ISSN 0956-9979 DECEMBER 1998

VIRUS BULLETIN

THE INTERNATIONAL PUBLICATION ON COMPUTER VIRUS PREVENTION, RECOGNITION AND REMOVAL

IN THIS ISSUE:

• **GameBoy:** He's come a long way since an encounter with Stoned on one of his beloved computer games. He is now *Data Fellows'* Lead Virus Analyst. Read Péter Ször's story on p.6.

• **Head to head to head:** At VB'98 the hot topic was the debate about the *WildList Organization* supplying reviewers with virus samples. Peter Morley, Vesselin Bontchev and Shane Coursen lock horns, starting on p.8.

• **That's a first!** A brief look at HTML viruses is followed by this month's virus analysis on the first polymorphic *Excel* macro virus. Meet XM/Compat on p.15.

CONTENTS

EDITORIAL

Clean Me Up, Scotty!

During a session at VB'98, an observation by the presenter gave rise to an interesting discussion during the question and answer session. Hmmm – that did not come out too well… I'm sure it happened in more than one session, but a particular instance inspired this column.

The observation that triggered the discussion was that the addition of 'auto-disinfection' or 'on-the-fly disinfection' of macro viruses to his company's product had been a great boon to his customers. There were several reasons for this, but the main one was that macro viruses account for by far the largest proportion of gateway or desktop detections, and as they are easily disinfected, automating that step saved a lot of work for the customer.

"… urgent need for yet another configuration option"

Talking, as I do, with many system administrators responsible for anti-virus software, it appears that the huge proliferation of macro viruses has meant that many organizations which had never experienced any form of virus incident before – let alone an in-house infection or LAN-wide outbreak – now see many interceptions per week in their centralized logs. In fact some, who a few years ago were deeply worried about the 'digital scourge' computer viruses were painted as, now see so many macro viruses intercepted and cleaned at the 'company borders', that they have stopped looking at the logs altogether. Some have even turned such logging off!

On reflection, this should probably be neither surprising nor, to many, worrying. The attitude seems to be that macro viruses are more of a nuisance than a problem. 'Sure, get rid of them and prevent them getting loose inside my company so we don't send infected files out and upset anyone else – that looks bad – but these viruses don't really cause any trouble.' Fictional, yes, but how many of you recognize elements of an organization you are familiar with there?

It is increasingly common to view macro viruses as maddening, but non-dangerous. Coupled with the burgeoning size and complexity of IT systems and the reputedly less-than-commensurate increase in support staff, I'd expect administrators to look for easy solutions. The speaker at VB'98 suggested that enabling auto-disinfect on gateway and/or desktop scanners be standard practice.

Such an approach certainly reduces the potential overload on administrators who would no longer have to manually check each 'suspect' file reported or quarantined on the network. It would reduce the hectic phone calls from people needing documents disinfected straightaway so they can complete some time-critical task. All round, it should be a good thing, right?

Well, a few months ago I probably would have answered 'Yes'. Now I see a fairly urgent need for yet another configuration option in anti-virus programs!

A few *Word* macro viruses modify document contents. One of the earliest, WM/Wazzu.A (and most of its variants), randomly moves a word in documents and inserts 'wazzu'. About a year ago, WM/Switcher.A appeared with its payload of swapping two numbers within documents.

Such 'data-diddling' may be stepping up, and – perhaps more worryingly – appearing in *Excel* macro viruses. In the grand scheme of things, an ungrammatical sentence or two and the occasional 'wazzu' might not be too great a problem, but spreadsheets where typically long chains of calculations, with chains (and even loops) of dependencies very quickly become meaningless gobs of data at the smallest unwanted change are another story. The likes of XM/Compat (see p.15) and a couple of recent Laroux variants with data-diddling payloads are most unwelcome arrivals.

So what does this have to do with automatic disinfection? You receive data from a partner company, in XLS files and (ultimately) make significant investment or purchase decisions based on it. Your email gateway finds and disinfects Compat. You receive this week's update without so much as a warning that something unwanted was there, let alone that it was a data-diddler and thus you now cannot trust the contents of the file.

Where's the 'quarantine data-diddlers, clean others' option when you need it?

NEWS

Taking it Personally

It can be entertaining to check the *Virus Bulletin* email. Often a source of interesting snippets, people looking for assistance or advice with computer virus issues, and the like, it also occasionally delivers surprises.

A string of such started on 29 October when a person unknown to the editor, apparently sent a message making a suggestive proposition to him. 'Something seemed odd about the message, so I just left it in my in-box', Nick recounted. That was a Thursday and by the end of that weekend several near-identical messages had been received, all from different people.

Whilst deciding on a diplomatic approach to enquire of the senders what might be behind these messages, another one arrived. This time it was closely followed by another from the same address. The second apologized for the first message and claimed it was sent by a suspected new *Word* macro virus the sender was trying to track down and eliminate from his workplace.

A sample was duly received and analysed. Known as W97M/ColdApe, it is a simple macro virus that drops a Visual Basic Script virus (see p.13) and a separate VBS script that the macro virus invokes. This script sends an email message to Nick and another with the infected machine's IP address to an anonymizing email redirector. Most major anti-virus products were updated to deal with this virus by the end of the first week in November ▮

Reflections on Rusting Trust

Following on from its UK advertisement consisting of an obituary marking the demise of *Dr Solomon's Software*, *Sophos* ran a new ad in November. Under the caption 'Toolkit seized?' the advertisement featured a large and heavily rust-encrusted wrench.

An obvious spoof on the *AVTK* trademark spanner logo, the advertisement continued with an invitation to potentially unhappy, former *Dr Solomon's* customers to evaluate *Sophos Anti-Virus*. While this is an interesting marketing angle, it is unlikely to feature on any T-shirts in the future!

In the normal course of events, *NAI* may not have been too concerned by this. Unfortunately for *NAI*, at about the same time the *Sophos* ad appeared in the UK, *Ingram Micro* ran its own advertisement for *NAI's* full Net Tools suite in the Strategy Guide supplement of VAR Business. This included a description of the *VirusScan* component of *NAI's Total Virus Defense* product as 'the most rusted virus detection and removal solution in the world.'

Virus Bulletin presumes that *Ingram Micro's* copywriters do not have a secret holding in *Sophos Plc* ▮

Prevalence Table – October 1998

Virus	Type	Incidents	Reports
Class	Macro	160	24.0%
Laroux	Macro	119	17.9%
Win95/CIH	File	36	5.4%
Cap	Macro	34	5.1%
Compat	Macro	30	4.5%
Groov	Macro	25	3.8%
Paix	Macro	22	3.3%
Wazzu	Macro	13	2.0%
Steroid	Macro	11	1.7%
AntiEXE	Boot	7	1.1%
Form	Boot	7	1.1%
Jedi	Macro	7	1.1%
Munch	Macro	7	1.1%
Cartman	Macro	6	0.9%
Marburg	File	6	0.9%
NoNo	Macro	6	0.9%
Npad	Macro	6	0.9%
NYB	Boot	6	0.9%
Appder	Macro	5	0.8%
Concept	Macro	5	0.8%
Extras	Macro	5	0.8%
Niceday	Macro	5	0.8%
Nottice	Macro	5	0.8%
Parity_Boot	Boot	5	0.8%
Ripper	Boot	5	0.8%
Win95/Fono	Multi-partite	5	0.8%
AntiCMOS	Boot	4	0.6%
Chack	Macro	4	0.6%
Cheval	File	4	0.6%
Hark	Macro	4	0.6%
MDMA	Macro	4	0.6%
Showoff	Macro	4	0.6%
Baph.1536	Multi-partite	3	0.5%
CopyCap	Macro	3	0.5%
DelCMOS	Boot	3	0.5%
mIRC/Gerre	File	3	0.5%
Sampo	Boot	3	0.5%
Shiver	Macro	3	0.5%
TWNO	Macro	3	0.5%
Others [1]		73	10.9%
Total		666	100%

[1] The Prevalence Table includes a total of 76 reports across 60 further viruses. Readers are reminded that a complete listing is posted at http:/www.virusbtn.com/Prevalence/.

IBM PC VIRUSES (UPDATE)

The following is a list of updates and amendments to the *Virus Bulletin Table of Known IBM PC Viruses* as of 15 November 1998. Each entry consists of the virus name, its aliases (if any) and the virus type. This is followed by a short description (if available) and a 24-byte hexadecimal search pattern to detect the presence of the virus with a disk utility or a dedicated scanner which contains a user-updatable pattern library.

Type Codes

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

Anna.737

CN: An encrypted, appending, 737-byte, direct infector which targets one file at a time. It contains the texts '[ANNA]', 'Slartibartfast, ARCV NuKE the French' and ' Have a Cool Yule from the ARcV xCept Anna Jones I hope you get run over by a Reindeer Santas bringin' you a Bomb All my Lurve - SLarTiBarTfAsT (c) ARcV 1992 - England Raining Again'. Infected files have the string 'ImIr8' at the end of their code.

```
Anna.737        B440 8B9C 3004 B9E1 028D 940E 01CD 21E8 D6FF E8C3 FFC3 496D
```

Anni

CN: Two encrypted, appending, fast, direct infectors. The 237-byte variant contains the plain text signature '[ANNI-VCS]' and the encrypted text '*.com'. The 253-byte variant has the text '*.com'. Infected files have their time-stamps set to 62 seconds.

```
Anni.237        8BEB 8DB6 2701 568B 96F2 01B9 6000 8BFE FCAD 33C2 ABE2 FAC3
Anni.253        8BEB 8DB6 3D01 568B 9618 02B9 6D00 8BFE FCAD 33C2 ABE2 FAC3
```

AMorph.367

CN: An overwriting, 367-byte, fast, direct infector with the encrypted text '*.com'.

```
AMorph.367      B800 25CD 210E 1F0E 07B4 1791 B457 2AE5 5A59 5BCD 00B4 3FBE
```

Barrotes.1292

CER: An appending, 1292-byte virus containing the text 'c:\command.com' and the encrypted message 'Iniciando Filo-Windows 95 Virus by ...'.

```
Barrotes.1292   B90C 05BA 0001 B440 CD21 7303 E948 012E 832E 1B01 0333 C933
```

Birgit.310

CN: An encrypted, appending, 310-byte, fast, direct infector containing the texts 'Birgit? Yvonne? YES!!' and '*.com'. Infected files have the byte 72h ('r') at offset 0003h.

```
Birgit.310      E817 005B B409 80C4 37B9 3601 8D96 0B01 CD21 E805 00E9 4EFF
```

Birgit.1000

CN: Two variants of an appending, 1000-byte virus which infects one file at a time (in a subdirectory only). They both contain the texts '[DVG]chklist.ms' and 'anti-vir.dat'. The .A variant also contains the text 'Birgit, you are irresponsabily.' and the .B variant 'Doctor Rave need Birgit!.'. Infected files have the double word 26252321h ('!#%&') at offset 0003h.

```
Birgit.1000.A   43E2 FA5B 53B4 40B9 E803 8D96 0E01 CD21 B801 573E 8B8E 2204
Birgit.1000.B   43E2 FA5B 53B4 40B9 E803 8D96 0E01 CD21 B801 573E 8B8E 3E04
```

Cffl.2560

EN: A 2560-byte, fast, direct infector with the encrypted texts 'Program too big to fit in memory', 'This vi', 'CFFL', 'Borsó' and 'You've become an hour older again!'.

```
Cffl.2560       B440 3E8B 8E10 068D 9600 01CD 213B C175 4F3E 8BB6 F805 81EE
```

Codebreaker.448

ER: An appending, 448-byte virus containing the texts '[Insert_Name]' and 'Sea4, CodeBreakers'. Infected files have the word C401h at offset 0012h.

```
Codebreaker.448 A3C8 0189 16C6 01B4 4099 B9C0 01CD 21B8 0042 33C9 99CD 21B4
```

CrazyPunk.500

CN: An encrypted, 500-byte, fast, direct infecting appender with the texts '(C) Crazy Punk' and '*.com'.

```
CrazyPunk.500   B9E5 000E 8DBC 2201 5733 DB8E DBC6 06F0 04CB 0E1F EAF0 0400
```

DieHard.4000.I

CER: A minor variant of the DieHard virus.

```
DieHard.4000.I  E800 005B 8D7F 5B0E FD07 AB8B C3B1 04D3 E840 8CCA 03C2 8BD0
```

Diw.286

CN: An appending, 286-byte, fast, direct infector containing the text '*.com'. Infected files have the byte 90h at offset 0003h.

```
Diw.286         B440 B91E 018B D7CD 21B8 0042 33C9 33D2 CD21 5E56 8B44 1A2D
```

DrRad.376

CN: An appending, 376-byte, fast, direct infector containing the texts 'AToM', 'v1.0', '[Dr. Radiaki]', '..[AToM]..' and '*.COM'.

```
DrRad.376       B978 018D 9608 01E8 B000 E89A 00E8 8700 8B0E 6802 8D96 E702
```

DrRad.456

CN: An appending, 456-byte, fast, direct infector containing the texts '[X-Worm] v1.0 by Dr. Radiaki \\ [UvC]', '*.COM', '*.RAR', '*.ARJ' and '*.ZIP'.

```
DrRad.456       B9C8 018D 9608 01E8 3700 B000 E828 00C6 86D9 02E9 8B86 3603
```

DrRad.580 **CN:** An encrypted, appending, 580-byte, fast, direct infector containing the texts '[AToM] v1.05 by Dr. Radiak // [UvC]' and '*.COM'.

```
DrRad.580          B944 028D 9608 01E8 60FF E813 FFB8 0042 33C9 2BD2 CD21 E807
```

Enmity.808 **CN:** An encrypted, appending 808-byte, fast, direct infector with the texts 'Enmity by Arion', '*.*OM', 'C:\WINDOWS\COMMAND', 'ANTI-VIR.DAT', 'CHKLIST.MS', 'CHKLIST.CPS' and 'IVB.NTZ'. Infected files have the word 3636h ('66') at offset 0003h.

```
Enmity.808         B978 018B 960A 048D B61A 018B FEAD EB03 ABEB 0433 C2EB F9E2
```

Example.472 **EN:** An appending, 472-byte, fast, direct infector containing the text '*.com'. Infected files have the word 4456h ('VD') at offset 0010h.

```
Example.472        3EC7 86F4 0256 445A 585B 05D8 0183 D200 B900 02F7 F140 3E89
```

HCarry.850 **CN:** An encrypted, overwriting, 850-byte, direct infector containing the texts 'HOLY COW! Whats your favorite planet?...Mines the SUN! One time i studied it for a whole hour i almost went BLIND! Hey!....Whats goin.....Hey! Now just for some silly crap! FLOCK! Hehehehe Look At YOU! Back to the Computer Store for you! This is HORRRIBLE!Who would do something like this? MY LEG DOESNT BEND THAT WAY! MOCB This Virus has infected this file if you havnt found that out yet! Please insert 25 cents! DO DO DO Were Sorry your call did not go threw please hang up and try again JERRY JERRY JERRY JERRY JERRY JERRY Jerry Springer to HOT for Television DOH!', '*.txt', '*.com' and 'This file is now infected! By The HCarry virus! MoCBDUKE[Codebreaker, 1998]'.

```
HCarry.850         BE18 0189 F7B9 3A03 E804 00E9 0A00 ??AC 3206 0E01 AAE2 F8C3
```

Kraken.1223 **CR:** A stealth, appending, 1223-byte virus containing the texts '[Virus KRAKEN 1.1] by Int13h - HeZkRiTo En PaRaGuAy' and 'Dedicado al buenísimo grupo colombiano de heavy metal'. The virus is equipped with an anti-tracing mechanism. Infected files have their time-stamps set to 62 seconds.

```
Kraken.1223        B440 33D2 B9C7 04CD 21E8 C000 B440 B903 00BA BD04 CD21 2E8B
```

Light.1054 **CEN:** An appending, 1054-byte, direct infector containing the text '(c) Light General.Kiev.1995.For free use!'. Infected COM files have the word 2424h ('$$') at offset 0003h and infected EXE files have the word 7878h ('xx') at offset 0012h.

```
Light.1054         BA00 01B9 1E04 B440 CD21 33D2 2689 5515 2689 5517 C3B8 2012
```

Mdrg.533 **CR:** A stealth, appending, 533-byte virus with the texts 'Mandragore', '[Mdrg v3.7]' and 'Mandragore for president !!!'. Infected files have their time-stamps set to one second.

```
Mdrg.533           FEC4 CD21 B43F B912 0233 D2FE C4CD 21B8 0042 B900 00BA 0000
```

Nucleii.1388 **CN:** An encrypted, appending, 1388-byte, fast, direct infector with the texts 'F-PROT anti-anti-virus program Version 1.0 !nUcLeii Software International Too bad your now infected with the frisk virus. Sorry.,.hehe., but thats the way shit works. If you weren't stealing soft ware, or trying to get p0rn or something, then this might not of happened. Don't buy products that harass their user Stay away from things like McAfee, Norton Invircible, err well,.hehe., seems like everyone is selling out these days. Greetings to fridrik and frisk software. Information about antivirus scanners, and how most are just crap not worth wasting your money on. Hope this is " nit - witty " enough for ya fridrik!!!', '*.com', and 'frisk by nUcLeii 9/09/98'.

```
Nucleii.1388       F6D0 F6D8 3E32 8649 01F6 D8F6 D0D0 C8D0 C8D0 C8D0 C8AA E2DF
```

Pollute.405 **CR:** An encrypted, appending, 405-byte virus with the text 'Prodi culo biancorosso'. It replicates on systems on which the Interrupt 21h service routine is located at 011Ch:109Eh.

```
Pollute.405        8B86 7F02 8DB6 0301 B9B4 0031 0446 46E2 FAC3 ???? BC02 01E8
```

PSV.2135 **ER:** An appending, 2135-byte virus containing the texts 'SEPULTURA', 'v.3.0 [XX.07.94]', '[Made In Portugal]' 'Chaos AD', 'Gongratulations! ', 'You are NOW the HAPPY owner of the latest Nightmare Production of Portuguese Software VIRUS', 'YES! this is one little SON of a BITCH especialy maded to KICK your ASS!', 'Your System is NOW on CHAOS AD', 'SEPULTURA for ever!' and '(Agradecimentos a MAX C. & SEPULTURA .... Keep Trashing!)'.

```
PSV.2135           B957 08B4 40CD 2150 558B ECC7 4602 DC00 5D50 558B ECC7 4602
```

Skin.490 **ER:** An appending, 490-byte virus containing the text 'Skinner / BY SPO0KY/ AUSTRIA'. Infected files have the byte 53h ('S') at offset 0012h.

```
Skin.490           B440 8D96 0001 B9EA 01CD 21B8 0042 33C9 33D2 CD21 B440 8D96
```

Youc.1648 **CER:** An appending (EXE) and prepending (COM), 1648-byte virus with the plain text message 'FUCK YOU' and the encrypted text 'C:\COMMAND.COMÿAID,VIR,DINF,CHK,TEST,AUR,PAV,NAV,-V,SENT,ASM,SCAN,LEAN, ANT,SAFE,BOOT,STRA'. Infected EXE files start with 4D5Ah ('ZM').

```
Youc.1648          CD21 E9A4 FEB4 40BA 0001 B970 06CD 21C3 B802 4233 C98B D1CD
```

Zarma.2408 **CER:** An encrypted, appending, 2408-byte virus containing the texts 'ZARMA-VIR by T.Power *** Claudia Schiffer Lives !!!..', '*.COM', '*.EXE', 'SMARTCHK.*', 'CHKLIST.*', 'ANTI-VIR.DAT', '*.VIR', 'NAV_._*', '*.IM', '*.NTZ', 'FI*.FF?', '*.CRC', 'SCAN', 'F-', 'VIR', 'VS', 'AV', '.S', 'BMB', 'BMD', 'TB', 'IM', 'IV' and 'SOP'. Infected files have their time-stamps set to six seconds.

```
Zarma.2408         BD?? ??6A 0007 8D86 7401 BF04 00AB BF0C 00AB 8CC8 ABB4 CCEB
```

INSIGHT

Szöring to New Heights

1998 has been one hell of a year for Péter Ször, in his own words 'the most difficult time of my life'. His young wife Natalia, whom he married only last summer, was taken seriously ill in Finland just one week before VB'98. She insisted that he go to Munich and present his paper entitled 'Attacks on Win32'. After an agonizing decision, he did, and it proved one of the most popular and well-received talks of the conference.

Péter is optimistic, 'things are getting better now and we hope for her 100% recovery. God was keeping an eye on her. I'd like to take this opportunity to thank her for her support over the last 10 years – I would not be the same person without her, it is as simple as that.'

Just a year ago, he was thrilled to be asked to join *CARO*. 'We are friends in *CARO* and it is the highest honor I could get in my life. We help with each other's work. I learn a lot from other *CARO* members, especially Vesselin Bontchev and Eugene Kaspersky.'

Today Péter is the Lead Virus Analyst at *Data Fellows* and spends his time analyzing new viruses and designing scanning and disinfection technology. He also worked on the design and partial development of the Virus Analysis Toolkit which facilitated the conversion of *F-PROT 2.x* database entries to *3.x* format. 'Analyzing viruses is very challenging. You have to teach yourself the virus' environment and learn the internal mechanism of the operating system. This is extremely interesting. Sometimes I feel burned out, but the feeling goes away again after a few weeks. I've spent eight years with viruses.'

He has firm convictions about the methodology of his work. He is an advocate of the exact identification process, the like of which is used in the *F-PROT* engine – 'the situation is getting very difficult with new viruses. It is impossible to stay expert with all the different virus types nowadays. That was why I concentrated on *Windows 95* and *NT* viruses during the last year. We anticipated a tough situation this year and Win95/CIH and Win95/Marburg came along and proved it. Now, we are pretty close to shipping a good heuristic engine. I believe in heuristics. I think the only way to keep up with the number of new viruses is to develop new heuristic engines.'

Péter thinks that most anti-virus researchers were not convinced that polymorphic viruses could cause chaos a few years ago. Win95/Marburg, a complex polymorphic virus, is now firmly established on the WildList and he is convinced that more intensive research time must be devoted to such viruses. While he admits 'our technology did become better against traditional, decryptor-based, polymorphic viruses, I'm afraid that in less than a year we will see more polymorphs which are not based on randomly generated decryptors and keys. I have just received a virus which mutates itself without decryptors. All the subroutines in the virus body are mutated to include new junk code which changes the structure of the virus code completely. There is no constant code in these viruses and we will have to develop new algorithms to deal with them.'

Péter is the first to admit that he is something of a pessimist when it come to viruses, but he is not a quitter. 'We may have solved all the problems pretty well during the last decade but it will become more difficult in the future. I am ready for anything.'

Born in 1970 in the cheerfully named Hungarian town of Pa'Pa, Péter Ször was brought up in the popular lakeside resort of Balaton. His affinity with computers did not begin until he started at secondary school. 'I remember playing Galaxian in a shop in 1981 and praying to God to give me a machine like that at home as I did not have the money to play on it as much as I wanted. A few years later I found myself in the school computer lab playing games on 8-bit computers like *HT 1080Z, Primo, Spectrum* and later on a *Commodore 64*.'

His new passion was not indulged at first and this he blames on his upbringing. His mother was a music teacher, his father a maths and physics teacher and his grandparents were teachers too. The pressure was on – his parents were very keen for him to learn anything and everything and, naturally, he did not share their feelings. Nevertheless, Péter played the piano from the age of nine for six years, until he came across computers when piano lesson time suddenly became PC time.

It is an era he recalls fondly, 'I had special permission to go to the lab whenever I wanted with a few guys from school. We were there even in winter when there was no heating, writing all kind of games. Handling the keyboard was very difficult in gloves, but we loved doing it. There were different students there who were about to graduate and their experience helped me a lot. No-one knew about *Microsoft* then – now all of them work for *Microsoft* (Hungary), one of them is the director there!'

Péter and his school friends channelled their energies into programming, which they found a real challenge. They rewrote the same program hundreds of times until there was no way to make it any shorter or faster. He remembers receiving a *Commodore 64* from a West German friend of his father's. It was then he learnt his first basic lesson. 'My father did not realize that the computer needed at least a tape to save programs and reload them. Since this was the first *Commodore* we'd seen here, everybody was program-

ming it with me. We had to write the source code on paper and retype it next time, and so on. This happened for two months until I got a tape.'

Having graduated from school, Péter enrolled at the University of Veszprem to learn professional programming. By his own admission, he was not really in the market to study in the conventional sense, and remained a rebel. 'In fact, I formed an underground band called 'Negative Lehetoseg' or 'Negative Possibility'; I played the bass guitar there. I was very into music and this was a good way of expressing myself as well as being a method of kicking against my country's political system before the revolution'. That was soon to change. In 1990 Hungary was to experience a political revolution and Péter's father would become a member of the first free parliament.

Before his last year at university, Péter took a summer job in a hotel. At the end of the summer, and at the suggestion of his new girlfriend (now his wife Natalia), he was able to buy a computer – a Taiwanese 286. He went back to college in September and collected some new programs from home, including a game called J-bird. Then his new computer became infected with a very common virus: Stoned. At this point, he knew very little about viruses and was lucky in that Stoned immediately displayed its self-identifying message on the screen.

He remembers the experience well. 'I was afraid to turn on the infected PC for a day, but it was a Saturday and I had no one to help me. Finally I turned it back on – there was no real risk, just a few Pascal sources and some games to lose. Since I was booting from the diskette and the infection happened that way, I realized that something hidden had occurred, since no programs had been executed.' He referred to some books about boot sectors and saw how the boot sector can be loaded with DEBUG.EXE.

'Sure enough, I found the virus' message in the boot sector. It was an unbelievable discovery. I saved the code and tried to analyse it in debug, but did not know assembler at all. Then I tried to write a program in Pascal which would detect the virus in the boot sector of diskettes. The following week I spent all my time reading a book on assembler and analysed the code on paper. I had big problems reading the hard drive MBRs since I did not analyse the virus completely and did not know the identifier of the hard drive 0x80 for the disk interrupt, but a teacher of mine helped me with that and then I had my first scanner ready to test.' It turned out that 90% of the university's PCs were infected with Stoned.

The search was on for a remover and Péter had to write one. From then on, the subject of his final paper for the university was established. He started looking for new viruses everywhere and people sent them in – Jerusalem, Vienna, Cascade and Yankee Doodle came in month after month and he had them printed out and analysed them line by line. He put detection and disinfection routines into his diploma work under the name of Pasteur Anti-Virus.

Péter graduated in 1991 with an 'A' grade in spite of his poor form in earlier years. The result of his research formed a popular freeware progam which meant no money was coming in. He started looking for some kind of job, but there was nothing available in his area. Moving to Budapest, he started to work for a joint French-Hungarian venture called *SG2-H* where he helped to develop financial software for large banks. The company soon became involved in the development of the Hungarian GIRO project, the electronic transaction system.

Soon he was to come across viruses again, 'I worked for the Hungarian Virus Buster team for a few months analysing viruses on a byte per forint (Hungarian currency) basis. So, a one-sector boot virus meant 500 forints. If you got a polymorphic one, you had to spend ten times as long and got a few thousand. It was very difficult to do, but I learnt a lot and it was the only way to earn money at the time.' He soon realized it would be better to develop his own scanner based on the knowledge he had accumulated from these complete analyses.

While working for *SG2-H*, Péter's Pasteur program gained reknown and was professionally developed into a commercial program in 1993. *Pasteur Anti-Virus* retailed to the large banks and to other big corporate clients in Hungary. From the end of that same year he was head of his own business and was selling *Pasteur* himself, doing everything from virus analysis to packaging and posting. All the time, he was on the lookout for partners.

With the help of a company called *HELIX* the *Pasteur Plus* NLM version was developed for *NetWare* during 1994. Unfortunately, the same company had a large Unix-based project which caused them to go bankrupt in 1995. Péter remembers a very bleak time, singlehandedly supporting all the platforms while supplementing his wages with a full-time position in a bank.

By the end of 1995 the situation had become untenable. For the previous few months Péter had been corresponding with the director of *2F*, the Hungarian distributor of *F-PROT*

Professional. He says, 'I had always liked *F-PROT* and looked up to it. I was naturally interested in working for *Data Fellows* in Finland when they suddenly offered me a job. I knew that Vesselin Bontchev had just left Germany to work for *FRISK*. I thought it was time for a change and left Hungary at the beginning of 1996.'

As a virus analyst, Péter is ambivalent about his natural enemies, the virus writers. 'I try not to hurt anybody. Well, sometimes I do. Some anti-virus researchers call virus writers idiots – some of them are pretty close to that. I consider virus writing a bad thing and try to help people get rid of viruses from their systems. I think most virus writers will grow up and turn to something else, the problem is that there is always someone after them. New faces...'

He is slowly making a name for himself and is optimistic that his message may be getting through. 'I think I have the respect of some virus writers. They respect my knowledge and what they know from my papers and articles. I think I help some of them to quit and do better things – I can't be sure about that but I hope so. Somehow I have the illusion that if they all quit (or at least the more influential ones) then I can do something else too.'

As for his own industry, Péter's prognosis is not as outlandish as it first sounds. 'Maybe all of us will work for the same company one day,' he jokes. 'Seriously, it is getting harder and harder, even for big companies, to keep up with the situation nowadays.'

This may be why he is determined to stay with viruses for the forseeable future, despite what he told fellow *CARO* members recently. 'I suggested that everybody should retire at around 34. Most *CARO* members are older than this, so you can imagine their reaction! I think I should stay with virus research if only because there are fewer and fewer people who are staying with this subject for a decade or more. It is very important to get to know the situation globally and this will make our work more and more important during the next few years.'

As VB'98 attendees will have seen, Péter speaks and writes commendable English, 'the only way I could work in the PC environment' and it does not stop there. He learnt German and Russian for twelve years at school but has never used it, a situation he is keen to rectify. 'Often, at conferences, I suddenly realize that everybody around me only speaks Russian. They all say speak Russian – they say it is the natural language of the anti-virus researcher!'

When he is not analysing viruses, he can often be found working out in a gym with Natalia. Typically, the two have something in common for him, 'it's hard to say which is more challenging, analyzing Win32/Cabanas or doing 5-6 reps with 160Kilos in squat position!' He still loves good games. He thinks that the reason he spends most of his free time analysing new PC viruses is attributable to the fact that no computer game has ever beaten him. 'When I can I get back to my *Nintendo*. Super Mario 64 is amazing...'

OPINION

Wilder and Wilder

Peter Morley
Network Associates, UK

At VB'98, Shane Coursen from *NAI* presented an excellent paper on the advances being considered in WildList practices. His proposals signified a massive stride forward from current, archaic arrangements but I felt they did not go far enough. Nevertheless, if one is to criticize proposals, however constructively, it is essential to do two things. Firstly, you must define the problems to which a solution is being considered. Next, you must show how the suggested amendments address some of them.

Irrespective of WildList practice, there is another problem which has festered for years, and which was discussed extensively in the forum following Shane's presentation. It is the problem software reviewers in news-stand monthlies face – they do not have proper virus collections to evaluate anti-virus products against.

Despite this, they still have to do the reviews and most of them gloss over the most important point – the ability of the product to detect viruses. The resulting article is often highly entertaining, readable, gives useful feedback on usability and interfaces, and helps to sell magazines. Unfortunately, to a knowledgeable reader wanting to compare detection capabilities, it is at best useless, and at worst misleading.

The reviewers are, of course, well aware of this, and some of the more responsible ones telephone an anti-virus vendor and say 'I am doing an anti-virus review. Is there any chance of you letting me have access to a comprehensive virus collection to do some tests, please?' I know of no case where such a request has been refused. The vendor makes a virus collection available, provides a competent person to give advice and help, and allows the reviewer to conduct any relevant tests, using any anti-virus product.

This is not enough. Despite maximum cooperation, the reviewer does not get adequate 'trial and error time' and may not be able to repeat the testing, if and when they suspect something should perhaps be changed. Worse – unless the reviewer liaises with three or more vendors, an accusation of bias towards the one he rates 'Top' may be made, and time may not be sufficient to avoid this.

My proposals address the above problem. They also address WildList problems by eliminating the WildList altogether at a later time, when relevant people agree it can happen. While my proposals may be followed by a comment (to encourage debate!), they will have to be acceptable to the present *WildList Organization* (*WLO*), because a high proportion of the implementation workload falls there.

The Proposals

1) A ReviewList should be prepared, and then updated on a continuous basis. The ReviewList should include:

i) Everything on the WildList. Should it be everything which was ever on the WildList?

ii) Everything submitted by the additional reporters suggested by Shane. This will include many viruses found in the wild locally and then killed off, but which the reporters feel are worthy of inclusion. (History suggests that such viruses sometimes reappear, particularly macro viruses sent to customers, before being killed.)

iii) A limited number of viruses submitted by anti-virus vendors. This should be biased towards viruses received from the field, and they should be identified, without naming the original sender. However, it can also include viruses which the sender feels may be of interest to reviewers, for whatever reason. The reason will often be difficulty of detection. (The sender will probably have detected it.)

The initial submission should be limited to 50 viruses from each vendor to make the initial workload viable. Thereafter, submissions should be at the sender's discretion.

At this point the ReviewList will probably be less than 1500 viruses, and rising slowly. Compare this with my firm belief that there are well over 2000 viruses in the wild, at the time of writing.

2) All submissions *must* be accompanied by replications. This may reduce the workload on the *WLO*. It will also avoid the submission of Trojans and wannabes.

3) The rejection of samples must remain in complete control of *WLO*. There are several good reasons for rejection but the one which I like most is the elimination of minor variants of macro viruses, to avoid the discussion about whether they are variants at all.

4) The ReviewList should be followed by the preparation of a Review Collection. The workload will be minimal, but not zero, if the guidelines above are followed.

Now, we get to the controversial part.

5) The Review Collection should be given to a controlled list of reviewers, accompanied by a legally enforceable agreement that total responsibility for any virus be taken by the reviewer accepting the collection. If this proves to be difficult, or not practicable, then we should forget the whole project.

6) Updates of the Review Collection should be sent out every six months to reviewers still on the list. It must be made clear that all updates are subject to the above legally binding agreement.

7) The Review Collection and updates should be accompanied by a set of comments about some of the viruses in them, and the initial Collection by a few notes on possible testing procedures, to help the reviewers get started.

8) When the ReviewList is accepted, at a time to be agreed, the WildList can be abandoned easily, because it is merely a subset. This should be done as early as possible.

General Comments

In my opinion, these proposals will give anti-virus software reviewers something they have wanted for some time, if they are prepared to accept the conditions and responsibility. They will soon learn how to handle a small virus collection, in a secure way (it is not difficult). Armed with this information, their reviews will be better, and much more useful to their readers.

Escapes *will* occur but I do not believe this invalidates the proposal. These are my reasons behind that rationale. Internal escapes may even be a good thing! The outside world will not get to hear about it but all hell will break loose at the magazine/publisher. At the cost of internal productivity, a lot of lessons must be learned quickly.

Further, escapes via a cover disk or CD-ROM are most unlikely because most magazines are already paranoid about testing them. Any instance, even if it does happen, will be much less serious than the recent CD-ROM distribution of the CIH virus. This leaves other external escapes. They will be very rare, but if they do happen, they are much easier to handle than they used to be. The most likely external escapees are macro viruses, and most recipients can already handle them very effectively.

Wild and Dangerous

Vesselin Bontchev
FRISK Software

I must respectfully disagree with Peter Morley's qualification of Shane Coursen's VB'98 conference paper as 'excellent'. In his speech, Mr Coursen made several misleading statements (e.g. he claimed that it had already been decided to send viruses to magazine reviewers while, in fact, the matter is still being discussed). Furthermore, his proposal for so-called 'corporate reporting' (when the WildList will be compiled from reports received by unqualified people, sent anonymously, not stating which scanner was used, and not sending a sample) will severely reduce the already doubtful usefulness of the WildList.

In his article, Mr Morley points out, correctly, that magazine reviewers should be given access to rich and well-sorted virus collections for the purposes of their reviews. He also emphasizes, again correctly, that this should not be done by any particular anti-virus vendor, in order to avoid bias. However, he fails to realize some very important additional points and his proposal is far from the best that can be done in this respect.

The bias issue is easily resolved – the entity providing access to the viruses must not be a commercial anti-virus organization. It should be an independent, non-profit entity. *WLO* fits this role nicely – but is far from the only one. Academic research centres like VTC-Hamburg and the University of Tampere have both the expertise and the resources to do the same.

While, as I mentioned above, the current usefulness of the WildList is at best doubtful since it does not reflect the reality well enough (I will write a future article on this issue), I disagree with Mr Morley that we should get rid of it completely. It is still important to emphasize whether the scanners can deal with the viruses that are actually out there – we just have to improve the WildList and make sure that it indeed lists the viruses that are actually 'out there'.

Furthermore, in addition to the in-the-wild viruses, the testers should be encouraged to run tests against much larger virus collections. While the detection of those other viruses is not as important as the in-the-wild ones, it is by no means unimportant because these viruses are publicly available on the virus exchange web sites and can be used by malicious people to wreak havoc in someone's system.

I see no reason why the number of these additional viruses in the test-sets should be 'limited', as Mr. Morley suggests. Just the opposite – we, the anti-virus people should make sure that the test collections are as complete as possible. There is no need to increase the collection by a few viruses at a time. All of us have reasonably complete collections of known viruses. They contain many thousands of viruses and we should be happy to provide them for test purposes, provided that we are ensured that proper controls will be in place and no leaks will occur.

I also disagree that such a collection should exclude the minor variants of the macro viruses. I would like to remind Mr Morley that the difference between a Wazzu variant which does the worst possible damage a virus can do (data diddling – slow, generally unnoticed corruption of information) and a Wazzu variant which does absolutely nothing is exactly one bit. I am sure users would like to know which of those two variants they had and whether their product is able to detect them both and distinguish between them.

I very strongly disagree that a collection prepared this way should be given to the reviewers. From personal experience, I know that the reviewers simply lack the expertise to handle viruses properly and leaks will occur. No amount of legal agreements will help here. They will be simply unenforceable.

Due to the self-replicating nature of the viruses, it is impossible to prove beyond reasonable doubt that a particular leak has occurred from a particular person. A virus cannot be 'marked' without creating a new virus – or the mark disappearing during replication – so no proofs can exist. Instead, the reviewers should be provided access to the collection. This can be done in a special virus lab provided by the independent entity. If *WLO* does not have the means to establish such a lab, many other independent anti-virus entities already have one.

Alternatively, the collection could be brought to the reviewer's site for the duration of the test only, and the test itself supervised by a competent anti-virus person. That person would have to make sure that no viruses were leaked during the test and that no viruses remained in the possession of the reviewer after the test had been completed. In addition, this person could provide useful advice to the reviewer about how to conduct a meaningful test of anti-virus products.

I also strongly disagree with Mr Morley's suggestion that while virus escapes will occur, they do not invalidate his proposal. They do. We, the anti-virus people, must be responsible and do our best to prevent such escapes, not contribute to them. Giving viruses to people who lack the expertise to handle them in a secure way, like the magazine reviewers, is definitely not doing 'our best'. If it is done, we will become part of the problem. The general public will not like it either – as was illustrated by its reaction when it became known that some of the distributors of the anti-virus product *InVircible* were sending viruses to their prospective customers, so that the latter could better 'evaluate' the product.

Even giving reviewers the limited number of viruses currently listed in the WildList is something that must not be done. Many of them are 'in the wild' only in some localized regions, others are no longer in the wild but are still listed because of the way WildList reporting is currently managed. Giving them to people who may let them escape increases the probability of those viruses spreading in areas in which they are not currently widespread.

Furthermore, a well-sorted collection of viruses very likely to spread well is much more dangerous than the huge collections of junk currently available from the Web. Even if that junk contains (among others) the viruses currently on the WildList, the number of non-viruses or viruses in it which are difficult to replicate and unlikely to spread means that a random leak from it is much less likely to cause an outbreak than a random leak from a WildList collection.

Finally, I would like to point out a fact Mr Morley has not noted. While providing a well-sorted virus collection to the magazine reviewers is what many of them want, this will not necessarily increase the general quality of the reviews. Firstly, many reviewers will still prefer to download a junk virus collection from the virus exchange web sites – instead of going through all the trouble to obtain one from *WLO* or any other similar entity. Secondly, having a well-sorted virus collection is necessary, but far from sufficient to conduct a good anti-virus test.

This last point is perfectly illustrated by a recent test published by a German computer magazine. The magazine obtained the actual virus detection data from a competent

and independent anti-virus organization – VTC-Hamburg. Nevertheless, the review was full of basic mistakes – mistakes introduced by the magazine due to lack of expertise and competence in the field of anti-virus testing.

Reviewing What's in the Wild

Shane Coursen,
Nework Associates, US

As the movie Contact opens, Ellie Arroway stares at the dial of a short-wave radio repeating 'CQ, CQ, this is W9GFO, is anybody out there?' Over and over again, she repeats the phrase, but there is no response. Ellie turns in frustration to her father and says, 'I'm not getting anything'. Seeing that she is rather discouraged by her failure to contact other people on the same frequency, he says 'Small moves Ellie, small moves'.

Ellie is impatient and desires immediate success – making small moves is not in her makeup. A moment later, she is rewarded when another ham operator returns her hail. The movie then takes us forward in time, focusing on the adult Ellie. It is immediately obvious that she is still driven by an equal amount of energy. Ellie still wants it all. Even when others believe her pursuits to be 'crazy' or a 'waste of time', she refuses to yield to outside pressures – even if it means being ostracized by her peers.

I draw this parallel because I have felt like Ellie since that fateful presentation I gave at VB'98. After announcing that *WLO* would in fact be giving viruses to reviewers, I feel that I have been regarded as having little forethought to possible consequences of giving viruses to reviewers, to being too aggressive as a board member of *WLO*, and as a person who is a danger to himself and others around him.

While some may disagree with me, I keep coming back to the fact that if WildList participants do not support the idea of supplying viruses to reviewers (i.e. if the 'vote' is against the idea), *WLO* cannot do it. After all, the participants supply the viruses in the first place. If a participant disagrees with the actions of the *WLO* board, they can choose no longer to supply the viruses. If enough participants fail to support the idea, the WildList source of viruses will eventually dry up.

While I agree that my announcement should not have been an announcement at all, rather, presented as a suggestion, I still very strongly believe that supplying computer viruses to reviewers who can prove themselves competent virus handlers is a good idea. I very strongly believe it is something that should occur sooner rather than later.

For Starters

I would first like to touch upon the possibly false belief that generalist magazine reviews of anti-virus products are important to the world at large. While we, as anti-virus researchers and vendors believe this to be the case, apparently not everybody thinks along the same lines. In the corporate world at least, this may not be the case.

In the panel following my presentation, when the audience (which I considered to be largely corporate) was asked if they based their purchasing decision on generalist magazine reviews, very few people raised their hand. (Were they just being shy? Was it because the audience was primarily European corporate IT, and the attitude towards anti-virus reviews differs from that of US corporate IT? These are questions that I do not have an answer to, but which could play a role in the true importance of anti-virus reviews in generalist magazines.)

What then is the target audience for generalist magazine reviews of anti-virus products? Small office, home office, and individual home? Possibly. Let us assume that to be the case for a moment. Given that assumption, now ask a different question. Is it worthwhile for the reviewer to place such a great emphasis on one small set of computer viruses – the set of viruses based on WildList data? Based on the assumption that a SOHO/homebound user is more likely to visit the Internet where a corporate user would not, and based on the assumption that the Internet is more likely to harbour infected files, the answer is yes, it is worthwhile.

Should a generalist reviewer test against only those viruses contained within an official collection? I agree with Vesselin by answering with a resounding no. Larger virus sets should be tested against. Again, based on the assumption that SOHO/homebound users are much more likely to visit the Internet where a corporate user simply would not, and especially based on the assumption that the Internet is likely to harbour infected files (of new and unknown viruses), it is actually more worthwhile for the reviewers to test against viruses that are not officially in the wild.

Second, before attempting to tackle the main issue, something that I was able to establish with accuracy during my presentation, was what the WildList actually is. Or, at least, what it has become. For those people who were not at the presentation, I will describe it again. The WildList is not a list of all viruses known to be in the wild. It is a list of viruses that are known to be spreading in the wild. Given that one statement, I will now attempt to describe how the WildList contents come about.

In the past, the basis for a participant reporting a particular virus is that they must have received two valid reports of the virus within a one-month timeframe. This first criterion excludes all viruses that have been reported only once and so many macro viruses 'found in the field' are excluded. As many people are aware, macro viruses seem to be the most often experienced type of virus – reason one for the WildList not being an all-encompassing virus list.

Furthermore, the participant must send a replicable sample of the virus when making their report. There are many valid reasons that a replicable sample may not be available to the

participant and we find that even viruses that are spreading in the wild do not show up on the WildList until well after their original 'reported in the field' date. The reason for not accepting reports without valid replicable samples is due to the fact that if a virus makes it to the WildList, anti-virus products will be tested against it. If a sample does not exist or cannot be provided to all anti-virus vendors for analysis, then the test against WildList data becomes unfair and weighted towards the vendor/participant who reported the virus but failed to provide a sample – reason two for the WildList not being an all-encompassing virus list.

So, we now understand the two basic criteria that form the basis of how the WildList comes about. Unfortunately, those are rules that can only be adhered to in an environment that never changes. As I was able to point out clearly in my presentation, our environment has changed. Is it possible that WildList reporting criteria also be redefined?

Since the time WildList reporting criteria were first defined, we see that people's (specifically referring to users) interest in viruses has lessened. People simply are not reporting viruses the way they used to – reason three why the WildList is not as all encompassing as it once was.

This is due in part to the evolution of anti-virus products. To a great extent, products have got better at detecting viruses. Better detection means fewer users reporting suspicious events and undetected viruses to their vendors (ergo, WildList participants no longer hear about all of the viruses that are being found) – reason four why the WildList is not as all encompassing as it once was.

Once again we see that advancement of anti-virus software technology changes which viruses make it on to the WildList. Auto-disinfection is a common feature employed by many anti-virus products. Even if the user had an interest in reporting a virus, fact is the end-user may never even know they had a virus incident. This is reason five (and the final major reason I will point out) why the WildList is not as all encompassing as it once was.

Due to the fact that all viruses found in the field are not showing up on the WildList, Vesselin refers to a 'certain level of doubtfulness to the usefulness of the WildList'. Ironically, he rejects the idea of corporate reporting without having heard all of its details. If there is enough interest, I will write an article outlining how corporate reporting and its fairly loose 'rules of reporting', may fit in as a valuable addition to current WildList practices.

Putting aside the issue of the importance of generalist magazines reviewing anti-virus products, and the general usefulness of the WildList, we now come to the issue of whether *WLO* should supply generalist reviewers with a set of viruses based on WildList data (henceforth referred to as 'WildSet'). All aspects of the topic of giving computer viruses to reviewers cannot be presented in the limited space available, so I will respond to a few specific passages in the articles by Peter Morley and Vesselin Bontchev.

My views on this should already be quite apparent. If it can be determined that a prospective reviewer-recipient is trustworthy, competent, and has the proper facilities to perform live virus testing, then the reviewer should be provided the WildSet. (I am specifically referring to this type of testing as live-virus testing and not just testing against the detection of a virus set. There is a difference. Those reviewers who want to perform the latter type of testing can simply be referred to an existing official agency that keeps such records.)

If the reviewer wants to do live virus testing, providing access to viruses – as outlined by Vesselin – is simply not good enough. That is not to say that 'providing access to viruses' does not have its place. If the reviewer has the wherewithal to travel to an official and secure lab, then by all means we should recommend they do just that. In my experience, generalist reviewers neither have the time nor the inclination and especially lack the necessary budget.

Vesselin disagrees that we should supply generalist reviewers with the WildSet. At the same time he contradicts himself by saying 'we should be happy to provide them for test purposes, provided that we are ensured that proper controls will be in place and no leaks will occur'. I am aware that the 'them' he is referring to are most likely the various anti-virus vendors, and not generalist reviewers. I cannot but think, however, that a generalist reviewer could not ensure *WLO* that all of the proper controls could be put into place.

I have two more comments on what I have just quoted from Vesselin's piece. One – large virus libraries are not provided to all vendors. For reasons historic and otherwise, this is simply not the current state of virus sample sharing. Two – if we, the anti-virus people, create guidelines to ensure the reviewer puts the proper controls in place, such that no leaks can occur, then no leaks (at the generalist magizine reviewers lab) will occur. This is simple logic, and in this sense I am in disagreement with Peter Morley when he states that 'escapes will occur'.

I believe that almost any generalist reviewer has enough competence to handle live viruses. If a generalist reviewer follows written guidelines to the letter, a leak simply cannot occur. Granted, the guidelines have not yet been written in full, however the most obvious guideline is for the reviewer to perform tests on a closed system, or within a closed network, and properly destroy the contents of all data-holding objects (i.e. format disks, hard drives, rewrite hard drive MBR…). If a leak does occur, it will simply be due to the fact that the reviewer performed tests outside of a closed system – in which case a legally binding contract will absolve *WLO* from liability.

This attitude does not address moral liabilities. It also would not change the perception people would have of *WLO* if a magazine were to release a virus on an accompanying CD that happened to coincide with an anti-virus product review (results obtained by testing against live

viruses provided by *WLO*, of course). In this sense, I find myself in agreement with Vesselin. Unfortunately, there is no simple answer to the issue of ethical and moral responsibilities if we ultimately pursue giving viruses to reviewers.

Peter, Vesselin, and I are in agreement when it comes to the danger of a wild virus escaping. In fact, there is a greater likelihood of a 'wild virus' propagating successfully if it escapes as opposed to a Vx 'library' virus escaping and propagating. After all, an official WildList virus would not be considered wild if it had not already spread to a certain degree. This so-called danger does not deter my belief that a generalist reviewer cannot handle the responsibility of live-virus testing, however. I must reiterate that I do not believe a leak will occur from a generalist reviewer's site.

In my presentation, I asked the audience for a show of hands of those who performed live virus testing to evaluate anti-virus products for their corporation. Given the response, it was easier to count the number of people who did not raise their hands. I then asked the follow-up question: 'Who here has accidentally released a virus during their live virus test?' No hands. Even after prodding a little and saying there is no shame in telling the truth, still not a soul would admit to having accidentally released a virus.

Excluding people who were simply not telling the truth, the reasons for no accidental releases probably vary. It could be the person was smart enough to test on a closed system. It could be the person was not actually launching or replicating the virus (which would not fall within the definition of 'live-virus testing', however I cannot assume that everybody in the room understood my exact question).

It could be that all the people in the room were far more competent compared to any generalist reviewer. It could be a combination of all of the above. In the end, it comes down to a simple truth. Those people who have the confidence to work with live viruses for specific reasons and in a controlled manner, are the same people who are the least likely to leak a virus accidentally. I believe that generalist magazine reviewers fall into this category.

Conclusion

Towards the end of Contact we hear Ellie ask her father (actually an alien) 'What happens now?'. He says, 'Now, you go home'. Again, frustration sets in and Ellie says, incredulously, 'Home? But I have so many questions. Do we get to come back?'. The alien responds 'This was just a first step. In time, you'll take another.'. Not satisfied, Ellie responds 'But other people need to see what I've seen, they need to see… '. Once again, she is reminded of the unfortunate obvious, 'This is the way it's been done for billions of years. Small moves Ellie, small moves.'.

Eventually, we see that Ellie is personally rewarded far beyond anything she might have imagined. By sticking to her guns, the world realizes that just maybe, Ellie's crazy ideas had some merit after all.

FEATURE

What the HTML?

During early November there was a great deal of discussion around the Internet regarding the discovery of 'HTML viruses'. There was much excitement, and as is so often the case, much hype. This caused significant confusion among computer users all over the world. With various experts proclaiming positions ranging from 'it is marketeering to sell anti-virus software' to 'it is the end of civilization', wary observers were left with many questions about the potential danger and what might be a reasonable response to this new threat.

It is important to immediately make it clear that the so-called 'HTML viruses' are unlikely to have been responsible for any real-world incidents, and due to their nature, are also unlikely to cause any serious problems in the near future. The real threat from the first of these viruses comes not from surfing the Net but from knowingly downloading virus samples from the Web and then running them in a 'suitable' environment.

When is an HTML Virus not an HTML Virus?

These viruses were labelled 'HTML viruses' by their author. He conveniently includes descriptions of the viruses and names for them at his web site. Given the reasonably established tradition within anti-virus circles that viruses should be named something other than what their authors desire, it seems unusual that several vendors have adopted the name chosen by the virus author. Independent of failing this test of 'accepted practice', the virus author's chosen names are just wrong!

The 'first HTML virus' (HTML.Offline, according to its author) was actually about the seventh Visual Basic Script (VBS) virus. It should have been named VBS/Offline, not HTML/Offline (HTML.Offline, etc.). So why the fuss?

Well, Offline *is* distributed in HTML files so the confusion is understandable. However, its code is written in VBS and not in HTML itself. This means that it will affect users of browsers supporting VBS – today probably only those running recent versions of Internet Explorer.

Raw HTML cannot be viral, so browsers supporting only HTML are not at risk. Browsers supporting HTML and other scripting languages will have to be considered in light of the capabilities of those scripting languages.

Are VBS Viruses a Threat?

VBS viruses are becoming a threat, due to *Microsoft's* wish to provide the latest versions of *Windows* (*9x* and *NT*) with a powerful, easy to use language that can easily access the

resources provided by these operating systems. VBS code is currently always in human readable form and therefore easily understood by anyone wishing to get into the virus writing scene.

In the past, virus writers had to develop considerable expertise and learn about low-level computer operations before their creations had much chance of successfully spreading. Then, with the arrival of macro viruses for the *Microsoft Office* environment, the tools for virus creation were readily available and much less knowledge was required to produce a successful virus. The same is likely to happen with VBS viruses – the environment they require will soon be commonplace and there are already web sites providing programming tips and example code.

The introduction of a new scripting language would not in itself be a cause for concern. Script viruses of one form or another already exist on many platforms and do not pose a significant threat. VBS is rather different because it can be embedded within HTML pages and when viewed by one of the most popular Web browsers the code will be executed.

Offline is a very simple VBS virus embedded within an HTML document. It is an overwriting virus. When executed under the correct conditions and providing the user clicks 'Yes' to allow the script to run properly, it will look for every file matching '*.htm' and '*.html' (this test is case sensitive) in the current folder and every folder up to the root directory. It overwrites each file it finds with a copy of itself.

To less technical users, this might not be a stern-enough 'warning'.

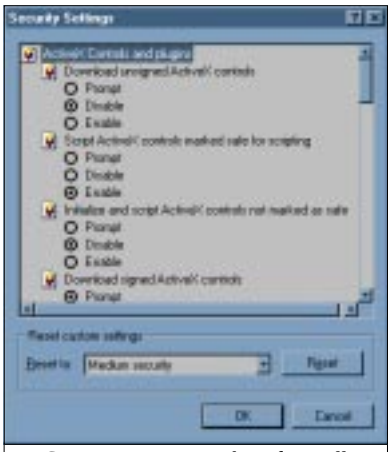
It would be trivial for anyone to create a variant by simply inserting a space somewhere or swapping around lines or statements that are not order-critical. As with VBA macro viruses, this is something for developers to consider when designing 'proper' detection for these things.

The name Offline is apt because it is designed to work only when an infected HTML page is viewed offline, rather than across the Web. This is an important fact to stress. Web browsers do have security settings and at the moment it is unlikely that a virus could run properly and silently from a page viewed across the web unless the user has explicitly reduced their security or had it crippled by some other malicious code.

Several viruses followed Offline, each being cleverer in its implementation or more ambitious in its aims. Similarities between Visual Basic Script and Visual Basic for Applications (VBA – as used throughout the *Office* suite and beyond) have been used to the virus authors' advantage.

It is possible for viruses to share common portions of code between these two environments and already there are cross-infecting viruses which will infect *Word 97* from the *VBS* form and *vice versa*. In these cases there is no HTML involved and the virus either exists as pure standalone VBS script or as macros within infected *Word* documents.

The latest offering from the author of these viruses targets even more file types – HTM, HTML, HTT, HTA, VBS and DOC. It happily infects from any environment to the others and reuses a great deal of its code in doing so. The addition of HTT to the target file type improves its chance of spreading. The Active Desktop in *IE 4.x* uses HTT files as the backdrop of disk folders for its 'view as Web page' option. As these are local files, by default they have rather low security settings associated with them. Additionally, if the virus has spread from the DOC form the unsuspecting user will already have had their security disabled through VBA code tweaking the registry.



Cautious IE-using web surfers will ensure settings here do not change.

The common link between these viruses is not HTML – it is VBS. HTML is everywhere – for now, VBS is not.

The power of VBS comes from its ability to interact with ActiveX objects present under *Windows* operating systems, and often extended by subsequently-installed applications. These objects can be used to manipulate files, walk the filing system, control applications, change settings in the registry, start and stop processes, alter network settings and create and manipulate documents, databases and the like.

With *Windows 95* and *NT 4.0* not all of the necessary ActiveX controls are installed by default and users have to install Windows Scripting Host before they can use them. *Windows 98* and the forthcoming *NT 5.0* (*Windows 2000*) have them installed by default. It will soon be the case that most desktop PCs have the required environment for these viruses to spread, even if not in all their possible forms.

Implications for Anti-virus Software

Scanners may soon have to search from the beginning to the end of every HTML file in order to find segments of potentially viral script, which could theoretically be scattered throughout the file. Searching the entire file will be a slow process compared to the usual entry-point scanning employed for other file types. Detection of a virus that chooses to polymorphically chop itself into segments scattered through the file is not going to be easy, and therefore neither will disinfection. Developers employing heuristic analysis will now have to investigate how to apply their analysers to VBS.

VIRUS ANALYSIS

Compat and Bijou

Stuart Taylor
Sophos Plc

XM/Compat.A first came to our attention in August this year. It has the distinction of being the first polymorphic virus to be written in VBA3 and it is also the first *Excel* virus to manipulate user data.

From the virus analyst's point of view, a great deal of thought has obviously gone into the creation of this virus. There are many aspects of it that are designed to confuse and it incorporates a lot of work aimed at concealing its presence and operation from the casual observer. Further, the complexity and obfuscation of its code would likely have been a stumbling block to many heuristic detection methods at the time of its apperance.

From the user's point of view, the most worrying aspect is the fact that data is randomly changed. It was originally thought that such changes would be in the range ±5% of the original value, but there are circumstances where the change can be significantly larger. This is of course manageable if users have backups and complete transaction records. However, many still do not have sensible backup policies and re-entering large amounts of data is never a popular task.

More worrying is the fact that if infection goes wrong and does not take place, the corruption of user data can still occur. Further, a perfectly clean workbook can have had its data modified without Compat having infected that file. Users are faced with the uncertainty that once Compat is detected on their system, data in all spread-sheets accessed since initial infection is immediately suspect, regardless of the containing workbook's infection status.

Structure

Compat consists of one module with four principal macros – Auto_Open, Auto_Close, Auto_Exit and Auto_Help (Auto_Help just calls Auto_Open). There are also seven private macros (Macro1 to Macro7) and these do the real work. The virus installs itself as an Add-In, placed in *Excel's* library directory (default MSOffice\Excel\Library) with the filename OFF97COM.XLA. As the name suggests, it masquerades as an *Office 97* compatibility Add-In, purporting to allow users to share data between *Excel 5/7* and *Excel 97*.

Infection

The main method of infection is through Auto_Open. It is this macro that creates the Add-In. The macro first checks the version number of *Excel*, and provided it is less than version 8, initiates the infection mechanism. It checks that the Add-In is of a reasonable size and if it is outside the expected range the Add-In is uninstalled and its file deleted. The Add-In is then recreated and the morphing routine is called (Macro2) to create a new source code file. Once all the file creation is done Compat sets the OnSheetActivate mechanism to run Auto_Exit.

The infection method is very complex. Two files – VBA_XL.TXT and VBA_XL.XLS – are created in the application directory. Macro1 is used in the creation of these files and is devious in the extreme. It first creates a randomly named text file with the contents of a module. The module contains the subroutine Auto_Open. It also contains a module with a random name which saves itself as a text file and then runs Auto_Help.

The Tools/Add-Ins dialog displays something like this on infected PCs.

After a great deal of checking and manipulation, the source code file is read into the current workbook which is then saved as the file VBA_XL.XLS in the application directory. At this juncture, the randomly named subroutine from the original text file is called. This is used to make everything very hidden and to call Auto_Help which in turn calls Auto_Open to complete the installation. Even the method of calling other macros is deliberately tortuous and complex. For example, in a possible attempt to disguise from heuristic analysis a call to a macro it has just made, at one point the virus calls <workbook>!<sheet>.<macro> thus:

```
Application.OnTime Now l & j & "!" & d & "." & d
```

The creation of the Add-In itself is found in Macro5. The process is to create a new workbook, add a module to it, delete all the worksheets, read in the source code text file created earlier, execute the macro supplied with *Excel* for creating Add-Ins (VBA.MAKE.ADDIN), save the changes and set the installed flags – a very standard procedure.

As mentioned earlier, the infection does not always work properly. In a fully installed environment it should work but in a custom environment there can be problems. Macro4 sets up global variables for the rest of the macros. One of these is the application library path, which, in a typical *Office 95* installation points to C:\MSOffice\Excel\Library. However, in a cut-down installation, the library directory may not have been created. In this situation, installation of the Add-In fails as the virus is hard-coded for the default; the user is dumped into the Visual Basic Editor (VBE).

Should this happen, the virus writer has another piece of trickery up his sleeve. The first few lines of the macro code are faked to make it look like a user-recorded macro. This is achieved by prepending the message 'Macro recorded <date> by <username>' to the top of the file. Macro4 is also responsible for handling the changes necessary for infecting in the Macintosh environment. It expands file names into typical Mac ones.

The Auto_Close macro is also used in the initial infection procedure and sets up the Add-In installed flags if they have not already been set.

Auto_Exit is the mechanism through which *Excel* workbooks are infected. This calls Macro6 which is used to create a module before the first worksheet in the workbook. The module is given a random name of random length. The virus source code is read in from the text file written earlier. Considerable effort has gone into making it appear to the user that nothing has happened. The currently selected sheet is remembered and avoided in subsequent actions, and screen updates are turned off during the copying process. Remember that the OnSheetActivate mechanism is set to run the Auto_Exit macro.

Polymorphism

Compat's polymorphic engine is contained in Macro2, which can be called from Auto_Open or Auto_Exit. It first creates a new file with a random name and writes out the bogus 'Macro recorded by' message using the current date and the application username. It then proceeds to read in the original text file line by line. Each line is checked for the presence of the comment operator, the apostrophe. If this operator is found then the comment and the operator is removed from the line.

Following this, there is a one in two chance of a new comment being added to the end of the currently processed line. This comment is of random length up to ten characters. There is then a further one in ten chance that a completely new comment line, which can have up to six leading spaces, will be added as the next line in the new file. The comment itself can be up to 49 characters long.

Eventually, after the whole of the original file has been read in and modified, the new file is saved and is renamed to the original fixed filename. As part of this mechanism all blank lines, including those that consisted originally of just comments, are removed. Typically, new files have 10% extra comment lines within them, but because all comments were stripped to start with, the virus does not keep growing.

Payload

The payload is to be found in Macro3 and this is called solely from the Auto_Close macro. It is a very complex payload, designed to be subtle and very difficult to detect with the naked eye. The first thing to note is that the payload only runs after 31 August 1998.

It consists of checking every 'used' cell in all the worksheets except the currently active one, until a maximum of 1000 cells have been visited. For each of the selected cells, a random test is made which has a 1 in 100 probability of success. Should this test pass, then checks are made that the cell is not empty, that it has a numeric value and that it does not contain a formula. Should all of these conditions be met, the value of the cell is multiplied by a randomly-selected value such that the result will be within ±5% of the original value.

The virus remembers the length of the original value and truncates the new value to that length. It is during this truncation that larger changes can occur. For example if the number 999 appears in a cell and is multiplied by 1.04990% then the result is 1048.85. However, after truncation from the left to three digits (the original value's length) the new value is 104. If not caught early, errors approaching an order of magnitude could have profound effects on any decisions based on the results of calculations in Compat-affected spreadsheets!

Under *Excel 5*

Compat cannot infect the global environment of *Excel 5*, but its payload works just fine. Perhaps 'fortunately', a series of errors are generated when infected workbooks are opened, as the VBA3 in *Excel 95* contains some extensions to that of its older sibling and Compat depends on some of these. Affected sites should check for *Excel 5* users who have seen 'Run Time Error 76 : Path not found' errors while opening files.

Verdict

The virus writer spent a lot of time on Compat and it is to be hoped that we do not see its like again. It contains some technically challenging code and those who profess to have heuristic detection of this virus may well spend many hours wondering about whether they can achieve 100% detection. Users who get this virus are in for a very tough time, being unable to trust any of their data from the point of infection.

Let us just hope the taxman does not catch it!

XM/Compat

Aliases:	XM/Compatable, XM/Import.B.
Infects:	*Excel 5/7* spreadsheets during File Save operations.
Self-recognition:	Looks in the application library directory for OFF97COM.XLA.
Trigger:	On closing a speadsheet.
Payload:	Randomly modifies up to 1000 cells per spreadsheet.

PRODUCT REVIEW 1

Vet Net Surfer 98 v9.9.1

The Australian-made *Vet* has a well-deserved reputation for its lightning speed. Typically, it and *Norman ThunderByte* vie for the place at the top of that rating list in *VB* comparative reviews, both well ahead of the rest of the pack. Such speed, however, would not be much of a virtue were it coupled with poor virus detection. So, with the release of a version for a 'new' operating system, your intrepid reviewer subjected it to a thorough barrage of tests to see how it fared in this new environment.

The Package

Vet Net Surfer 98 (*Vet98*) is one of several shipping forms of *Cybec's Vet95/98* product. Apart from the box art, the main variations between the differently-named forms are the licensing and support options available for each. These obviously have cost implications. It would appear that apart from versions designed strictly for evaluation and OEM purposes, the *Net Surfer* option is the cheapest, and intended for the small office/home office market.

Although the flap indicated a CD, the box contained a manual and two permanently write-protected 1.44 MB diskettes. The registration card mentioned in the list of contents was obvious by its omission.

The setup program displayed some fairly involved licence terms as well as the usual 'you do not own this software', 'we are not liable for any damage…', etc. To make sense of the proffered terms (that you were required to accept before the program completes its task) you have to be aware of what kind of licence you have. Having looked through all this, matters were not entirely clear to this reviewer.

A *Cybec* representative explained that the *Net Surfer* range did not include any direct updates or upgrades. So long as users register, they obtain unlimited access to the subscription area of *Cybec's* web site and can download as many (or as few) data file updates and feature upgrades as they desire throughout the course of that licence – usually one year. Corporate licensing programs allow for site licences, delivered (rather than sought) updates and upgrades.

Documentation

The printed manual, at 58 pages, is not a heavy tome, nor should it be. The days when scanner manuals were expected to contain a historical description of computer virus development are – thankfully – long gone.

Eschewing the common notion that such chapters should be replete with screen shots that duplicate the visual environment of the installer, the opening chapter seems denser than its counterpart in many such manuals. This is not a bad thing though. Perhaps the authors took the realistic view that few people would read the manual before 'giving it a go' at least once. When the manual is referred to for such information, those doing so mostly read along as they install the software.

Following a logical progression from installation through a description of the main program interface to using and configuring the on demand and resident scanners and then onto the more specialized options, the manual is easy to follow and navigate. It suffers a few inconsistencies of editing. For example, in places, vestiges of material that more correctly applies to using *Windows 3.1x* remain, despite these interfaces having changed dramatically in *Windows 95* and subsequent versions.

The on-line help built into *Vet98* is somewhat idiosyncratic. Despite its dialog boxes being all but universally festooned with Help buttons, pressing F1 throughout the program – as this keyboard addict is accustomed to – did not see it forthcoming with assistance. The Help buttons lead to good, context-sensitive help, but it would be nice if this could be summoned at the press of the standard 'help key'.

Installation

That the setup program refused to run under *Windows 95* was not surprising for a *Windows 98*-specific product. The installation program allows a wealth of configuration options and the production of a network 'master' installation, from which many identical setups can be rolled-out. This functionality seemed odd in an avowedly single-user pack, its availability reinforcing the impression that one program is packaged under several guises.

Tracing a route from splash screen to welcome message to licence agreement, the *InstallShield* setup program should feel familiar. The all but expected 'typical or custom' option was offered next.

Selecting the Typical option requires very little by way of user interaction – the product is installed to the default directory in a sensible configuration or an existing installation updated. Once the files have been copied to the hard drive, the readme file is proffered. Next an option to make a

reference disk is made, then all local hard drives are scanned. Finally, it is suggested that the PC be restarted to activate the on-access components or to allow some of the updates to currently active processes to take effect.

The custom installation option differs primarily in that once the files are copied, a configuration wizard is run. This allows the configuration of the options normally available inside *Vet98* anyway. Setting some of them here is handy though, as a few options require a reboot to effect, so one restart can be avoided by configuring the program thus.

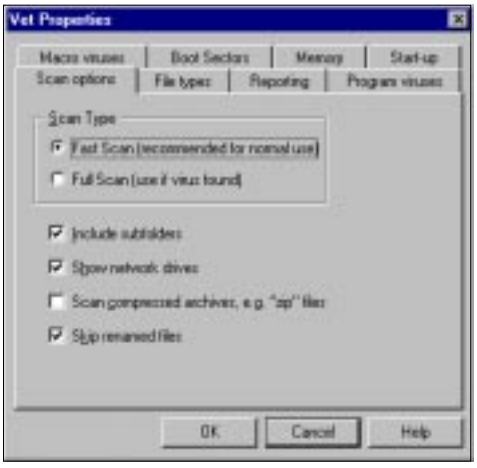
Configuration and Use

In its default configuration, *Vet98's* presence on a machine is mainly noted by its icon in the system tray. Should the mouse be left hovering over the icon for a few seconds, a popup displays the status of its two resident monitors (it has separately configured monitors for boot viruses and for file and macro viruses). Right-clicking drive, folder or file icons in Explorer will also reveal an option to scan the object with *Vet*. In the case of files, this only appears for those files whose extension is on the 'files to scan' list.

Right-clicking this icon opens a menu offering access to the main program window (as does double-clicking the icon), another method of displaying the status of the resident monitors and removing *Vet* from the tray. This option can be pre-configured as part of a master installation, and such options can be 'enforced' through password protection.

Vet98's main program window is divided into two main panes – one allowing the selection of drives, folders or files for scanning and the other displaying progress reports and scan summaries. This is an easily used interface and for a workstation user probably quite adequate. However, the lack of any mechanism to store and recall specific scan configurations would make things onerous for some typical network administration situations.

A healthy range of configuration options is available. From the Explorer-like settings affecting the browse window (on the View menu) to the configuration of email alerting of virus detection incidents, there is plenty to keep the gadget freaks busy. The Options menu holds the main program and resident protection configuration dialogs. For those new to the program or who are less familiar with anti-virus terminology, the Options Wizard allows the user to step through the configuration options, providing some direction and advice to the process.



The on-demand scanner has choices between fast and full scan modes, and scanning an editable selection of file types or all files. Log files can contain listings of just suspect files or all files scanned, and typical logging options such as append/overwrite, log-file size limitation and log-file name are provided.

On detecting a file virus the choice of actions is report only, clean, rename or delete. With suspicious files the options are report only, rename and delete. Two dialogs deal with these settings, allowing macro and program file viruses to be handled differently. Caution is advised with the delete option, as it is a 'destructive delete' – overwriting the entire file before unlinking it. Perhaps changing the option's name to 'Erase' would help here, as the destructive nature of the option is only mentioned in the manual and on-line help.

'Bad' boot sectors can be defined as sectors infected with a known virus, those plus invalid boot sectors (according to *Vet98's* heuristics) or both plus any not known to *Vet98's* library of 'known good' boot sectors. The scanning of boot sectors can be disabled separately and 'bad' boot sectors can be replaced, though with exactly what is not specified.

An interesting feature in *Vet* – barely necessary given its speed – is the progressive scan at system start up. Perhaps originally an idea to overcome user resistance to on-demand scanning at boot, this breaks up scanning your local drives into small-ish pieces that can be achieved quickly. The default is to scan 100 files at a time and then cycle back to the start once the last file is done. A niggle with the implementation of this is that although all other useful options can be set through the configuration dialog, the scan type used (fast or full) is not documented. Thus the only way to be sure of which is used involves digging around in the help for the command line switches and fully configuring the startup scan that way.

Provision is also made for recording the boot sectors of local drives with the Record Templates function on the Tools menu. In fact, this is done during the installation process and the recorded sectors are used during the startup scan integrity check. These templates are also copied to the Reference Disk should the creation of one – also from the Tools menu – be selected.

Attempts to access the menu function that restores recorded boot sectors result in a prompt for a password – a wise thing given the trouble that ill-advised use of such a function can cause. Apart from this Emergency menu option always being protected, a requirement for entering the password can be applied to all the configuration dialogs. In light of the growing popularity of tools such as Partition Magic, which allow resizing, moving and other like procedures involving disk partitions, users should be mindful of updating these templates after using such tools.

Updates and Alerts

New in this version is support for SMTP email alerts from the resident scanner. As more users come to rely on resident scanning, logging of virus incidents has often been lost as many products do not have good logging options for their on-access components. Apart from now having support for SMTP email in both on-demand and resident modes, the product supports writing log files in both modes. It can also be configured to pop up a warning dialog box whenever a virus is encountered in either scanning mode.

Support for incremental DAT file updates has also been built into the product. Initially only updates for macro viruses were available, but file and boot virus updates have been possible since v9.9.0. This reduces the size of the update and also means less time is required to install it.

All registered users receive a login name and customer number for one year's worth of access to the subscription content area of *Cybec's* web site. This allows download of any updates or upgrades that are made available there. Depending on the product type purchased, quarterly upgrades are provided. Corporate licensees may prefer more frequent updates without having to download them from the web and this service is available for a fee.

This feature was too new to test properly – the previous version of *Vet*, v9.8.5, was still posted on the web site a couple of weeks after v9.9.1 was received for this review! Wondering what would happen should an unsuspecting user attempt to 'upgrade' to an older version, the installer was run on a machine with v9.9.1 installed and active.

Surprisingly, the 'new' version ran, and without questioning whether a retrograde was really what was intended, it set up v9.8.5. Following a reboot, various VxD errors and the like suggested things were not right. Uninstalling this version then re-installing v9.9.1 put things right. This may seem petty, but allowing your users to mess up their machines like that is bad form. As it is easily fixed programmatically, it should be so future versions either retrograde cleanly or, probably better, refuse to retrograde, suggesting that first uninstalling the newer version is the preferred option.

Performance and Virus Detection

As mentioned at the outset, *Vet* has a reputation as a speedster and did not disappoint on that front. Taking just 91 seconds to scan the 5500 files of the *VB* Clean test-set, represents a data throughput of approximately 5.8 MB per second. That was in the default 'fast' scanning mode. Changing to Full mode, restarting the machine and repeating the test, it took 237 seconds. At a throughput of 2.2 MB per second, this is still considerably faster than many products and would place around mid-field in recent *VB* comparative reviews.

Vet98 scored well against the *VB* test-sets. It detected all 84 boot sector viruses from the In the Wild test-set, and did so under both on-demand and on-access test conditions. It is encouraging that there was no sign of the inconsistencies that many products have exhibited on this test in recent *Windows 95/98* comparative reviews.

The ItW File test-set was a little more challenging. The recent addition of Groov.B to the WildList has caused some products trouble. An early, polymorphic *Word 97* virus, it has gained some 'success' in spreading and several products have trouble detecting it reliably. All five samples were *Vet98's* only misses on-demand, resulting in a detection rate of 99.4% for the ItW File set. This result combined with the boot sector results, gave an ItW Overall detection rate of 99.5%. Detection by the on-access scanner was lower by dint of missing nine Marburg, and three TVPO.3783.A, samples (as well as those of Groov.B). These 'additional' misses were due to the on-access scanner not checking *Windows* screen savers (SCR files).

Against the other test-sets, *Vet98* also staged respectable detection. Polymorphic macro viruses were mainly responsible for the misses that provided 98.4% (on-demand) and 97.9% (on-access) in the Macro test-set, and 97.7% in the Polymorphic set. The latter result was reduced to 96.2% under on-access testing due to a third of the Marburg samples being in SCR files. 99.2% of the Standard test-set was detected under either test condition.

Conclusion

Speed lovers will not be disappointed with this new version of *Vet*, and its virus detection rates are at its typically high levels. Polymorphic macro viruses are apparently an area of current weakness in *Vet98*. With increasing interest in this line of 'development' among virus writers, and some of their creations successfully getting into the wild, it is hoped that this is addressed soon.

Technical Details

Product: *Vet Net Surfer 98*.

Developer: *Cybec Pty Ltd*, 1601 Malvern Rd, Glen Iris 3146, Victoria, Australia; Tel +61 3 98255600, fax +61 3 98860844, email info@vet.com.au, WWW http://www.vet.com.au/.

UK/European Distributor: *Vet Anti-Virus Software Ltd*, 342 Glossop Rd, Sheffield, S10 2HW, England; Tel +44 114 2757501, fax +44 114 2757508.

US Distributor: *Ontrack Data International Inc*, 6321 Bury Drive, Eden Prairie, MN 55346, USA; Tel +1 612 9375161, fax +1 612 9375815.

Availability: Pentium with 8MB RAM, 3MB of free hard disk space, *Windows 98*. Internet access to obtain updates.

Version Evaluated: 9.9.1.

Price: Single licence £50; 25 user £480; 250 user £3033 (all exclusive of VAT). For multiple product and site licences, and pricing outside Europe, please contact the appropriate vendor.

Hardware Used: 166MHz Pentium-MMX PC with 64 RAM, 4 GB hard disk; 3.5-inch floppy and CD-ROM drive. The machine can be configured to run *Windows 95*, or *98* by restoring an disk-image backup.

Virus Test-set: Complete listings of the test-sets used are at http://www.virusbtn.com/Comparatives/Win98/199811/test_sets.html.

SUBSCRIPTION RATES

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

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

Editorial enquiries, subscription enquiries, orders and payments:

Virus Bulletin Ltd, The Pentagon, Abingdon Science Park, Abingdon, Oxfordshire, OX14 3YP, England

Tel 01235 555139, International Tel +44 1235 555139
Fax 01235 531889, International Fax +44 1235 531889
Email: editorial@virusbtn.com
World Wide Web: http://www.virusbtn.com/

US subscriptions only:

Virus Bulletin, 18 Commerce Way, Woburn, MA 01801, USA

Tel (781) 9377768, Fax (781) 9320251

END NOTES AND NEWS

InfowarCon'98 will take place at the Mount Royal Hotel in London from 8–9 December 1998. On Monday 7 December optional, full-day tutorials will be held. The conference focuses on military operations, infrastructure protection, and the growing threat of high-tech terrorism and espionage. It is aimed at corporations, infrastructure firms, and finance, military, intelligence and law enforcement organizations. Registration is from 7.30am on Tuesday 8 December. For more details about the conference, contact organizers *MIS* in London; tel +44 171 7798944, fax +44 171 7798293.

***Network Associates Inc* (*NAI*) announces the recent development of AutoImmune technology**, soon to be integrated with its *Total Virus Defense* product range. The new technology is designed to detect, remove and create a cure for previously unknown viruses in coporate networks. It is currently to be found in the Enterprise Edition of *VirusScan v4.0*. For more information contact *Network Associates* in the UK; tel +44 1753 827500, or see http://www.nai.com/.

Trend Micro* announces the availability of *eManager, a set of Internet email controls the key element of which is a spam filter. It is being introduced in conjunction with *Trend's* Internet gateway virus scanning software. Prices for *InterScan VirusWall 3.x* start at £1295 for 50 users and from £259 for *eManager* for up to 50 users. For further details, contact Steve White at *Peapod*; tel+44 181 6069924 or email trend@peapod.co.uk.

***Quarterdeck* has launched an on-line store exclusively for its European customers.** The company's full product range is available at http://store.qdeck.co.uk/quarterdeck/ in your chosen European language and prices are quoted and can be paid for in local currencies. For more information contact Christine Allenet; tel +44 1628 666322.

Following news of the *IBM/Symantec* alliance, ***IBM* is to remarket the *Norton Anti-Virus Solutions Suite* through the *IBM/Lotus Passport Advantage Programme*.** For further details about this service, contact *Symantec's* UK Customer Service Centre; tel +44 171 6165600, or see http://www.symantec.co.uk/.

***Calluna Technology* has recently entered the data security market with the launch of *Hardwall* technology.** The company has recruited a new European OEM (original equipment manufacturer) Manager formerly at *Intel* to oversee the marketing of *Hardwall*, a hardware device claimed to provide anti-virus, anti-hacker and anti-system corruption functionality. For more information contact *Profile PR* in London; tel +44 181 9486611 or visit http://www.calluna.com/.

Secure Computing* has announced plans to extend its services** to provide UK customers with advice on Penetration Testing, Security Policy Development and Security Assessment among other topics. In an independent move ***Reflex Magnetics* has teamed with *Heimdall to provide services analogous to those just decribed. This is an effort to help corporate clients formulate a healthcheck for their networks. For details contact *Secure Computing*: tel +44 1753 826000 or *Reflex Magnetics* +44 171 3726666.

***Computer Security Institute* (*CSI*) has released details about its 9th Annual Network Security Conference.** NetSec'99 is to be held from 14–16 June, 1999, in St Louis, Missouri at the Hyatt Regency Hotel. Over 1500 computer and information security professionals are expected to attend the conference and its concurrent exhibition. For more details, contact *CSI*; tel +1 415 9052626, fax +1 415 9052218, email csi@mfi.com, or visit http://www.gocsi.com/.

Sybari Software* has released *Antigen 5 for Exchange, which it claims solves reliability and performance problems found in current, MAPI-reliant anti-virus products for the *Microsoft* mail server. *Antigen 5* is said to scan all mail messages, mailboxes and public folders in real-time, overcoming alleged synchronization issues seen in its competitors. It is priced at $4995 for a two-year licence for 250 users. For more details, contact the Director of Product Management Tom Buoniello; tel +1 516 6308503, email tom@sybari.com or visit the web site http://www.sybari.com/.

Symantec AntiVirus Research Center (*SARC*) has developed Bloodhound for Trojan Horses to combat the threat of unidentified password-stealing Trojan Horses. Contact Symantec Customer Services; tel +44 171 6165600 or visit http://www.symantec.com/.

The VB'98 conference proceedings are available on CD-ROM, priced at £150. For more information, please contact conference co-ordinator Jo Peck at the *Virus Bulletin* offices; Tel +44 1235 555139, fax +44 1235 531889, or email Joanne.Peck@virusbtn.com.