which the customer needs? ‘I’m just amazed that anyone should ask the question! When I started out in this business there were something like thirty-five word processors in the UK. People were saying “well of course, you buy a British word processor, because it deals with the pound sign properly” and some local word processors did very well. Now how many local products are there in the market?’

The facts speak for themselves, says Horsburgh. ‘The fact of the matter is that the biggest market in the world is the US. It is a matter of great pride that a lot of great technology that is in US software has come from European development, and also specifically UK development, but the fact of the matter is that that is the big marketplace, and that the leading word processor, spreadsheet, database, graphical software - practically everything - is US developed’.

‘I’ll tell you now that this is where anti-virus software is going to go in the long term. When the barriers get very high to compete in this technology, if you don’t compete in the biggest and most competitive marketplace in the world, you won’t have the bucks to go all the way. We tell this story to our customers and they buy it completely. Not because they are naïve, but because they understand, because they have been buying technology like this for the last thirty years.’

VIRUS ANALYSIS 1

Jim Bates

Daemaen: Multi-multipartism

There has long been a community spirit of self-help and enlightenment amongst the old-style computer programmers and this has produced its own stylised language and code of behaviour. The sharp wit and penetrating observation which abounds within this group has enabled rapid and beneficial development of computing around the world. As the computer underground and virus writers have accumulated, they have attempted to emulate the habits and idiosyncrasies of these original 'hackers' (a once complimentary term).

One of the ongoing fads (particularly in the US) has been to uses IBM graphics characters to identify oneself and one's creations. This virus highlights just this trend as well as the self-centred arrogance mentioned above. The virus is called (by its creator) DäeMåên and it attempts to function both as a boot sector virus and parasitic file infector of COM, EXE, BIN, OVL and SYS files.

As usual, the code is riddled with errors and it will undoubtedly cause system malfunction and corruption on most infected machines. The active code is encrypted when infecting files, but the encryption routine is extremely simple and should pose no problem for most scanning engines.

There are three distinct modes of installation depending on the nature of the infected source. These are executable program file infection, device driver infection and boot sector infection. I shall describe each of these in turn.

Parasitic Installation (COM and EXE files)

When an infected file is executed, the virus code first calls a simple decryption routine before issuing an 'Are you there?' call which consists of placing a value of A7CEh in the AX register and issuing an INT 13h request. If the virus is resident, the same value is returned in the BX register.

If this call fails, processing jumps to the installation routine which checks to see whether the current Memory Control Block (MCB) is the last in the chain. If the MCB is larger than 15359 bytes, it is divided to allow space for the virus code; otherwise the previous MCB (without any size checking) is used. The MCB is divided by creating a new 3072 byte MCB in its final section and then decreasing the machine's base memory size pointer by 3 Kbytes. After this memory has been allocated, the virus code is moved into position and hooked into the system.

Once the virus is installed, it attempts to hook itself in to the DOS services. This is done in a particularly inept manner. The virus searches the DOS code in a highly specific location for a long CALL instruction followed closely by a RETF 2 instruction. If such a structure is found, the far address of the long CALL is treated as the DOS services entry address. Since the author obviously cannot believe that this technique could be flawed, there is no fallback arrangement and if the code structure is not found, the program will simply hang.

If the DOS entry point*is* determined correctly, the virus collects it and inserts its own vector address so that DOS requests can be intercepted by the virus code. Processing continues with a routine which attempts to infect the Master Boot Sector (MBR) of the machine. If this attempt is unsuccessful, the virus attempts to infect the boot sector of any floppy disk in the A: drive. The operation of both of these routines is described in the Infection section below.

Parasitic Installation (SYS files)

A slight variation on the above installation process occurs if the host file has a SYS extension. In this instance, the writer makes the (erroneous) assumption that all SYS files are device drivers and treats them as such. This will undoubtedly damage a target file that has this extension and is*not* a device driver. The device driver infection process is described in the 'Infection' section below. The effect of loading a successfully infected file is as follows:

After loading the file in memory, DOS first calls the area within the device driver known as the 'Strategy' routine. The address of this routine will have been altered by the virus so that the virus code executes first. After the mandatory check to ensure the virus is not memory-resident, the virus code is installed and hooked into the system services. The memory installation point is set to offset zero of the original load address and clashes with the existing host code. This clash is solved by moving the host code temporarily up in memory and then relocating the virus code to offset 0. Then the host code is moved back down to a point 1000h bytes above the start of the virus code.

Once this relocation is completed and the system hooks are set, control is passed to the original Strategy routine. After installation, the device driver is located above the virus code in memory, and the Strategy and Interrupt addresses point into the virus code.

Boot Sector Installation

When booting from a disk containing an infected boot sector, the virus code calculates the top of memory and attempts to read the whole of the virus from pre-determined sectors of the disk into high memory. The ‘top of memory’ pointer is decremented by 3 Kbytes and processing passes to the boot sector installation routine. This hooks the INT 13h disk services address into a temporary routine which will later hook the INT 21h address (which is not available at boot time) and replace itself with the permanent INT 13h handler.

If the system was booted from an infected floppy disk, an additional routine is called which attempts to infect the Master Boot Sector of the first fixed disk on the system. The installation completes by loading the original boot sector into its correct position in memory and passing control to it.

Operation

Once resident, this virus intercepts both INT 13h and INT 21h. Device driver communications from DOS through the Strategy and Interrupt channels are not intercepted, but simply redirected to a corrected segment and offset address.

The INT 13h interception checks first for the ‘Are you there?’ call and for reads or writes to the Master boot sector of either the first hard disk or any floppy drive. Any request to access the boot sector is then redirected so that the contents of the original boot sector are returned/edited.

If the access request is for a floppy disk, the intercept routine checks to see whether the relevant drive motor is already running; if it is, the request is allowed to continue unhindered. Otherwise, a check of the contents of the floppy disk is made, and if formatted at 9 sectors per track, it is infected. This routine can give rise to unexpected system errors if there is no disk in the drive, or the OEM name is non-standard.

The INT 21h infection routine is more comprehensive. The intercepted functions are as follows:

Functions 11h and 12h: The FCB Find First and Find Next requests are subverted so that the returned file length is reduced by an appropriate amount (1000h for SYS files, 800h for others) if the file date indicates that it is infected.

Function 3Ch - The Create a File function is intercepted so that new file handles can be stored within the virus.

Function 3Eh - The Close a File routine is intercepted so that the target file handle can be compared to that collected during file creation. If they match, the file is infected before processing returns to DOS.

Functions 3Dh - Open a File, **Function 43h** - Change Attributes, **Function 4B00h** - Load and Execute, **Function 56h** - Rename a File and **Function 6Ch** - Extended Open/Create are all subverted to enable the file to be infected before returning control to DOS.

Infection routines - Parasitic

The parasitic infection section of this virus targets files which have the extensions COM, EXE, BIN, OVL and SYS. However, there is a possibility that files with an extension that matches intermediate sequential combinations of these characters (i.e. MEX, VLS etc.,) may be infected.

Once the target file extension has been verified, a flag is set to indicate the extension type and the DOS Critical Error Handler vector is hooked to prevent spurious messages appearing on screen. The routine collects the file attributes and ensures the file is not read-only. The attributes are stored for later use.

At this point the file date is checked to see whether a value of 100 has been added to the year field, indicating that the file is already infected. If the file is deemed suitable for infection, the first two bytes are checked for the ‘MZ’ header used in EXE files. It is at this point that processing branches depending upon the file type. There are three branches, for SYS, EXE and binary files.

SYS Infection - Device Drivers

Device drivers have two entry points (known as the Strategy entry and the Interrupt entry) which are stored as offset pointers at pre-determined locations within the code. When DOS loads a device driver, it places the file in available memory at an offset of zero. Communication with DOS occurs in two stages: first DOS calls the Strategy routine with the address of the forthcoming request header block and then an immediate call is made to the Interrupt routine with the relevant driver command. With this virus resident, calls are routed through the virus code into the host code.

The virus achieves this during infection by appending its own code to the driver file and then inserting a modified jump address into the Strategy vector position so that the virus code gains control as soon as DOS begins its initialisation. It should be noted that infected SYS files increase in length by 1000h (4096) bytes.

EXE Infection

For this type of file, the 800h (2048) bytes of virus code are appended to the file and the header information is altered to ensure virus code execution as soon as the file is loaded.

Binary Infection (COM, BIN and OVL)

For the case of binary files, 800h bytes of the virus code are appended to the file and the initial three bytes modified to jump directly to the virus code. In all three cases, the file year field has 100 added to it to indicate infection.

Infection routines - Boot Sector

There are two Boot sector infection routines, one dealing with fixed disks and the other with floppies. On a fixed disk, the original Master Boot Sector is collected and stored on Track 0, Head 0, Sector 9. Then the virus boot routine (69 bytes long) is copied over the original boot code and rewritten back to Track 0, Head 0, Sector 1. Finally, the whole of the virus code (800h bytes or 4 sectors) is stored at Track 0, Head 0, Sectors 10 to 13.

On floppy disks (in either drive A: or drive B:), the process is a little different. First, it should be noted that only floppy disks with nine sectors per track are infected. The infection process is similar to that used on fixed disks except that the last track is used for storage of the original boot sector and the complete virus code. In this instance, depending upon the floppy capacity, the original boot sector is stored at Track 39 or track 79, head 1, sector 4 and the virus code occupies the following 4 sectors (5 to 8). No check is made to see if these sectors are already in use.

In both boot sector infections, the presence of an ID marker word (A7CEh) is used as an indication of prior infection. On fixed disks this will be found at offset 2 of the Master Boot Sector and on floppy disks it will be at offset 9 of the floppy boot sector.

Additional Observations

With the usual sententiousness, this virus contains a number of messages within the unencrypted code. The first is a ‘name plate’ which appears at offset 0103h of the code and affectedly identifies our hero and his creation as‘[DäeMåên] by TäLöN-{NûKE}’

The next message at offset 01B6h offers ‘Hugs to Sara Gordon’. This refers to the lady who has recently published a purported ‘interview’ with the Dark Avenger. Fairly predictably, this may be an attempt to gain that lady’s attention so that she might grant him some sympathetic publicity too.

A message apparently addressed to John McAfee appears at offset 039Ah and this offers - ‘Hey John! If this is bad, wait for VCL20]!’. I doubt that Mr McAfee needs to worry too much, as this is a poorly written virus and is unlikely to cause any problems in detection or eradication. The comment about VCL20 may be a reference to The Virus Construction Laboratory - the magnum opus of the NUKE group of virus writers. The final message in this code is a sweet little dedication - ‘For Dudley’ at offset 05CCh. I can only suggest that if Dudley reads this and can identify the writer of this virus, he should let us know immediately so that we can take his admirer into care for gentle remedial treatment with thumbscrews and hot irons.

DAEMAEN

Aliases:	None known.
Type:	Multi-Partite (Parasitic and Boot Sector)
Infection:	EXE, COM, BIN, OVL and SYS files (any length), Master Boot sectors.
Self-Recognition:	
File	Date value has 100 added to the years field.
Boot Sector	Word value of 0A7CEh at offset 2 of Master Boot Sector and offset 9 of floppy boot sectors.
System	‘Are your there?’ call. INT 13h call with a value of 0A7CEh in AX, returns 0A7CEh in BX.

Hex Pattern	In memory, and for Boot Sector infections (no file recognition hex pattern is possible since parasitic infections are encrypted).
	`3DFF 1E75 F9B8 CA02 3944 0474`
	`0539 4405 75EC AD96 56BF 2108`

Intercepts:	INT 13h for redirection of boot sector read/write requests. INT 21h Functions 11h and 12h for hiding changes in file size Functions 3Ch and 3Eh for targeting newly created files Functions 3Dh, 43h, 4B00h, 56h and 6Ch for infection INT 24h for internal error handling.
Trigger	This virus has no trigger routine but will cause occasional corruption to both files and disks.
Removal	Specific and generic disinfection is possible. Under clean system conditions identify and replace infected files. For removing Boot Sector infection, use the command FDISK /MBR under DOS 5.0. Under earlier versions of DOS replace the infected boot sector with the contents of Track 0, Head 0, Sector 9.

VIRUS ANALYSIS 2

Eugene Kaspersky

8888: The Poor Man's Commander Bomber

The many different ways in which computer viruses may replicate have been well analysed by virus researchers over the last few years, and are well described in numerous articles. In the case of parasitic viruses, there are a number of 'standard' ways to infect a file. The most common technique is that of the ordinary appending file infector, where the virus code is stored at the end of the infected file and a JMP instruction inserted at the start.

In some cases, the virus code is inserted into the middle of the file. For example, the Commander Bomber virus adds its code into the middle of a file, and then inserts a sequence of jumps and 'junk' code which eventually executes the virus proper. Notwithstanding this extra complexity, these methods all have one thing in common: it is possible to trace through the file and find the starting point of the virus.

All viruses analysed to date have this one feature in common: it is not necessary to scan the entire contents of a file in order to detect the presence of a particular virus - only examining the entry point of the file and seeing where this leads to is necessary. [*Although it is* possible *to trace through the 'junk' code in the Commander Bomber virus, some vendors do a complete file scan anyway. Ed.*]

Sneaking In

The 8888 virus removes the delightful certainty of being able simply to trace the execution path, albeit at the cost of reliability and viability. However, the virus does not need either: simply by existing and functioning it forces the anti-virus software vendors to sit up and take notice.

This feat is accomplished in a relatively simple-minded way. The virus utilises two different infection mechanisms. The first is a simple appending file infector technique, complete with JMP instruction inserted at the start of the file. In this case, the virus is executed as soon as the host program is run, and behaves just like any other file infector.

The second technique is slightly more complex. The middle of the host code is overwritten without any alteration to the start of the file. Therefore the virus code is not called upon execution, but in the event of the piece of overwritten code being executed, the virus will run and become memory-resident. In effect, the virus leaves a 'dropper' concealed within any executable infected in this manner.

Infection

Due to the novel way in which the virus can be executed, it is more expedient to examine the operation of the virus code when it is memory-resident before going on to examine how the virus actually becomes active in memory.

When the virus is memory-resident, it intercepts five INT 21h subfunctions: 7777h, which is used as the virus' 'Are you there?' call, and 3Dh (Open handle), 4Bh (Load and Execute), and 6Ch (Extended Open/Create) for file infection. If any of these functions are called, the virus checks the name of the file in question to see if its extension ends with OM. If it does, it assumes that the file is a COM file, the file is opened and the first four bytes are read in. If the fourth byte of the file is F4h, the virus assumes that the file is already infected and the routine aborts. In addition to this check, the virus also ensures that the length of the file is less than EE00h (60928) - if this condition is not met, the file is again deemed unsuitable for infection.

If the virus uses the common appending file infection technique, it then writes 512 bytes of code at the end of the file, and saves four bytes (a JMP Address, and an F4h to serve as an infection marker) at the beginning.

"The 8888 virus removes the delightful certainty of being able simply to trace the execution path''

During the infection process, the virus neither examines nor preserves the target file's time/date stamp or attributes. It is incapable of infecting read-only files, and the time and date of infected files is different after infection. In addition to these oversights, the virus does not trap the DOS critical error handler, INT 24h. This will cause the standard DOS message 'Write protect error writing drive A Abort, Retry, Fail?' when the virus attempts to infect files on a write-protected floppy disk.

A Corrupting Influence

When the virus intercepts any INT 21h subfunctions 4Bh and 3Dh, it uses its second type of infection routine. This time, rather than appending the virus code, it overwrites a random sector of the file with the body of the virus. This occupies exactly one sector within the file, as the virus code is 512 bytes in length.

Nothing within this corrupted file is specifically altered to point to the virus code. However, if the overwritten section of the host file is executed during program operation, the virus can become memory-resident.

In many cases, the last instruction of a corrupted program will overlap with several bytes of the virus code. For example, a simple MOV seg:offset, AX instruction consists of the instruction identifier A3h followed by the address. If the virus code happens to be inserted immediately after an A3h which forms an instruction boundary, the next thirty-two bits will form part of the preceding instruction. The start of the virus code is written to take this eventuality into account: the first eight bytes of the virus code are instructions which are designed to take up this overlap.

Slipping Through

When an infected file is executed, its first operation is to check whether the virus is already memory-resident. This is done by using an 'Are you there?' call - if INT 21h is called with AX=7777h, the value 8888h is returned. If the call is not returned, the virus begins its installation.

The virus uses two different methods to install itself: one when it is executed from an infected COM file, and the other when it is executed from a corrupted file. The virus attempts to identify the two cases by examining the address of the current Program Segment Prefix (PSP) (obtained by using INT 21h, AH=62h). If the CS register value is equal to the segment address of the PSP, the virus assumes that it is being executed from within a 'standard' infected file. The virus then relocates itself in memory, and uses the DOS functions INT21h, AH=48h (allocate memory) and 4Ah (change the size of allocated memory) to become memory-resident. This technique is very well known, and was used in viruses as old as Cascade.

If the virus is being run from a corrupted file, it copies itself into the top of the PSP segment and uses the DOS call INT 21h, AH=31h (terminate and stay resident) in order to remain memory-resident.

If the 'Are you there?' call is answered, the response again depends on whether the virus was loaded by running a corrupted file or not. In the case of a normal infection the virus restores the original bytes of the program and jumps to its beginning.

If the virus has been executed from within a corrupted file, it is likely that if control is simply allowed to return to the corrupted file, program execution will fail. Therefore the virus deliberately generates an INT 00h (divide overflow) error. This causes DOS to display an error message, and terminates the execution of the program.

In summary, if the file is infected, it appears to function correctly. If it is corrupted, the virus passes control to DOS, simulating a bug in the host program. Therefore the presence of infected files is concealed reasonably well.

INT 10h, AX=1001h?

During the infection of a new file, the virus disables INT 10h, subfunction 1001h. This is ostensibly part of the ROM BIOS video services, and is used to specify the border colour. It seems probable that this is done not out of some misplaced sense of aesthetics, but in order to disable some kind of anti-virus reference monitor (although at the time of writing this is yet to be confirmed).

Detection

Although this virus does not pose a very large threat, it does raise some problems for the virus scanner manufacturers. As the start of the virus code can now no longer be traced by a simple analysis of the file, the entire contents of an executable file will have to be scanned in order to ensure that it is not infected. This could have a large impact on the speed of virus scanners. This type of approach may well become more common as the virus authors continue to fight back against the anti-virus research community, and is yet another straw placed on the back of the vendors.

And the final piece of good news: this virus is in the wild.

8888

Aliases:	None known.
Type:	Memory-resident parasitic
Infection:	COM files smaller than EE00h bytes.
Self Recognition:	
Disk	Checks the ID-byte for the value F4h at the file beginning at the offset 04h.
Memory	INT 21h with AX=7777h, returns 8888h in the AX register.
Hex Pattern:	
	CD11 5B59 5AE8 0000 5D81 ED0B 00B8 7777 CD21 3D88 8874 5CB4
Intercepts:	INT 21h for infection and trigger routine INT 10h for video trigger routine
Trigger:	Overwrites the random sectors with its code. Disables the Set Border Colour function of INT 10h.
Removal:	Specific and generic removal is possible under clean system conditions. Corrupted files should be deleted.

ROGUES' GALLERY

Keep It To Yourself

It has been almost two years since *Virus Bulletin* last published a summary of viruses which are actually in the wild. The last report (*VB*, September 91, p. 13) showed that New Zealand was the most common virus - these were of course the days before Form.

Since then, things have changed surprisingly little. It would seem that once a virus gains a foothold it is extremely difficult to eradicate it from circulation. The most common viruses back in 1991 (New Zealand and Cascade) still appear in this year's listing - albeit displaced by some of the new contenders.

Almost without exception, the viruses listed below are detected by all of the main anti-virus software vendors. It should therefore be possible to eliminate them from the user community, simply by adopting clean computing practices, and ensuring that all disks used are scanned.

Form

To use the patois from *The Top of the Pops*, this relatively new entry has taken the world by surprise by moving straight in to the number one spot! The Form virus is now the virus most commonly reported to *Virus Bulletin* by far. Quite why this virus is so widespread is not known, but it seems likely that somebody somewhere (possibly a supplier of pre-formatted floppy disks) has distributed a large number of disks infected with the virus.

It is a DOS Boot Sector virus, which becomes memory-resident when the system is booted from an infected disk, taking 2 Kbytes from the top of RAM.The virus then hooks INT 13h and, if the date is the 18th of any month, also INT 09h, the keyboard interrupt. The new INT 09h handler causes a click to be issued every time a key is pressed - a harmless but annoying effect.

The virus contains the following text message, though this is never displayed:

```
The FORM-Virus sends greetings to everyone
who's reading this text. FORM doesn't destroy
data! Don't panic! Fuckings go to Corinne.
```

Form removal is trivial under any version of DOS by using the SYS command. This command writes the boot code into the DOS boot sector of the disk, thereby overwriting the virus. For a full report on the Form virus see *Virus Bulletin*, November 1991, p.16.

New Zealand 2

Still near the top of the table, New Zealand is a Master Boot Sector virus. It was first reported in late 1987 in New Zealand and continues to spread widely throughout the world. In the last published survey of virus prevalence published by *Virus Bulletin*, the New Zealand virus was the most common virus found in the wild.

The original virus stored a copy of the clean Master Boot Sector on hard disks on Track 0, Head 0, Sector 2, but has since been extensively modified; later variants use various different locations.

When the virus triggers, it displays the message 'Your PC is now Stoned', but there is no deliberately destructive trigger routine. The virus can be removed under DOS 5 (and certain OEM versions of DOS 4.x and 3.x) by using the command FDISK /MBR under clean system conditions. The full report can be found in *VB*, May 1990, p.8, and a disinfection procedure without using the FDISK utility appears in *VB*, September 1990, p.9.

Tequila

This virus is now much more prevalent than it was two years ago. It is of Swiss origin, and uses self-modifying encryption and stealth techniques in an attempt to avoid detection. Tequila is a multi-partite virus capable of infecting both the Master Boot Sector and EXE files.

The trigger routine occurs at random and displays a crude graphical representation of the Mandelbrot set (a mathematical oddity which was catapulted to fame by such excellent programs as *Fractint*) on the screen. Although the trigger is benign, corruption will occur if the area the virus uses to store its code contains data.

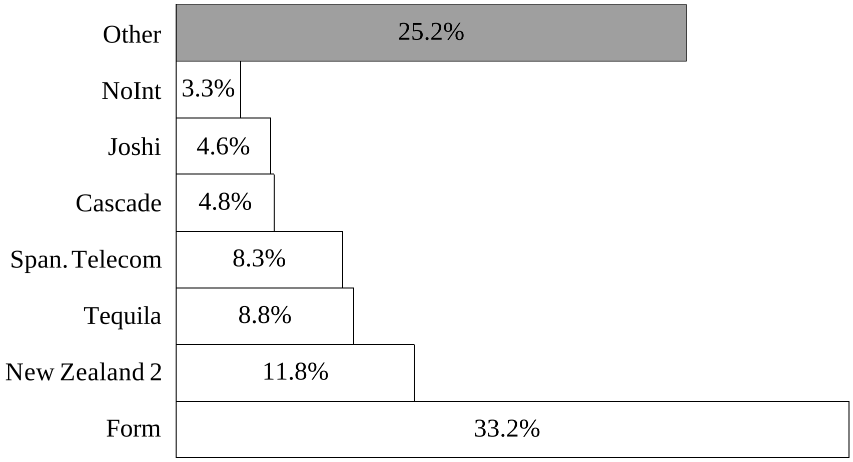
Tequila may be removed from a DOS 5 system under clean conditions by the command FDISK /MBR. However, because of the way in which the virus has allocated its disk space, this makes 3K of disk unusable - in this case, third party software is required for accurate disinfection. Infected files should be deleted and replaced. For further information see *VB*, June 1991, p.16.

Spanish Telecom

Spanish Telecom is another multi-partite virus, though in this case the Master Boot Sector and COM files are infected (COMMAND.COM is excluded from infection).

This virus has a highly destructive trigger which overwrites all data on both the first and (if there is one) second hard drives attached to the machine. The trigger routine is invoked 400 reboots after the machine was first infected.

Figure 1. Viruses reported to *Virus Bulletin* over the period July 1992 to April 1993



Collecting Statistics: The Problems

It is easy to criticise the lack of accurate statistics concerning virus prevalence on a worldwide basis. However, compiling accurate virus attack numbers is hampered by the stigma still associated with virus attacks. It is only recently that companies have been prepared to come out into the open and discuss the problems which they have had in public.

In addition to this, the lack of naming conventions make the data gathered rather prone to errors, and care must be taken to avoid duplicate entries. It is especially important that false positive problems do not cause spurious viruses to be entered into the list of those known to be in the wild.

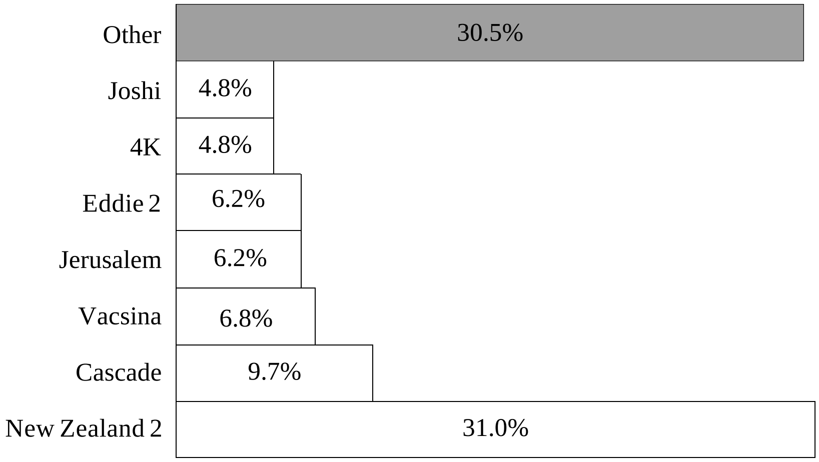


Figure 2. Viruses reported to *Sophos* over the period 25th January 1991 to 22nd June 1991

Virus Outbreaks in the United Kingdom

The demography of virus attacks is still a science in its infancy, and there are few effective models which describe how a particular computer virus will spread and grow, once released in the wild.

The data presented here shows which different viruses were prevalent in the first half of 1991, as compared to the viruses which are common now. In both cases, the data gathered is predominantly (though not exclusively) from within the UK

Even though the number of viruses has grown phenomenally over the last year and a half, it is easy to see that the number of viruses which cause the majority of reported viruses incidents is small: 45% of all virus attacks are caused by only two different viruses.

Another interesting point to note is the similarity between the two different sets of statistics. The only real difference is the demise of most of the file-infecting viruses in the later table. It would seem that if one wishes to write a successful virus, it should be a boot sector virus.

New Scotland Yard's Computer Crime Unit has recently published its virus report statistics for the year running from May 1992 to May 1993. These are in very close agreement with the statistics shown here. In order of prevalence, the *CCU* finds that the top seven viruses are: Form (43), Tequila (37), New Zealand 2 (34), Spanish Telecom (13), Cascade (10), 2100 (7), Jerusalem (6). The numbers in brackets refer to the number of virus outbreaks reported to the *CCU*. The unit can be contacted on 071 230 1177, and any reports are treated in strict confidence.

The boot sector part of Spanish Telecom does not contain the code needed to infect COM files, and so if only the Master Boot Sector is infected, the virus will behave as a boot sector virus from that point onwards. For this reason, the file-infecting version of the Spanish Telecom virus is comparatively rare.

It is important to ensure that a clean boot has been completed before scanning for the Spanish Telecom virus, as it is capable of stealthing virus scanning programs. This is doubly pressing in view of the destructive trigger routine.

Infected files should be deleted and replaced. Disinfection of the Master Boot sector can be achieved in a similar way as for New Zealand - either through FDISK /MBR or by copying the original boot sector (located at Track 0, Head 0, Sector 7) back into place. A full report on this virus was published in *VB*, January 1991, pp.22-24.

The virus became widespread in *Oxford University*, during the period 1991-92. An interview with the staff at the University discussing the problems it caused was printed in *VB*, September 1992 pp.7-9.

Cascade

Cascade is the only simple file-infecting virus in this list. It attacks COM files (including COMMAND.COM), adding approximately 1700 bytes to its length. The virus was one of the first to use simple encryption, although the encryption routine does not change and is easy to detect.

The Cascade virus has one of the most well known trigger routines - it causes letters on a text screen to appear to fall to the bottom of the screen (accompanied by a clicking noise from the PC's speaker). This 'cascading' of the letters combined with its prevalence have earned it a place in the computer virus hall of fame, and it is extremely common to see pictures of scrambled screens appearing in the popular press. For a full report on the Cascade virus, see *Virus Bulletin*, September 1989, p.9.

Joshi

This is a Master Boot Sector virus, first reported in India in August 1990. The original virus triggers on January 5th and displays the message '`Type 'Happy Birthday Joshi'!`' If these instructions are not followed the machine must be rebooted..

Disinfecting Joshi involves either copying the original Master Boot sector back from Track 0, Head 0, Sector 10 into its rightful place, or using the command FDISK /MBR to rewrite the master boot sector. For a full report on the Joshi virus see *VB*, December 1990, p.17.

PRODUCT REVIEW

Keith Jackson

Better *CPAV* than *CPAV*?

I think I am right in saying that *Central Point Anti-Virus* is the program that has been reviewed most by *VB* over the past few years. I have reviewed it as a constituent part of *MS-DOS* (*VB* May 93), and as part of the *PC Tools* package (*VB* January 93). It has also been reviewed in its own right in the June 91 and May 92 issues of *VB*.

Central Point Anti-Virus contains one main program which provides virus detection and clearing, integrity checking, immunization and a scheduler. An on-line 'virus list' is available which can be interrogated to provide information about particular viruses, and programs are included which provide memory-resident protection against virus activity.

Given that *Microsoft* has stated publicly that it is not paying any royalties for including *Central Point Anti-Virus* with *MS-DOS*, my recent review of *MS-DOS* led me to query exactly what *Central Point* would get out of such an arrangement. The only logical conclusion seems to be that they intend to get users hooked into using their anti-virus software, and then persuade them to upgrade for new/improved features. To this end it is probably no coincidence that v2.0 of *Central Point Anti-Virus* has just been released.

Among the plethora of new features on offer are options such as scanning within compressed files (any files in LZEXE, ARJ, PKLITE or PKZIP format), a virus analyser which looks for virus-like executable code, 'smart' (their word, not mine) signature files, a fast verify option, usage of memory-resident programs from upper memory, and an audit trail. Is it worthwhile upgrading to version 2 for all this?

Installation

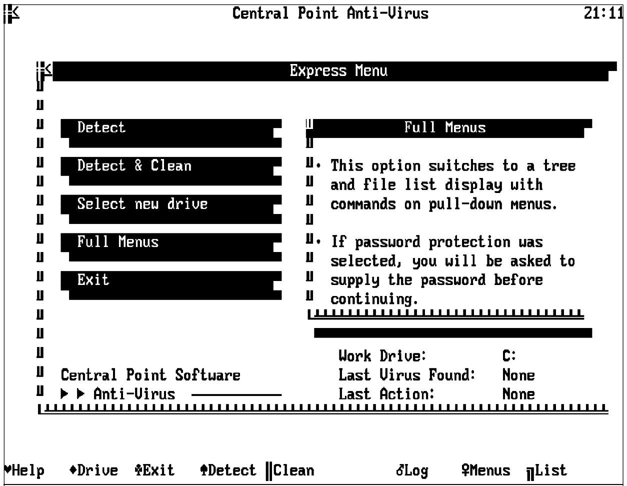
Central Point Anti-Virus has always been very straightforward to install, and this latest version is no different. The available options are all very clear, and the installation process can be either mouse- or keyboard-driven. The installation program sensibly offers to scan the drive to which software is installed before files are copied across. Note that users have a choice between 'express' menus in which just the basic virus detection and cleaning options are available, and 'full' menus in which drop-down menus provide access to all the product's features. This allows extra information, such as 'last action taken' or information on viruses detected, to be displayed. *Central Point Anti-Virus* occupied 1.87 Mbytes of space on my hard disk.

Central Point Anti-Virus was provided only on 1.44 Mbyte floppy disks (two of them), and the program specifically requires a high density floppy disk drive, leaving those users with older machines stuck. [*Central Point claims that the majority of its clients have 286 or above machines. If a corporate is still using XTs with no high density drive, they can either adopt a 'sheep dip' policy, or install version 1.x of CPAV on the machine. Ed.*].

Previous reviews of *Central Point Anti-Virus* have variously described the documentation as 'professionally produced', 'uninspiring', and 'very readable'. It is all of these, and nothing much has changed in version 2.0. The documentation provided with *CPAV* contains very few technical details, and with the exception of the chapter entitled 'Troubleshooting', would be useless when problems are encountered. In any manual it is difficult to judge the trade-off between technical content and readability, but I feel that *Central Point* should be able to achieve a better balance then this. This deficiency is exacerbated by the fact that many of the options are not indexed properly. The on-line help is quite good, but digging for specific information is rather hard.

There are a few specific points in the documentation with which I disagree. A Trojan Horse program is described as being 'one type of virus'. I thought that a necessary condition for a computer program to be called a virus was that it must be capable of replicating?

With regard to signature files from previous versions, the documentation states that 'You may delete any previous CHKLIST.CPS files on your system'. This is rather misleading. At first I thought that this would require the user to wander through every sub-directory on the machine deleting files, but later on in the manual the user is informed that it is possible to get *CPAV* to carry out this task.



```
                    Central Point Anti-Virus                      21:11

                          Express Menu

     Detect                         Full Menus

     Detect & Clean             • This option switches to a tree
                                  and file list display with
     Select new drive             commands on pull-down menus.

     Full Menus                 • If password protection was
                                  selected, you will be asked to
     Exit                         supply the password before
                                  continuing.

                                    Work Drive:         C:
     Central Point Software         Last Virus Found:   None
     ▶ ▶ Anti-Virus                 Last Action:        None

♥Help  ♦Drive  ♣Exit  ♠Detect  ‖Clean      δLog   ♀Menus  ╖List
```

CPAV 2.0 can be installed in one of two different types of configuration, in order to satisfy different levels of user sophistication.

Virus Detection

The scanner program and the integrity testing are integrated and maintain what *Central Point* calls 'SmartCheck' files in each subdirectory. An option is available to remake these files under user control, but most unfortunately, when they are remade, *Central Point Anti-Virus* always seems to mark the files (all called SMARTCHK.CPS) with the date that the software was installed, rather than the actual date of file creation. This is particularly irksome as it means that it is impossible to tell whether the files have been remade. [*Central Point informs VB that this is not how the program is intended to operate and is looking into this report. Ed.*]

The accuracy of virus detection by *Central Point Anti-Virus* has always been reported by *VB* as reasonable, but it is somewhat irritating to see viruses that were reported 6 months ago as undetected still being missed. Still worse, there are also some viruses which *have* been detected by a previous version, which are no longer detected by v2.0.

> *"I would contend that this result is risible in the extreme. An expert system? I think not."*

The version of *Central Point Anti-Virus* included with the *PC Tools* package reviewed in the January 93 edition of *VB* detected all but four (Kamikaze, Rat and two samples of the Amstrad virus) of the viruses that it was tested against. Six months on, and against a slightly extended test-set, *Central Point Anti-Virus* still failed to detect all of these viruses, as well as failing to detect one sample each of Vienna, Dos Hunter, Pitch, Power Pump, Todor and Tremor. This gives a scanning accuracy of 96%, missing 10 out of 228 virus samples. [*Central Point claims that it is not aware of any problems with the new scanner but will look into it. Ed.*]

When tested against 1024 samples of the Mutation Engine, *Central Point Anti-Virus* detected 221 out of the first 256 samples (86%), and always locked up when 255 infected files had been detected. This is the third review to point out this 'lockup' problem, but it is still there. I had to measure the detection rate by recovering remnants of report files using the *MS-DOS* program CHKDSK. This should not be necessary [*Central Point claims that it has not been able to recreate this bug, and is looking into the problem. Ed.*]. It is illuminating to note that the Mutation Engine detection rate has got worse, having previously been 92%.

My worries about the above results do not centre on the overall scanning accuracy (which is good), but upon *Central Point's* quality control. How can a scanner detect a sample in one issue of the program, and then in a later issue fail to detect the same sample? How can its detection rate get worse? This is especially worrying when the viruses concerned are not particularly difficult to detect.

Variable Speed Scanning

The rate at which scanning can be performed got me thoroughly confused as it seems so variable. The first time that I requested a scan of my hard disk, using the default options, *Central Point Anti-Virus* took 16 minutes 47 seconds to complete the scan. This hard disk contains 711 files which occupy 379 Mbytes, spread across 25 subdirectories - not a particularly large drive.

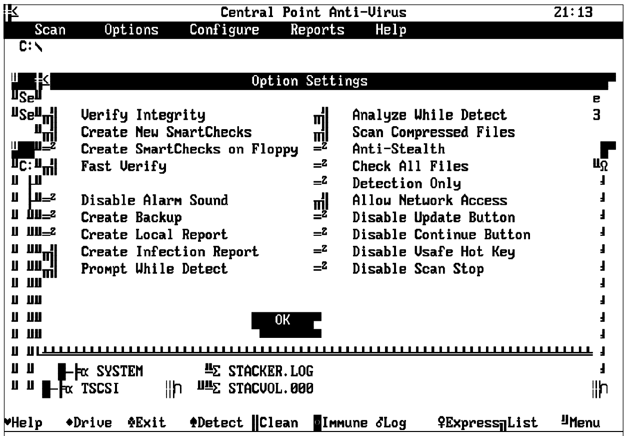
With the 'Scan compressed files' option switched off, subsequent scans of this drive took 15 seconds, 19 seconds and 32 seconds. Note that these varying times were to scan the same drive using exactly the same configuration. Why do they vary? I tried for a long time (all of one afternoon) to get to the bottom of this variation and failed. During this investigation, at one point I decided to start again and remake the *SmartCheck* files. This took 1 hour 53 minutes.

Note that one would expect the scan times to vary if different options were selected within the program. The first scan of the drive involves not only scanning every file on the disk, but also creating a set of *SmartCheck* files - however different times were recorded for the *same* settings. I tried removing all disk caching software from memory, thinking that the contents of the cache varying between scans, but this did not help. *Central Point* has stated that it cannot reproduce these variable scan times, but is looking into it.

By way of comparison, *Dr Solomon's Anti-Virus Toolkit* took 2 minutes 22 seconds to scan the same hard disk. *Sweep* from *Sophos* took 7 minutes 57 seconds to perform a full scan, and 3 minutes 2 seconds to perform a quick scan.

The scanning time for *Dr Solomon's Anti-Virus Toolkit* reduced to 40 seconds when the *Central Point* memory-resident software was disabled. Therefore the scanning time had been increased by 255% by the memory-resident software. Under similar conditions, the time taken by *Sweep* to perform a quick scan time reduced to 1 minute 19 seconds (an overhead of 130%), but only to 6 minutes 10 seconds for a complete scan (an overhead of 29%).

It is interesting to note that these timings all show increases of approximately 100 seconds added by *CPAV*. I would humbly suggest that the overhead introduced by this memory-resident software is unacceptable when large amounts of file access are involved. The manual discusses the amount of memory required, but does not discuss the overhead introduced.



```
                     Central Point Anti-Virus                21:13
    Scan     Options    Configure    Reports    Help
 C:\
                          Option Settings
      Verify Integrity                   Analyze While Detect
      Create New SmartChecks             Scan Compressed Files
      Create SmartChecks on Floppy       Anti-Stealth
      Fast Verify                        Check All Files
                                         Detection Only
      Disable Alarm Sound                Allow Network Access
      Create Backup                      Disable Update Button
      Create Local Report                Disable Continue Button
      Create Infection Report            Disable Vsafe Hot Key
      Prompt While Detect                Disable Scan Stop

                           OK

        SYSTEM              STACKER.LOG
       TSCSI                STACVOL.000
Help  Drive  Exit  Detect  Clean  Immune  Log  Express List  Menu
```

CPAV appears to redefine part of the character set in order to beautify its user interface further. However, on my machine, this resulted in screens which looked like this when running *Windows*.

The 'Virus Analyser'

Scanning is now augmented by the new feature of 'Analysing files for virus infection'. This is offered as an 'expert system' which looks at files to see whether they contain typical virus code. The best way to test such a system is to let it inspect the set of real viruses used for testing, and see what it comes up with. Unfortunately, the virus analyser could only detect 12 out of the 228 viruses in my test-set - a mere 5%. For the record, the analyser spotted files that were infected with 1260, Casper, Flip (x2), Virus-101 (x2), Whale, WinVir14, Maltese Amoeba, Spanish Telecom, Tequila, and V2P6. I would contend that this result is risible in the extreme. An expert system? I think not.

[*Central Point is less than pleased with the results of this test, and states that the virus analyser has achieved high scores against other test-sets. In addition,it says the analyser is not intended to replace other methods of virus detection, merely to augment them. However, Virus Bulletin can only rate a product on the results obtained against the VB test-sets. Ed.*]

Added Integrity

In its description of 'smart signatures', the manual now says that there is 'a database of statistics about each executable file's size, attributes, date, time and checksum'. This differs from previous versions in that it includes the word 'checksum'. The other addition to this part of the product is that *CPAV 2.0* can attempt to disinfect files which have been infected by an unknown virus, by utilising the data stored within the *SmartCheck* checksum files.

As a test, I created two files which were identical except for one byte. I checksummed the two files, and then copied one over the other. The result was that *Central Point Anti-Virus* did not detect that anything had changed, yet the content of second file was different. The manual does not give any inkling of what the 'checksum' really is, which is poor to say the least. I stated in a previous review that 'other anti-virus programs cope with checksumming much better', and there is nothing within *CPAV v2.0* to make me alter this conclusion. *Central Point* has explained that when a file is altered, the changes made are analysed heuristically to determine whether they are due to a virus infection or not. If this is the case, it should be explained in the manual so that it is possible to evaluate what the program actually does.

Immune to Problems

I am not going to discuss the Immunization features offered by *Central Point Anti-Virus*, as I do not believe that users should get into the habit of altering their executable files. Making a change to an executable file does not seem to be a particularly clever tactic as it may well introduce subtle faults. Even if my advice is ignored, beware that *Central Point Anti-Virus's* immunization may be quite hard to comprehend if you are non-technical.

My previous review pointed out that the process of immunization was not fully explained in the documentation, which merely states that 'Once immunized, a file has its own anti-virus capabilities', and an 'immunized file can ''heal'' itself, returning to its original state'. Neither of these statements would win prizes for technical detail, and they are both still there, completely unchanged, in this version.

Conclusions

I find it particularly worrying that some of the problems discussed within this review have been reported before and are still extant. The documentation still has little technical content and the virus detection seems to have got worse.

Last but not least, when tested against multiple samples of the Mutation Engine, *Central Point Anti-Virus* consistently locks up. This fault was exhibited previously, has been reported in *two* previous reviews by *VB*, and is *still* present. Don't they listen?

Many of the problems described in this review are encompassed by the phrase 'Quality Control'. I would contend that this is lacking in v2.0 of *Central Point Anti-Virus.* The poor scores by both the virus analyser and the scanner itself give cause for concern - especially as the detection rates are worse than those for previous offerings by *Central Point.*

On the plus side, the product can now communicate with its server-based bigger brother, offers compressed file scanning, a good user interface and the ability to detect some unknown viruses. It is up to the users to decide whether this is enough.

Technical Details

Product: *Central Point Anti-Virus*

Developer: *Central Point Software*, 15220 N. W. Greenbrier Pkwy, Suite 200, Beaverton, Oregon 97006-5798, Tel: +(503) 690-8080, Fax: +(503) 690-7133.

Availability: A PC, PS/2 or IBM compatible using a 286 processor (or better), with DOS v3.30 or higher, 640 Kbytes of RAM, 2 Mbytes of hard disk space, and either a 1.44 Mbyte (3.5 inch) or 1.2 Mbyte (5.25 inch) floppy disk drive. If so desired, *Central Point Anti-Virus* can operate under *Windows 3.x*, and/or *Novell Netware* v2.12 or later.

Version evaluated: 2.0

Serial number: None visible

Price: £99 with four quarterly updates.

Hardware used: a) *Toshiba 3100SX*, a 16MHz 386 laptop, with 5 Mbytes of RAM, one 3.5 inch (1.44M) floppy disk drive, and a 40 Mbyte hard disk, running under *MS-DOS* v5.0. (b) 4.77MHz 8088, with one 3.5 inch (720K) floppy disk drive, two 5.25 inch (360K) floppy disk drives, and a 32 Mbyte hard card, running under *MS-DOS* v3.30

The Test-Set

This suite of 143 unique viruses (according to the virus naming convention employed by *VB*), spread across 228 individual virus samples, is the current standard test set. A specific test is also made against 1024 viruses generated by the Mutation Engine (which are particularly difficult to detect with certainty).

The test set contains 6 boot sector viruses (Brain, Form, Italian, Michelangelo, New Zealand 2, Spanish Telecom), and 218 samples of 138 parasitic viruses (nb Spanish Telecom appears in both lists). There is more than one example of many of the viruses, ranging up to 12 different variants in the case of the Tiny virus. The parasitic viruses used for testing are listed below. Where more than one variant of a virus is available, the number of examples of each virus is shown in brackets. For a complete explanation of each virus, and the nomenclature used, please refer to the list of PC viruses published regularly in *VB*:

1049, 1260, 12 TRICKS, 1575, 1600, 2100 (2), 2144 (2), 405, 417, 492, 4K (2), 5120, 516, 600, 696, 707, 777, 800, 8 TUNES, 905, 948, AIDS, AIDS II, Alabama, Ambulance, Amoeba (2), Amstrad (2), Anthrax (2), AntiCAD (2), Anti-Pascal (5), Armagedon, Attention, Bebe, Blood, Burger (3), Captain Trips (2), Cascade (2), Casper, Dark Avenger, Darth Vader (3), Datalock (2), Datacrime, Datacrime II (2), December 24th, Destructor, Diamond (2), Dir, Diskjeb, Doshunter, Dot Killer, Durban, Eddie, Eddie 2, Fellowship, Fish 6 (2), Flash, Flip (2), Fu Manchu (2), Hallochen, Helloween (2), Hymn (2), Icelandic (3), Internal, Itavir, Jerusalem (2), Jocker, Jo-Jo, July 13th, Kamikaze, Kemerovo, Kennedy, Keypress (2), Lehigh, Liberty (5), LoveChild, Lozinsky, Macho (2), Maltese Amoeba, MIX1 (2), MLTI, Monxla, Murphy (2), Necropolis, Nina, Nomenklatura (2), Number of the Beast (5), Oropax, Parity, PcVrsDs(2), Perfume, Pitch, Piter, Polish 217, Power Pump, Pretoria, Prudents, Rat, Shake, Slow, Spanish Telecom (2), Spanz, Subliminal, Sunday (2), Suomi, Suriv 1.01, Suriv 2.01, SVC (2), Sverdlov (2), Svir, Sylvia, Syslock, Taiwan (2), Tequila, Terror, Tiny (12), Todor, Traceback (2), Tremor, TUQ, Turbo 488, Typo, V2P6, Vacsina (8), Vcomm (2), VFSI, Victor, Vienna (8), Violator, Virdem, Virus-101 (2), Virus-90, Voronezh (2), VP, V-1, W13 (2), WinVirus 14, Whale, Yankee (7), Zero Bug.

COMPARATIVE REVIEW

Mark Hamilton

OS/2 Virus Protection

The new 32-bit operating systems from *IBM* and *Microsoft* will, both companies claim, revolutionise personal computing as we know it. *Microsoft's Windows NT* - the letters 'NT' standing for 'New Technology', 'Not There', 'No Takers' or 'No Thanks', depending on your point of view - has yet to be released. Even when it eventually is, *Microsoft* admits that it will support only a small subset of the millions of programs and applications that have been written for the PC over the last ten or so years.

OS/2, on the other hand, really became usable last year, when *IBM* launched version 2.0, and version 2.1 is now on general release. It supports over 90 percent of all applications out in userland and is starting to prove itself as an industrial strength operating system.

Although *OS/2* has not really caught on among the private users, several of the larger corporates have adopted it as their operating system of choice, and it is this carrot which has lured some of the big names in the anti-virus community onto the platform. To date, four different companies have announced the release of *OS/2* specific versions of their anti-virus software: *IBM, Sophos, S&S International* and *McAfee Associates.*

Featured Features

At the time of writing, there are no known *OS/2*-specific computer viruses. However, research shows that DOS file-infecting viruses are capable of infected files in DOS or *Windows* (*Win-OS/2*) sessions, although those which have low-level trigger effects will have their destructive attempts thwarted. In addition, boot sector viruses are to a certain extent platform-independent, and those users who use their *OS/2* machine as a file server (for example, sites using *LAN Manager*) also require a method of ensuring that the contents of the server are virus-free.

For the anti-virus company, *OS/2* provides certain benefits: the need to employ special anti-stealth tactics disappears, their product is operating in a protected environment and they have access to linear memory, rather than the segmented memory DOS imposes. For the users, the benefits are no less tangible: one should be able to run these products so that they check files in the background meaning that users can get on with real work and not have to 'play' at detecting viruses. *OS/2* provides the opportunity of relegating anti-virus software back to the utility category where pretty interfaces requiring human interaction are unnecessary: how many companies, I wonder, will rise to that challenge?

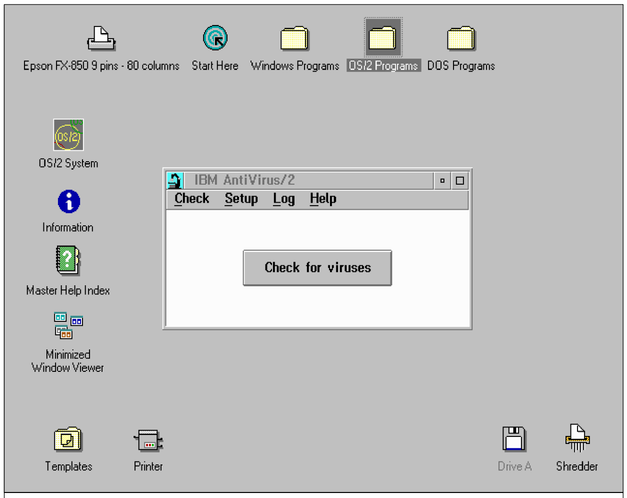
The following review examines the *OS/2* products from the four companies mentioned above, and forms *Virus Bulletin's* first ever comparative *OS/2* review.

IBM AntiVirus/2 - Version 1.02

IBM releases its product on a quarterly basis and this is the third such release. In designing this product, *IBM* went for the 'install and forget' philosophy - the user is not required to do anything, unless the product detects a virus.

My major criticism of this product is that, in the United Kingdom at least, it is not currently available as a shrink-wrapped product. With all other *IBM* hardware and software products, you simply send in a card to register for the *IBM Helpline* which is available 24 hours a day, 365 days a year - and an excellent service it is too. But not with *IBM Anti-Virus*: you have to subscribe to a special service and download the software from an *IBM* Bulletin Board. This is likely to restrict *IBM's* market share for *AntiVirus/2* which, like its DOS counterpart, could so easily be integrally bundled with the operating system.

The installation process is simple and efficient and it gives the user the option of invoking a DOS-based anti-virus TSR every time a DOS or *Win-OS/2* session is invoked. The software can also be configured to scan the files at periodic intervals: every boot, every day, once a week, once a month or only when specifically executed.



IBM AntiVirus/2 appears to be the only program which takes full advantage of the multitude of new features offered by *OS/2*.

Product	Version Evaluated	In The Wild Test-set (99)	Standard Test-set (364)	Mutation Engine Test-set (1536)
IBM Anti-Virus/2	1.02	100%	100%	99.9%
Sophos Sweep For OS/2	2.51	100%	100%	100%
Dr Solomon's Anti-virus Toolkit For OS/2	6.53	98%	98%	100%
McAfee Associates OS/2 Scan	106	97%	99%	100%

Didn’t they do well! All the products scored well in the detection tests, although with products from these manufacturers one would expect 100% scores in all tests. A question of poor quality control?

IBM AntiVirus/2 works in a different manner to the other products tested in that it includes both an integrity checker as well as a virus scanner. When it checks files, it looks to see if they have changed in any way and only scans those files which have been modified or are not included in the database for viruses. This makes it quite fast in operation since it does not need to wade through its virus signature database for every file - just those it finds suspicious. When it finds an infected file, it is capable of disinfecting it, as long as the file is infected with a virus which*IBM* deems common. Given that most virus infections are caused by a tiny minority of viruses, this restricted disinfection list is unlikely to cause any problems.

The on-line help system lists all the viruses detected by the current release as well as providing a brief description of the most common ones. Whilst this is not as comprehensive as Patricia Hoffman’s*VSUM* database, it is nevertheless accurate and concise.

I do not have many criticisms with the*IBM* product. It is both fast and accurate, and I like the idea of combining an integrity checker with a scanner. One annoying quirk is that it insists on searching the whole disk before it begins virus checking to discover how many files it needs to check. Why?

The only other criticism I have of the product is its update frequency - all the others are updated every four to six weeks; *IBM Anti-virus /2* is updated on a quarterly basis. In a fast moving field, is that enough?

Dr Solomon’s Anti-Virus Toolkit for OS/2 - Version 6.53

The *OS/2* specific version of*S&S’s* popular utility actually consists of the DOS version, an extra diskette and a very slim appendix for the DOS manual which contains details of the *OS/2* specific parts of the product.

Unfortunately, the documentation has not kept up with the software and the small card entitled ‘Installing the*OS/2* Anti-Virus Toolkit’ contains a completely fallacious set of installation instructions. There is even a typographical error on the card - it says ‘OS\2’.

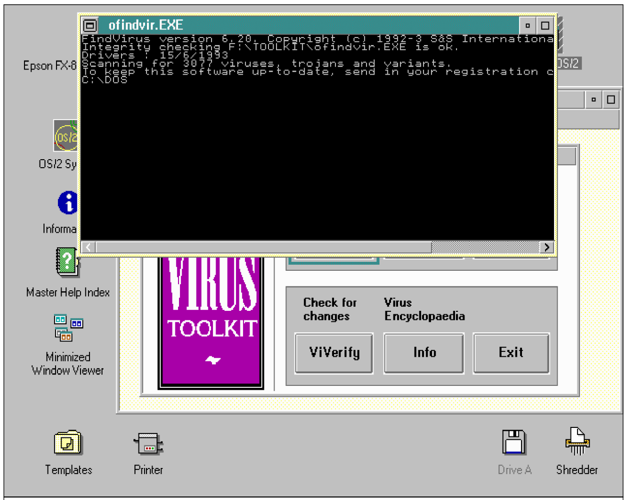
What the instructions should say - and what you in fact do - is run the setup program provided on the disk to decompress and install the *OS/2 Toolkit* (rather than copy the files from the floppy as the instructions would have you believe).

The *OS/2 Toolkit* comprises just four programs: VISION, the *OS/2* front-end menu program and equivalent to the DOS TOOLKIT program; OVIVERIF, an integrity checker; OFINDVIR, the scanner; and VDISPLAY, the on-line version of the virus encyclopedia. You can, if you wish, also install the DOS version although the only possible item you might want from it would be GUARD, the monitor, to protect your DOS and*Win-OS/2* sessions.

The setup program creates a new folder on the Workplace Shell Desktop - though I wish it had asked me first - which curiously contains just the front-end program, VISION. This program - in common with the DOS and*Windows* versions - has three big buttons to scan drives A, B and C. If a machine has more drives than that, it is not possible to use VISION, and the user has to run OFINDVIR manually.

The programs themselves are rather disappointing as (with the exception of VISION) they do not take advantage of the *OS/2* graphical environment, running only in text mode.

The OS/2 Toolkit has none of the scheduling niceties found in the *IBM* product and surprisingly did not fare particularly well in the detection tests. It missed both Tremor and V2P6 in the ‘In the Wild‘ test-set and was slower than the*IBM* product in two out of the three speed trials. When checking all files, it had problems with the Extended Attributes file



```
FindVirus version 6.20, Copyright (c) 1992-3 S&S Internationa
Integrity checking F:\TOOLKIT\ofindvir.EXE is ok.
Drivers : 15/6/1993
Scanning for 3077 viruses, trojans and variants.
To keep this software up-to-date, send in your registration c
C:\DOS
```

Even though *Dr Solomon's Toolkit* has a graphical user interface, the actual scanning is carried out in a window which looks not unlike the DOS version of FINDVIR

and 'SWAPPER.DAT' (the swap file) - it really ought to know about the significance of these files, and be able to deal with their presence gracefully.

VDISPLAY, the on-line virus database (or encyclopedia as *S&S* calls it) is, like the other elements in this package, an *OS/2* text mode application. One factor I found particularly annoying is that it does not recognise the mouse. Users would normally access this through the *Toolkit* front end, which is mouse driven. However, the encyclopedia is displayed as an *OS/2* windowed application in text mode, so the user is relegated back to keyboard control. *S&S* might find that reconstructing the encyclopedia as an *OS/2* help file might alleviate this problem and make the whole package appear to be more of a real *OS/2* application.

I am somewhat disappointed with this offering from *S&S* - particularly when considering the high standard of many of its other products. The detection results are surprising, and the unimaginative use of the *OS/2* interface makes the overall result rather uninspiring.

Sweep for OS/2 - Version 2.51

The *Sophos* product is a command line program and appears to be a recompilation of its DOS product. Unlike its DOS counterpart, it has as yet no front-end menu to make life easier and the various different options within *Sweep* are controlled by a plethora of command line parameters.

The diskette contains just the *Sweep* executable and its signature file - there is no installation routine provided.

Like the *S&S* product, *Sweep* had problems opening the Extended Attributes file and it shared the *Toolkit's* problem concerning the *OS/2* swap file. It also reported a spurious 'DOS error code'.

The documentation is a little misleading in places, and looks like a rewrite of the DOS *Sweep* manual. For example, it states 'SWEEP may be incorporated into the AUTOEXEC.BAT file...'. Following these instructions produces an error message whenever a DOS or *Win-OS/2* session begins. This is because the AUTOEXEC file is run under the DOS emulator, which does not understand *OS/2* executables. The manual should read 'SWEEP may be incorporated into the STARTUP.CMD file...'.

Sweep's operation is controlled by both command line parameters and a configuration file that stipulates which areas should be included in the scan. This latter file, SWEEP.ARE, was designed by systems programmers for the cogniscenti and not dumb users. It can be somewhat intimidating to set up, until you realise that '80' refers to the first hard drive and '81' to the second (physical) hard drive.

Like its DOS counterpart, the *OS/2 Sweep* does not disinfect infected files but it can optionally delete them. When *Sweep* discovers a virus, it turns the screen background colour to a vivid red and flashes 'Sweep Alarm!' on the screen.

Sweep proved to be the slowest of the four products tested as it took over three minutes to scan 649 executable files. Nevertheless, *Sweep* discovered all the viruses... speed isn't everything, but would be nice!

McAfee Associates OS/2 Scan - Version 9.17 V106

McAfee Associates OS/2 Scan is just like the company's other shareware scanners in that it presents a 'no frills' approach to the task. Like *Sophos*' and *S&S's* products, *McAfee's OS2Scan* looks like a simple recompilation of the DOS version's source code. Indeed, in *McAfee's* case, the developers have even left the SCAN name in an error message ('Type SCAN /help').

OS/2 Scan comes with no fancy installation routine - but really does not require one. Installation is simply a matter of copying the single executable OS2SCAN onto the hard drive. The product also comes with a documentation file, OSCNnnn.DOC, which explains how OS2SCAN is designed to be used.

OS2SCAN has no pretty GUI and is controlled by command line parameters. The switches used are a subset of those understood by SCAN, so regular*McAfee* users will have no problems converting existing batch files and modes of operation to the*OS/2* platform

Although quite fast,*McAfee's OS2SCAN* is not one of the most accurate scanners: it missed infections of Loren, Powerpump and Whale in the 'In The Wild' test-set - indeed it missed more viruses than the other three products making it the least reliable of those reviewed.

Like its DOS counterparts,*McAfee's OS/2* scanner is shareware which might make it an unacceptable choice for more traditionally minded business who prefer to purchase their software through conventional sources.

Speed Tests

Due to the multi-tasking nature of*OS/2*, the speed tests are rather less important than they would be for a DOS machine. However, for reference, they are given below:

Product	Diskette	HD (Turbo)	HD (All)
IBM Anti-Virus/2	0:45	0:53	2:15
Sophos Sweep	1:56	3:41	8:41
Dr Solomon's AVTK For OS/2	0:52	0:56	1:59
McAfee OS/2 Scan	0:35	1:13	4:20

Conclusion

It is early days for*OS/2* scanners and the anti-virus companies are only just starting to dip their development toes in the water. If I were actively looking to purchase an*OS/2* hosted anti-virus product, I would be disappointed by the lack of choice and, except in one case, the lack of attention paid by the anti-virus developers to*OS/2's* potential. Only one product,*IBM AntiVirus/2,* offers scheduling and, from a dumb user's point of view, seems better integrated into the environment and makes full use of its facilities.

Needless to say,*IBM's* product is the only one I didn't erase from my hard drive following the testing phase of this review - the other products don't really hit the mark. It is interesting to note that some of the other products seem to perform less well than their DOS counterparts. With reviews of*OS/2* products being a relatively new feature within*Virus Bulletin* it will be interesting to see if those caught on the hop improve their scores next time.

Vendor Details

Product: *OS/2 Scan*

Vendor: McAfee Associates, 2710 Walsh Ave., Suite 200, Santa Clara, CA 95051-0963, USA.

Price: $375, including *OS/2 Clean* and *Vshield*.

Product: *IBM AntiVirus/2*

Vendor: IBM AntiVirus Services, 1 East Kirkwood Boulevard, Roanoke, TX 76299-0015, USA

Price: $29.95, but prices vary according to country.

Product: *Sophos Sweep for OS/2*

Vendor: Sophos Ltd, 21 The Quadrant, Abingdon Science Park, Abingdon, Oxon. OX14 3YS

Price: £295 for Single User, £495 for the File Server, with monthly upgardes

Product: *Dr Solomon's Anti-virus Toolkit for OS/2*

Vendor: S&S International Ltd, Berkley Court, Mill Street, Berkhamstead, Herts. HP4 2HB.

Price: £149 +VAT with quarterly upgrades.

Technical Details:

Test Platform:

I used a *SIR 486DX50* with 8MB memory, 170MB IDE hard drive and a *Panasonic* Phase-Change Optical drive provided a further half gigabyte of disk storage. The machine was running under *OS/2* version 2.1 DAP (Developer Assistance Programme) release.

The IDE drive was used for the speed tests and there were 2,732 files, split across 126 directories, occupying 123,081,356 bytes of which there were 649 executable files occupying 46,236,452 bytes. For the diskette test, I used the *OS/2* Disk 1 which contains 52 files occupying 1,435,106 bytes.

Test-Sets Used:

Only the Common, or 'In The Wild', test-set has been updated, and now contains the following 99 viruses:

File Infectors: 1575, 2100 (C+E), 4k (C+E), 777, AntiCAD (C+E), Captain Trips (C+E), Cascade 1701, Cascade 1704, Dark Avenger (C+E), Datalock (C+E), Dir-II, Dos Hunter, Eddie, Eddie 2 (C+E), Father (C+E), Flip (C+E), Hallochen, Helloween (C+E), Invader (C+E), Jerusalem 1 (C+E), Keypress (C+E), Liberty (C+E), Liberty- E, Loren (C+E), Maltese Amoeba, Mystic, Necropolis, Necros (C+E), Nomenklatura (C+E), Nothing, PcVrsDs (C+E), Penza, Pitch, Powerpump, SBC, Slow (C+E), Spanish Telecom 1, Spanish Telecom 2, Spanz, Syslock, Tequila, Todor, Tremor, V2P6, Vacsina, Vienna 2a, Vienna 2b, Virdem, Virus, W13a, W13b, Warrier, Warrior, Whale, Winvir14, Yankee 1 and Yankee 2.

Boot Sector Infectors: Aircop, Beijing, Brain, Disk Killer, Form, Italian Generic A, Joshi, Korea A, Michelangelo, New Zealand 2, NoInt, Spanish Telecom and Tequila.

For details of the other *Virus Bulletin* Test-sets, please consult *Virus Bulletin*, May 1992, page 23.

END-NOTES AND NEWS

3rd International Virus Bulletin Conference, 9th-10th September 1993, Amsterdam, The Netherlands. Contact Petra Duffield. Tel. +44 235 531889

***Novell* has announced its plans to provide enhanced network security** under *Netware 4.0*. The company is attempting to lead a worldwide contingent of customers, industry partners and security experts to increase the level of security provided by *NetWare*. This news was released simultaneously with the news that *Novell* has submitted *NetWare 4.0* to the *National Computer Security Centre* (*NCSC*) for evaluation under *C2* security standards. ‘This announcement has significant impact for security-minded commercial and government customers,’ said Jan Newman, executive vice president for *Novell’s NetWare* Systems Group. ‘*Novell*, with the help of its partners and customers, can build on the security foundation of *NetWare* to offer an open, affordable and trusted network computing environment.’

Typo of the month comes from the July edition of *Computer Shopper* (a UK computer magazine). The August edition puts things right: ‘In last month’s Labs report on virus detection the ‘‘In the wild’’ percentage score for *Untouchable* should read 95.3% and not 5.3% as printed.’ Phew.

Excessive claims: According to a report in *PC Week*, *Central Point’s* claims about the new features in *CPAV 2.0* are doubted by experts within the industry. [*For a complete review of CPAV 2.0 see pp.16-19. Ed.*] Bryan Clough, co-author of the book *Approaching Zero*, claims ‘This is just one of several claims from people who are just trying to impress with claims of something new. Reliable tests show these are usually just outrageous claims from salesmen.’

Abacus, the company which brought users *DOS 5.0 Complete* and *Laser Power Tools*, is proud to unveil its latest publication: *Puzzles, Pranks and Games For Windows*. The book comes with a disk which contains 20 guaranteed non-productive (unless you are a practical joker) utilities which can be installed on unwitting colleagues’ machines. A typical prank would be to install UAE. Periodically the all-too-familiar UAE (Unrecoverable Application Error) message is placed on the screen, and system execution is halted. Everything looks like the real error message... until you spot that the text on the button says ‘Not!’ All harmless fun... or is it. Does the book constitute ‘Incitement to commit an offence under section 3 of the *Computer Misuse Act*’? Only time will tell...

Italy suffers more than any other European country from computer viruses and is amongst the top five victims of virus attack in the world, according to a report in the Italian daily *Il Corriere della Sera*. This is despite spending an estimated 62 *billion* lire a year on virus prevention.

Patricia Hoffman’s *VSUM* ratings for July: 1. *Command Software’s F-Prot Professional 2.09*, 96.4%, 2. *McAfee Associates ViruScan V106*, 95.9%, 3. *SafetyNet’s VirusNet 2.08a*, 92.5%, 4. *Dr Solomon’s AVTK 6.53*, 89.9%, 5. *Sophos Sweep 2.48*, 86.1%. **NLMS:** 1. *McAfee Associates NetShield 1.5V104*, 90.5%, 2. *Sophos’ Sweep NLM* 2.48a, 86.4%, 3. *Command Software’s Net-Prot 1.00s*, 71.8%, 4. *Cheyenne’s Inoculan 2.0/2.18g*, 69.4%.

Patricia Hoffman’s *VSUM* ratings for June: 1. *McAfee Associates ViruScan V105*, 97.4%, 2. *SafetyNet’s VirusNet 2.08a*, *Frisk Software’s F-Prot 2.08* 93.0%, 4. *Dr Solomon’s AVTK 6.51*, 89.9%, 5. *Sophos Sweep 2.48, 87.2%*. **NLMS:** 1. *McAfee Associates NetShield 1.5V104*, 92.2%, 2. *Sophos Sweep NLM 2.48a*, 87.5%, 3. *Cheyenne’s Inoculan 2.0/2.18g*, 71.0%, 4. *Intel’s LANProtect 1.53+1/93S*, 56.1%.

VIRUS BULLETIN

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 Science Park, Abingdon,
OX14 3YS, England

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

US subscriptions only:

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

Tel 203 431 8720, Fax 203 431 8165

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.