ISSN 0956-9979 JANUARY 2001

VIRUS BULLETIN

THE INTERNATIONAL PUBLICATION ON COMPUTER VIRUS PREVENTION, RECOGNITION AND REMOVAL

IN THIS ISSUE:

• **The 11th Virus Bulletin conference:** the long-awaited venue of VB2001 is finally announced and the call for papers goes out in earnest. See p.3 for details.

• **Our man in Japan:** Matt Ham reminisces about his recent trip to Tokyo for the 3rd Annual AVAR conference. His report starts on p.10.

• **Banking on it:** how does one of America's biggest financial institutions keep its networks virus-free? Max Morris explains the setup at *First Union* on p.14.

• **IMO:** Padgett Peterson makes a few wise predictions for the future of corporate AV protection while Eddy Willems considers the legal implications at home in Belgium. Our Opinion columns start on p.16.

CONTENTS

COMMENT
Expert Advice 2

VIRUS PREVALENCE TABLE 3

NEWS
1. Czech Out VB2001 3
2. ProLin around the Net 3

LETTERS 4

VIRUS ANALYSES
1. Harnessing Hybris 6
2. Drill Seeker 8

CONFERENCE REPORT
Tales from Tokyo 10

FEATURE SERIES
The Usual Suspects – Part 2 12

CASE STUDY
The State of the First Union 14

OPINION
1. Compiled Trojans … 16
2. The Lawful Truth 18

PRODUCT REVIEW
IKARUS virus utilities millennium edition v.5 20

END NOTES AND NEWS 24

COMMENT

Expert Advice

‘The glossier the brochure, the poorer the product’. This little gobbet of refined cynicism was initially aimed at the computer software industry at a time when, having won the digital vs analogue skirmishes, the Digital Division had fragmented into System Wars and were pitting DOS against CPM against Unix. Even in those days, it wasn’t how sharp or accurate your weapon was but how loud you could shout on the battlefield. This philosophy permeated later engagements – the Battle of Visicalc/*Lotus*, the Word Processor Wars, the Multi-user Massacres, the *Windows* Blitzkrieg, and so on.

“ … maybe the lunatics really are running the asylum. ”

A million or so Davids, known then as pimplies, all heaved their half-bricks (viruses) at the current Goliath. The result showed that while Goliath made a lot of noise, he was by no means invincible. In this instance, a hastily cobbled together Anti-Virus Brigade hurried to Goliath’s defence and, for a small consideration in ready cash, provided defence against missiles both real and imaginary. It was even rumoured that some doughty defenders manufactured missiles of their own just to demonstrate their skill at stopping them.

The pimplies are becoming wrinklies but the self-promotional enterprise they displayed is alive and well wherever experts congregate to dip their bread into the judicial gravy boat. This centres around the notion that virtually anyone these days can stand up in an English Court of Law and claim ‘expertise’. They can deliver words of great wisdom and remain unchallenged unless another expert can *prove* that what they pronounce is nonsense. One expert recently suggested that the presence of the text ‘alt.binaries.multimedia.erotica’ on a computer was proof, in his not-so-humble opinion, that the operator had been downloading paedophile material for several years. The fact that no paedophile material was discovered was because the defendant had cleverly overwritten it all by copying one hard disk to another. This defendant was convicted and sentenced to three months imprisonment solely on the expert’s evidence. That the sentence was subsequently reduced on appeal does little to calm one’s fears that maybe the lunatics really are running the asylum.

Those elderly readers who recall my attempts to hoist virus writers up the nearest flagpole, pending the arrival of the Old Bill with an invitation from Her Majesty to attend one of Her residential academies, will be aware that there is a point to all this. Examine your computers in fine detail, locate and tabulate any emotive words. Certainly these will include any of the newsgroup lists but attention should also be paid to other, less obvious words – ‘terrorist’ and ‘bomb’ would have the average juror reaching immediately for his thumbscrew and squirting a precautionary drop of *3-in-1* on his rack but ‘bang’, ‘fuse’ and even ‘arm’ might be equally compelling in the hands of the pseudo-expert. ‘Tobacco’, ‘smoke’ and even ‘puff’ might provide evidence for conviction as a dangerously pollutive radical, and the likes of ‘fox’, ‘hare’, ‘stag’ and ‘hounds’ do not bear thinking about – even if a conviction failed, publicity might well double the sales of Swan Vestas in your immediate locality. A soundex search for ‘incest’ produced, in one memorable instance, oh joy, ‘insex’ (the somewhat convoluted entomological reference only became apparent after much ‘experting’).

If you find any of these words, and reliable legal advice is called for here, it is no longer enough to buy a new hard drive and copy loads of *Microsoft* stuff all over it. This would simply provide grist for the expert’s mill when explaining to the jury just how devious and clever us criminals are. Who amongst us would be brave enough to try to explain that ‘cells’ and ‘blocks’ were part of a spreadsheet or that ‘daemon’, ‘spell check’, ‘wizard’ and ‘hex’ did not refer to naked midnight Hallowe’en raves on the local blasted heath? The only solution is to destroy your computer. Crush the hard drive, burn all the software, go and buy half a dozen notebooks and pencils – in short, get back to basics. It isn’t that I’m getting paranoid, I’m just wondering who this General Failure Error is and why is he reading my hard drive?

Jim Bates, Computer Forensics Ltd, UK

NEWS

Czech Out VB2001

VB2001, *Virus Bulletin's* 11th annual conference and exhibition, will take place on Thursday 27 and Friday 28 September 2001 at the Prague Hilton, in the capital of the Czech Republic. The Welcome Drinks reception is planned for the evening of Wednesday 26 September and the traditional Gala Dinner for Thursday 27 September.

Virus Bulletin is currently seeking submissions from those wishing to present papers at this year's conference. As usual, there will be two concurrent streams of sessions – corporate and technical. Abstracts of approximately 200 words must reach the Editor by Friday 23 February 2001. Submissions received after this date will not be considered. Please send your abstracts (in ASCII or .RTF format only) to editorial@virusbtn.com. Authors are advised in advance that the submission date for completed papers selected for the conference programme will be Friday 29 June 2001. Companies wishing to enquire about sponsorship opportunities and/or exhibition packages are encouraged to contact Karen Richardson; VB2001@virusbtn.com ▮

ProLin around the Net

Botica Conroy & Associates, *Symantec's* New Zealand PR firm, was hit by ProLin mid-afternoon on 4 December 2000. Forty-six minutes after the virus mass-mailed itself to *BCA's* extensive mailing list, *BCA* distributed *Symantec's* description of the virus dated 30 November. Nice try guys.

Win32/ProLin mass-mails itself to all addresses in its victims' *Outlook* address lists with the Subject line 'A great Shockwave flash movie', and message body 'Check out this new flash movie that I downloaded just now . It's Great Bye' and an attachment named CREATIVE.EXE. ProLin also copies itself to the *Windows 9x* startup directory, and has a file-moving payload. All .JPG, .MP3 and .ZIP files are moved to the root of the C drive and renamed by adding 'change atleast now to LINUX' to the extension.

Win9x machines with many such files and FAT32 C drives will boot very slowly once this has happened. This is due to the size extension of the FAT32 root directory from all the files moved to it and scanning C:\ for DBLSPACE.INI and DRVSPACE.INI files (to assign drive letters if one of those compression drivers is needed to access the drive). ProLin writes a log of the files it moves and renames in the file MESSAGEFORU.TXT in C:\ – this can be used to match the files in C:\ with their original locations. Several AV vendors have tools that use this file to restore files moved by ProLin. The file also contains a rant against the user. On *Win9x* machines, once the moved and renamed files have been restored, the slow booting should be corrected by defragmenting the partition, as this will also shrink the root directory to a more typical size ▮

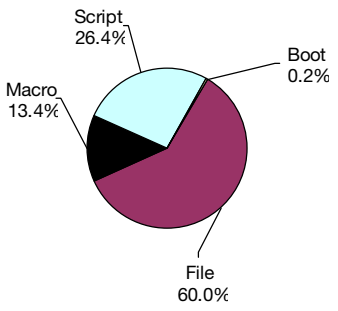
Prevalence Table – Month Year

Virus	Type	Incidents	Reports
Win32/MTX	File	1178	28.5%
Win32/Navidad	File	923	22.4%
LoveLetter	Script	739	17.9%
Kak	Script	298	7.2%
Divi	Macro	120	2.9%
Win32/Hybris	File	110	2.7%
Laroux	Macro	90	2.2%
Win32/QAZ	File	81	2.0%
Ethan	Macro	62	1.5%
Marker	Macro	58	1.4%
Win32/Ska	File	50	1.2%
Win32/Pretty	File	46	1.1%
Thus	Macro	30	0.7%
Stages	Script	27	0.7%
Class	Macro	24	0.6%
Tristate	Macro	21	0.5%
Win32/Funlove	File	19	0.5%
Win32/MSInit	File	19	0.5%
Win95/ClH	File	16	0.4%
Netlog	Script	15	0.4%
Jini	Macro	14	0.3%
Cap	Macro	13	0.3%
Melissa	Macro	13	0.3%
Eight	Macro	11	0.3%
Others [1]		152	3.7%
Total		4129	100%

[1] The Prevalence Table includes a total of 152 reports across 54 further viruses. Readers are reminded that a complete listing is posted at http://www.virusbtn.com/Prevalence/.

In order to avoid a distortion of the figures, data for the 'self-reporting' W97M/ColdApe virus (totalling 503 reports in November) have been omitted from the table this month.

Distribution of virus types in reports



LETTERS

Dear Virus Bulletin

Fed Up with IT

As an IT professional, I must take issue with your instructions on what to do if you receive a virus warning. Your instructions state that if you are unsure, you should forward the warning to your IT administrator, or *VB*. This applies not only to potential hoaxes, but to real warnings as well. DO YOU REALIZE HOW MANY OF THESE I GET A WEEK? I think people should be advised that IT administrators routinely subscribe to virus services that issue a warning long before their network of friends could possibly get around to it.

Virus warnings should be killed. Period. End of story. If they cannot identify if it is a hoax or real, then send it to someone like you guys, but leave IT out of it. We can't really act on such a warning anyway without some verification. The real virus warning problem is getting to be as bad as or worse than the hoax warnings – no matter what the company policy, someone will still send them. Putting information such as this on Web sites such as yours would go a long way towards curbing this problem, just as it has for hoax warnings. Thanks for listening.

Larry Friddle
KHBS/KHOG TV
USA

Playing Fair?

This letter is in response to Ray Glath's article 'Playing the Odds' in November's magazine (p.12). Mr Glath, you are right, all AV companies hold on stupidly to their old-fashioned technology – scan, scan, scan … update, update, update – and it seems to be a vicious circle that never ends. But the truth is that AV vendors have never really wanted any other 100% efficient 'New Technology' (or technologies) because the Virus Phantom – the only reason for us to purchase 'new' AV products and updates – will then be forever lost. In this case, we could get a perfect and steady protection on our PCs, without a need for frequent definition updates, and … they would lose their jobs.

I do believe that the 100% detection target should have been reached after 10 years, if not for all computer viruses then for the known and most dangerous classes of them, surely. You also write about this in *VB*.

It is not only a technical question 'How do we get 100%?', but more a political question 'Why do we not get 100%?'. Don't they see any other technological way to protect desktops from viruses? Of course they do or, I would say, professionally *must do*! Mr Glath, you write: 'Users must demand more from their vendors', but how? I believe that advanced security concepts, new security features and security interfaces cannot be sold separately as add-on tools and must be built directly in the desktop operating systems. I consider IT security not as a number of security components or AV products that a user can like or not like, but as an important public security service in the Internet age, to allow work on a PC to be secure, productive and effective.

It is well-known that increased security always reduces performance. IT security is not our primary need and it does nothing useful as far as data processing on our PCs goes. That is why we need to take it off the marketing battlefield. We must change our views and norms in order to be able to solve the problems for our customers.

It is also a fact that one software product can be marketed successfully, but a security software product on its own is not the same. These problems must be solved, but cannot or will not be solved in the near future. Or have you a practical idea of how users can demand more from, for example, *Microsoft Corporation*?

So, today we vendors and users are actually playing a marketing game in which the rules are never changed. Furthermore, we must base our needs on the security solutions of the same big AV vendors, when those vendors never really intend to give us 100% protection.

Sometimes I believe that the AV industry, virus writers and hackers are one big family working tightly together. If so, then it is logical to ask the next question: 'Are they all playing against us?'.

Eugene Bytschkow
Deutsche Post
Germany

No Competition

The mysterious East welcomed a diverse collection of AV researchers and delegates at the third *AVAR* (*Association of Anti-Virus Asia Researchers*) international conference. Personally, I think this is a significant milestone in the development of *AVAR* as it combined a trebling of size with increased international participation.

AVAR was established in 1998 at the first meeting of interested anti-virus researchers in Hong Kong. It set out the mission of *AVAR* as preventing the spread and damage caused by computer virus, and developing co-operative relationships between anti-virus researchers in Asia. The second conference, in 1999, was held in Korea and had about 50 participants. This conference, the third, was held in Tokyo and set some significant records: it was the first international AV conference to be held in Japan; it had the

first speaker from a Chinese government body and from a Singaporean government body on the programme; and there were 180 participants, the highest number yet.

The involvement of government bodies is significantly different from the *VB* conferences – 30% of the *AVAR* speakers were related to a government. Governments are important in anti-virus concerns; at worst, they can pass inappropriate and damaging laws, at best, they can provide advice unpolluted by marketing hype so that better understanding and co-operation is beneficial to all.

In between speeches, a significant part of my time was occupied by assisting people to meet, with a translator if necessary. Many researchers met international counterparts for the first time, and the importance of this in a field where personal trust is such a sensitive issue cannot be overestimated. I am sure we will see increased participation in international forums by Asian researchers as a result. Such co-operation is vital because viruses are an international issue that includes Asia – CIH and LoveLetter proved that.

I foresee that the *AVAR* conference will be one of the major international AV conferences. 'One of' because there is a need for conferences distributed around the world. Not everyone can jaunt around the world for these things, and on-line communications cannot replicate everything that happens at conferences yet (and I note, Steve White in his 'VB 2010 – A Retrospective' at VB2000, predicted many things, but did not suggest that conferences would disappear any time soon).

This letter would have been better if it had been written by Seiji Murakami, Chairman of *AVAR* and the conference Organiser, but the choice was to have it in Japanese by the deadline, or translated too late. However, this does give me the opportunity to thank Seiji, for all the hard work he and his team put into making *AVAR 2000* a success. I can also invite you all to *AVAR 2001*, which will be organised by me in Hong Kong. I hope to meet many of you there.

Allan Dyer
Yui Kee Co Ltd
Hong Kong

And the Good News is …

There are still few days left until New Year's Eve and I am already reading articles about familiar malware. While I don't believe things will change in those few days, Christmas provides us with the opportunity and ability to think about certain issues all over again.

In the second half of November 2000 Poland was hit with Win32/BleBla.A. Within the next few weeks we were hit again with the .B variant. This made me recall the good old days when worms were something from the Unix world and most viruses were written in Assembler. One might think that so much has changed that what started more than a decade ago doesn't have any impact on today's security scene. That is so false. I am still receiving messages infected with Win32/BleBla.A or B, despite the fact that most AV products can detect it successfully. Forget commercial products! A Polish vendor gives free access to their on-line scanner. You don't even need to register to use it and people *still* don't run AV software. The BleBla worm wouldn't be an issue if users would patch their system. Those patches were available from *Microsoft* long before BleBla was widespread in Poland.

Users not applying patches is nothing new. On the other hand, we could ask why software developers push users to apply patches to be secure when some problems have their roots in design or poor quality of code. *Microsoft* had some good ideas – anti-virus protection in *Word* or an AV API for *Microsoft Exchange*, to name a few. They could do their jobs quite well I guess, if the implementation was good. Unfortunately it isn't. People without security knowledge and experience shouldn't design such solutions.

Is there anything new? First of all, Win32/BleBla is one of very few creations from Poland. Polish virus authors never really caught up with *Windows 9x*. Could BleBla mean that situation is changing? Truly, I do not know. What is important is the new face of security: it is becoming more like a race than anything else. Short response times from vendors are as important, as worms – assuming they are properly written – will spread very fast around the globe.

Any other scary thoughts? Unfortunately, yes. The rapid development of DDoS tools. Win32/Doser proved that viruses could be used as a propagation platform for building DDoS networks. The same technology can be applied to worms – I cannot imagine a more efficient method of building DDoS networks.

Another rising problem is *Linux*. Mainstream *Linux* distribution is getting bigger and more user-friendly. This means ease of installation but more complex software. Neither goes together with security. As *Linux* gets more and more popular, more *Linux*-aware malware will appear. Oh, and don't forget malware for mobile devices and PDA. The first virus for *PalmOS* was primitive but wasn't that the case with DOS, macro and *Linux* viruses?

Was there any thing positive this year? If you attended VB2000, you would know the answer. I had a very good time and I made a lot of friends (I still owe a few people some drinks, actually). On a less personal note, you could see that anti-virus companies are trying to work together more closely. REVS is just one example of this.

VB2000 was the best conference yet and there are plenty more to come. *Micro-soft Windows 2000 Server* with Service Pack 1 isn't that bad after all. *Linux* viruses aren't so widespread. My *PalmPilot* isn't infected by a new virus yet. Some worms can be stopped with old patches. Don't you think life is getting quite boring?

Aleksander Czarnowski
AVET Information and Network Security
Poland

VIRUS ANALYSIS 1

Harnessing Hybris

Andy Nikishin
Kaspersky Lab, Russia

Everyone knows that Friday 13th is a frightful day, but did you know that Saturday 14th is an even worse one? On Friday 13 November we saw W32/Sonic and W32/Navidad introduce themselves. They are rather complex worms and use some interesting features.

However, on Saturday we received samples of W32/Hybris. Unfortunately, this is even more complex and uses the same ideas, but this time optimised.

W32/Hybris

This is an Internet worm that spreads as an attachment to outgoing emails. The worm only works on Win32 systems. It contains components (plug-ins) in its code which are executed depending on the worm's needs, and these components can be upgraded from an Internet Web site or newsgroup. The major versions of the worm are encrypted with a semi-polymorphic encryption loop.

Hybris and some of its plug-ins contain the text strings:

```
HYBRIS
(c) Vecna
```

How it Works

The worm's main target on PCs is the WSOCK32.DLL library. While infecting this file Hybris takes the following steps. It writes itself to the end of the last file section where it hooks the connect(), recv() and send() functions (by modifying the export table). Then it modifies the DLL entry routine address (a routine that is executed when a DLL file is being loaded) and encrypts the original entry routine.

If the worm is not able to infect WSOCK32.DLL (for example because it is in use and is locked) it creates a copy of that library with a random name, infects it, and writes a rename instruction to WINNT.INI file:

```
[Rename]
D:\WIN98\SYSTEM\WSOCK32.DLL=
D:\WIN98\SYSTEM\AIAHEJOE
```

As a result, WSOCK32.DLL will be replaced with the infected image on the next *Windows* startup. The worm also creates a copy of itself with a random name in the *Windows* system directory and registers it in the RunOnce Registry key in the HKEY_LOCAL_MACHINE or HKEY_CURRENT_USER section:

```
\Software\Microsoft\Windows\CurrentVersion\RunOnce
 {Default} = %SystemDir%\WormName
```

In this example, WormName is a random name, such as CCMBOIFM.EXE, LPHBNGAE.EXE or LFPCMOIF.EXE. There can only be one possible reason to register an additional worm copy in the RunOnce Registry key. If WSOCK32.DLL is not infected on the first worm run, and its infected copy is not created for whatever reason, the RunOnce worm copy will have another go at the next *Windows* restart.

W32/Hybris intercepts *Windows* functions which establish a network connection, including the Internet. For that reason, the worm is able to monitor data as it is sent and received, and scans it for email addresses. When an address, or a list of addresses, is detected, the worm waits for a while and then sends infected messages.

Plug-ins

The worm's functionality depends on the plug-ins that are stored in its body and encrypted with an RSA-like strong crypto algorithm with a 1,024-bit key. It also uses a 1,023-bit RSA signing with 128-bit hashing function. This makes module faking practically impossible. There are up to 32 versions of 11 different plug-ins in the various worm versions. These plug-ins perform different actions and, what is more, they can be updated from the Web site http://pleiku.vietmedia.com/bye/.

The functionality of the complete worm depends on the fact that its host can upgrade plug-ins from the Web page. These plug-ins are also encrypted with RSA-like crypto. In addition, the worm can update its plug-ins by using the alt.comp.virus newsgroup. If the worm is active on a machine, one of its plug-ins connects to a news server (using a randomly selected server – there are more than 70 addresses on the list), converts its plug-ins to newsgroup messages, and posts them there. The messages have random Subject headers, for example:

```
encr HVGT
GTeLKzurGbGvqnuDqbivKfCHWbizyXiPOvKD
encr CMBK
bKfOjafCjyfWnqLqzSTWTuDmfefyvurSLeXGHqR
text LNLM
LmnajmnKDyfebuLuPaPmzaLyXGXKPSLSXWjKvWnyDWbGH
text RFRE rebibmTCDOzGbCjSZ
```

The first four characters form the plug-in name and the following four make up the encoded plug-in version. As well as sending them, the worm reads these messages from alt.comp.virus, gets the plug-in name and version and compares them with the plug-ins it is currently using. If the newsgroup has a message with a higher plug-in version, the worm extracts it and replaces the existing one.

Hybris also creates these plug-ins as disk files in the *Windows* system directory. Their names and extensions are

constructed from randomly selected capital letters from A to P (matching the 8.3 format), but the worm keeps being able to access them:

```
BIBGAHNH.IBG
DACMAPKO.ACM
GAFIBPFM.AFI
IMALADOL.MAL
MALADOLI.ALA
```

There are currently 11 different plug-ins currently known for the Hybris worm, all of which have different functions. The first infects all ZIP and RAR archives on all available drives from C to Z. While infecting, the worm renames EXE files in the archive with .EX$ extensions and adds a copy of itself with a .EXE extension to the archive (a companion method of infection). The second sends messages with encoded plug-ins to the alt.comp.virus newsgroup, and gets new plug-ins from there. A third shows a large animated spiral on 16 or 24 September of any year, or at the 59th minute of any hour of any day in the year 2001. This animated stuff is very difficult to close because it registers itself as a hidden process (service) using the RegisterServiceProcess() function. Moreover, this plug-in uses one more trick to hide itself in memory – it hooks the Process32First() and Process32Next() functions and hides its own process during process enumeration.

The fourth plug-in spreads the worm to remote machines that have the SubSeven backdoor Trojan installed. The plug-in detects such machines on the 'Net, and by using SubSeven commands uploads a worm copy to the machine and spawns it in there. A fifth plug-in encrypts worm copies with polymorphic encryption loops before sending a copy attached to email.

The sixth, actually two separate infector plug-ins, affect DOS EXE and *Windows* PE EXE files respectively so that they become worm droppers. When run, they drop a worm EXE file to the TEMP directory and execute it. In affecting DOS EXE files, the plug-in adds dropper code and the worm body to the end of the file. In *Windows* PE EXE files, the plug-in compresses the original code and writes the virus body and compressed code to the code section (if it is big enough). It is possible to clean these files.

The plug-in neither touches the file header (including the entry point address) nor increases the file size. Moreover, it has an anti-CRC (checksum) routine that fills in special data in the plug-in code so that the file CRC becomes the same for a few commonly used CRC algorithms. That means that some integrity checkers will not detect changes in the affected files i.e. the file length and file body CRC stay the same as on a clean file.

Lastly, the seventh plug-in randomly selects Subject, Message text and Attach name while sending worm copies with email messages from: Hahaha <hahaha@sexyfun.net>. The subjects include: 'Snowhite and the Seven Dwarfs – The REAL story!', ' Branca de Neve porn ?', ' Enanito si, pero con que pedazo!' and ' Les 7 coquir nains'.

Message texts can be in several different languages, French, English, Portuguese and Spanish. The names used for the attachments differ according to the language version, but the English ones often include: SEXY VIRGIN.SCR, JOKE.EXE, MIDGETS.SCR and DWARF4YOU.EXE.

Depending on the plug-in version, the message attachment subject is a random combination of words, again differing according to the language version, including the names Anna, Raquel, Xena and Darian, to name a few. The attachment name is randomly chosen from a list of 40 which include FAMOUS.EXE, SEXY.EXE, PLEASURE.EXE, ASIAN.EXE, BLACK.EXE, BLONDE.EXE and AMATEURS.EXE. Others are often of a crude nature, something of a giveaway in this respect.

Conclusion

The Internet is getting increasingly huge and ever more speedy. This presents hackers with the chance to build more complicated Internet viruses. Déja vu! I wrote these sentences a month ago!

Internet worms are the most frequently occurring malware to date. According to the November 2000 WildList of the 21 most frequently reported viruses, 12 of them are mass-mailers or worms. This is a very, very dangerous trend. I do not like to repeat myself but I have to say again – never run programs from email attachments!

W32/Hybris

Aliases:	W32/Hybris.22528.dr, I-Worm/Hybris, W32/Hybris.gen.
Type:	Win32 worm which spreads as an email attachment.
Self-recognition in WSOCK32.DLL:	If the file size can be divided by 18 with remainder 16, this file may be infected.
Possible Payload:	Displays a large animated spiral in the middle of the screen on 24 or16 September of any year or at the 59th minute of any hour of any day in 2001.
Removal:	Use a reliable anti-virus scanner to disinfect WSOCK32.DLL. Other files detected as W32/Hybris contain only the virus body and must be deleted.

VIRUS ANALYSIS 2

Drill Seeker

Péter Ször
SARC, USA

At the end of last year we saw several variants of the W95/Drill virus. It uses the Win32 API and runs in user mode but only works under *Windows 9x*-based systems. However, the polymorphic engine of the virus, called TUAREG, sets it apart from the average 32-bit polymorphic virus. Drill is packed with functionality like per-process residency, a retro-mechanism and an activation routine in a huge 14–18 KB assembly-written virus body, depending on the variant. It implements anti-emulation as well as anti-heuristic features.

Initialisation

W95/Drill is executed via the main entry point of an infected Portable Executable application. First the virus decrypts itself. It is encrypted with two layers of polymorphic code. The first decryptor is very long (several KBs) and placed in the original code section of the application named '.text'. The second decryptor is in the last section at the start of the virus body. This one is short but also polymorphic. Basically, the TUAREG polymorphic engine supports two different polymorphic decryptor generators.

Eventually, the virus is decrypted, but in some cases several million instructions need to be executed. This makes the use of code emulation more difficult. Initially, W95/Drill gets the addresses of all the KERNEL32.DLL APIs it needs to use later on. The list is impressive, considering that there are 34 of them (such as GetProcAddress(), CreateFileA(), CreateProcessA(), FindFirstFileA(), FindNextFileA(), etc). Their names do not appear in the decrypted virus body because the virus uses only checksums of the APIs called. This routine is protected with Structured Exception Handling. If an exception should occur, the virus simply executes its host application. After this, the virus calls its direct action infection routine.

Direct Action Infection

W95/Drill checks if the SFC.DLL (System File Checker) library can be loaded. If it is available the virus gets the address of the SfcIsFileProteced() function. This is because *Windows 98's* second edition supports the SFC just like *Windows 2000*. The virus tries to avoid infecting files that are protected with SFC, a mechanism we see in viruses that try to spread on *Windows 2000*.

The virus loads the IMAGEHLP.DLL (if available) to get access to its CheckSumMappedFile() API in order to be able to recalculate the checksum of an infected file and place it into its header properly. Drill is a retro virus. Before any attempt to infect a file in a directory, it looks for and deletes the checksum files of various anti virus software such as AVP.CRC, ANTI-VIR.DAT, CHKLIST.MS and IVB.NTZ. That happens even if the files are read-only since the virus changes the attributes of the files.

Then it looks for files with .EXE, .SRC and .CPL extensions. However, it does not infect every file it could. W95/Drill uses a random infection algorithm. It will skip some of the files without any attempt to infect it. However, in other cases, the file infection routine is called. The same directory infection routine will be called for the current, *Windows* and *Windows* System directories respectively.

Infection of Portable Executable Files

The infection routine is rather complex. First, the virus checks the name of the file. If it is a known anti virus file the virus will not infect it. Anything that starts with 'tb', 'cs', 'f-', 'pa', 'dr', 'no' or contains the letter 'v' will not get infected. Next, the virus checks if the file is protected by the SFC and skips the infection completely if it is. Otherwise, it zeros the attributes of the file in order to be able to infect read-only files. After this, the virus checks if the file is indeed a PE application.

Drill then starts to traverse the section headers. It checks if the file has a '.text' (code section), a '.bss' (global data section) or '.reloc' (relocation section) and saves their offset for later use. If the file does not have a section named '.text' Drill will not infect. Thus, Drill will not infect a Borland-compiled application that has a code section with the name 'CODE'.

If the file does not have a section named '.reloc', the virus will check if the last section is a '.rsrc' (resource) section. If the last section is 'reloc', then the virus will turn off the relocations and rename the last section with a random name. The name is either five random letters starting with '.' or a section with the name '.?text' where ? could be any character of the alphabet. If the last section is not '.rsrc' and the file does not have a relocation, the virus will not try to infect. This way, Drill avoids possible double infections.

Otherwise, Drill will create a new section in the section table using the above algorithm. The characteristics of the section will include the flags MEM_EXECUTE and MEM_WRITE. The virtual size as well as the physical size of the last section is set to 0x8000 (32768) bytes – rather large, but the virus needs to save the original content of '.text' section that will be overwritten by the first polymorphic decryptor.

The first two versions of the virus only used 0x6000 as the physical size of the last section. Regardless of size the virus might not make the file bigger.

The virus also checks if the code section is long enough and compares that to 13,988 in the latest variant (1.2). This is because the first polymorphic decryptor will be rather long and placed into the code section of the application. Obviously, this is an anti-heuristic infection. The main entry point of the host will be changed to point to the start of the code section. Then, a couple of polymorphic engine functions are called and finally the infected file is written back to disk with the original file date stamp and file attributes. The checksum field of the PE header will be recalculated with the CheckSumMappedFile() API.

Polymorphic Engine and Anti-emulation Tricks

Drill's polymorphic engine is complicated. It has the support for two different polymorphic decryptors. Both layers support XOR and SUB encryption methods – the virus will select a combination of them. The first polymorphic decryptor is several kilobytes long. It has the support for 'do nothing' loops as well as random memory writes.

Interestingly, the virus pays attention to the '.bss' sections. If there is a global data section the virus will generate writes to that area. Some emulators might not be able to handle the situation properly since the physical size of the '.bss' section is typically set to zero. This section is at least page-size, however, and can be written, though some applications might not appreciate the changes.

Before the first decryptor is built Drill uses an interesting function. It checks a list of 32 APIs in the import address table of the host. More exactly, this table has 28 active API names. The virus uses a CRC calculation of API names here. One API CRC is set to -1 in the table, i.e. it is not active. Three other CRCs will not resolve to any known KERNEL32.DLL APIs of any *Windows 9x* release including *Windows ME*. The remaining set of APIs is:

GetCommandLineA(), GetStartupInfoA(),
GetEnvironmentStrings(), GetVersion(),
GetModuleFileNameA(), MulDiv(), GetACP(), GetOEMCP(),
GetCPInfo(), GetStdHandle(), GetLastError(), GetLocalTime(),
GetSystemInfo(), GetCurrentProcess(), GetCurrentThread(),
GetConsoleCP(), GetCurrentDirectoryA(),
GetWindowsDirectoryA(), GetSystemDirectoryA(),
GetDriveTypeA(), GetComputerNameA(), IsBadWritePtr(),
GetTickCount(), IsBadReadPtr(), IsBadCodePtr(),
LocalHandle(), LocalSize(), LocalFlags()

All of these are KERNEL32.DLL APIs. If there is an export to any of them in the host application's import address table, Drill will save a reference. The polymorphic engine will use these references later on. The virus will be able to make a call to any of these available imported functions. The polymorphic engine will support the proper number of parameters on the stack to make the call possible. After that, the virus will place code to check the proper or improper return values returned by the actual function called. This way it forces a 'proper' environment and thus this function is implemented against emulators. First-generation 32-bit emulators might not be able to emulate even a subset of the Win32 APIs.

However, several emulators that I know have the ability to be extended with any APIs that a polymorphic virus uses to challenge emulators. This was predicted by several AV researchers. W95/Drill is the first virus to do this with part of the TUAREG engine v1.2. The list of APIs could be changed in the virus forcing the emulation of a different subset. This makes detection of the virus more difficult.

Activation Routine

After infecting directories, Drill checks the system date. If it is a Friday before the 8th of any month or between the 14th and 22nd of any month the virus will activate. Drill loads the ADVAPI32.DLL and gets the addresses of five registry APIs and changes the start page of *Internet Explorer* and *Netscape* to www.thehungersite.com. After checking the activation routine, the virus hooks the import address table of its host application to become per-process resident, a technique first used by W32/Cabanas .

Per-process Residency

Drill hooks GetProcAddress() in order to return the original API addresses to the host. It will also hook a set of APIs and call its infection function from them. This way, every time there is an access to a PE file by the host application the virus will try to infect. Finally, the virus is loaded and executes its host application. The large part that is overwritten in the '.text' section is written back from the end of virus body to its place. The virus does not handle cases where the import address table is placed in the '.text' section. Apparently, some *Microsoft* applications are compiled this way and as a result, infected files like that will crash since the System Loader will patch the actual polymorphic decryptor of the virus.

Conclusion

Several AV products did not detect Drill even at the end of December 2000. Some researchers were taken by surprise, others need to take the time to go through the dirty details of the virus. I hope the details here will help them implement appropriate detection. The detection of such viruses is rather difficult. Their repair is state of the art.

W95/Drill

Aliases:	Tuareg, Mental.
Type:	Win95 PE appender.
Self-recognition in files:	The virus checks if the last section is not named '.reloc' or '.rsrc'.
Payload:	Changes the start page of *Explorer* and *Netscape* to www.thehunger.com/.
Removal:	Replace infected files from clean backups.

CONFERENCE REPORT

Tales from Tokyo

Matt Ham

The world of conference hotels is a small one. Outside, Tokyo is a definitely foreign city, but inside the woolly spectacle that is Nick FitzGerald greets me with the same climate-based complaints as were ever the case in his days as Editor of *VB*. It was this home-from-home event that began November's *AVAR* (*Association of Anti-Virus Asia Researchers*) conference for me – quite a shock after having played the tourist for a couple of days. Allan Dyer, in his letter on p.4, has covered the history and background of the event – subjects to which we will return later in this conference report. For now, the people and papers at the conference are sufficient subject matter.

Comments have been made to the effect that several anti-virus organizations and initiatives resemble old boys clubs. This is, to a certain extent, true of the public face of anti-virus as a whole. When contacting an anti-virus company, soliciting products or publishing articles, the same, usually bearded, faces turn up with startling regularity.

Thus, the first great novelty that *AVAR2000* had to offer was the cadre of usual suspects being outnumbered by new faces. Admittedly, some of these names turned out to be known to me through email, but there were also ample opportunities for new contacts to be made and several lapsed contacts renewed. As a result, there should be some newcomers to the various *Virus Bulletin* Comparatives in the near future.

The conference papers were presented in either Japanese or English, which could have posed something of a problem had not the simultaneous translation been of admirably high quality. The AV industry has a patois all of its own, which must have been a major issue for the translation staff, though no signs of this filtered through to the audience, at least as far as translations into English were concerned.

The speakers covered a wide range of subjects – some general patterns did emerge, which again will be left for later. For now, the specifics are the order of the day. This latest conference was *AVAR's* largest yet, with far more attendees than the previous conferences in Seoul and Hong Kong. A collection of newcomers was present and the opening address was directed primarily at them, including me, the *Virus Bulletin* representative. The speech by Seiji Murakami, *AVAR* chairman, ran through the history of *AVAR*, an association first started by him after his own company, *Jade*, was swallowed in one of *NAI's* frequent feeding frenzies. Joined in the beginning by the seemingly ubiquitous Allan Dyer and Korean developers Dr Charles Ahn (*Ahnlab*) and Seok-Cheol Kwon (*Hauri*) the group has now expanded considerably.

The core *raison d'être* of *AVAR*, however, remains the same. The organization forms a co-operative group for Asian anti-virus researchers, with the aim of reducing the virus threat throughout the area.

The most noticeable feature of the conference was the number of government organizations represented, a trend which began with the keynote speech and continued throughout. Since these government bodies are not solely devoted to the field of viruses this also led to many of these papers being slightly more wide-ranging, covering parts of associated security fields in addition to their core themes.

The keynote speech was given by Mondo Yamamoto of the *Japanese Ministry of International Trade and Industry* and set the tone for the rest of the governmental speakers. It was somewhat surprising to hear in this speech the degree to which virus incidents have been studied in Japan – a study made possible by the obligation to report viral incidents to the government. It was also explained how this control is being extended, with budgets being increased to further the implementation of new technology and new legislation in the area. With the depressing state in the UK, where only a decrease in privacy seems to be planned in the realm of IT laws, this seemed a refreshingly enlightened plan from the details presented here.

The same can be said of the presentations by Taisuke Hatsukawa of the *Japanese National Police Agency* and Hyun Woo Lee of the *Korean Information Security Agency*. The former built upon the information in the keynote speech, being more concerned with the legal ramifications of virus control, while the latter, in a change of published topic, described future plans to deal with the current mail worm epidemics. Both showed that the countries involved are planning in a public way for future problems.

In addition, the more technical aspects of the problem were addressed, with *Microsoft's* Randy Abrams' tales of the perils related to test procedures being particularly close to my heart. Allan Dyer covered problems with anti-virus implementation while Righard Zweinenberg of *Norman* and *Symantec's* Motoaki Yamamura between them presented an overview of current and future threats likely in the area. Then *EICAR's* chairman Rainer Fahs introduced his organization to a new audience in his presentation.

There was one surprise. This was the unfamiliar meekness of Nick FitzGerald, as he explained in detail how to have virus exchange sites removed without once becoming incandescent with rage nor venting his spleen on the virus writers with his usual passion.

Mention was made of the similarity of convention hotels the world over, though Tokyo's Shinagawa Prince did offer one attraction less common in the West, that being Japanese

cuisine. Although Western food was available, some delegates embraced the local delicacies wholeheartedly – a certain Doctor Bontchev was a great fan of the green tea ice-cream, much to the disbelief of the more strait-laced *NAI* contingent. Rumours that one delegation visited the rather more dubious confines of a Geisha house were not pursued for fear of exposing internationally renowned anti-virus figures to puritanical witch hunts.

Computer Associates sponsored the evening social gathering on 28 November which was hosted by Seiji Murakami and Nick FitzGerald. The event resulted in a hotbed of mingling and became the precursor to late-night plotting and planning. The food was again Western and Japanese in style, though full marks all round were awarded to the sushi dishes on offer.

The second day, Wednesday 29 November, saw further presentations by government representatives, this time from Zhang Jian of the *Tianjin Quality Testing and Inspection Service of China* and Martin Ku of the *Infocomm Development Agency of Singapore*. Again, the order of the day was a large portion of forward planning, though the historical background upon which this planning is based was also thoroughly discussed.

This was the day for statistics – concerning both speakers and the contents of their presentations. These ranged through world-wide from Shane Coursen of the *WildList Organization*, to regional from *NAI's* Jimmy Kuo, and national detail from Toshiaki Kokadu of the *IT Promotion Agency of Japan*. Also fitting into this category was the comparison of Japanese and Korean virus threats presented by *Ahnlab's* founder Dr Charles Ahn. When taken as a whole, these provided a great deal of food for thought on the global and local natures of viral threat.

Despite succumbing to the 'flu, Dr Jan Hruska of *Sophos Plc* managed to complement these presentations with a description of the co-operative methods by which companies receive new virus samples for analysis. This can easily be awarded the title of most animated paper, both in the movements of the presenter and the reaction engendered in some of the audience at his promotion of REVS.

A less controversial alternative to the statistical treatment was offered in a selection of more technical papers. Dr Vesselin Bontchev of *FRISK Software* gave a brief yet comprehensive overview of viral macros, while Yoshiiro Yasuda of *NAI* concentrated on the threats posed by the use of executable compressors to create new virus variants. Mikko Hypönnen of *F-Secure Corp* rounded off events with his paper on the future of viruses for PDAs and mobile phones, a paper made more apposite by the imminent release of a Java-enabled mobile phone.

As the conference proper drew to a close the main body of *AVAR's* committee, together with a selection of the event's delegates, retired to the confusingly named Hip Hop Beer Restaurant for some rest and relaxation. Thankfully, this

Seiji Murakami didn't need to resort to violence to make AVAR2000 the most popular Asian AV conference to date.

turned out to play no hip-hop whatsoever, a boon to any of a musical bent, but was certainly very well supplied with beer. The chopstick skills of the Western delegates were put to the test and proved quite worthy, even as the drinks flowed. The low-alcohol plans of *NAI's* Vincent Gullotto were well and truly scuppered when it was discovered that it was his birthday [*I swear I never said a word! Ed.*] and as a result he was presented with a glass of beer the size of his head – which he regarded with some degree of trepidation. On a more personal note, the 'small world' phenomenon was once more discovered when I tried out my very rusty Korean. It transpired that one of the *Korean Information Security Agency* representatives had lived a few hundred yards from me when I was living in Korea.

A palsied link it might be, but the smallness of the world was, along with the governments' roles within Asia, one of the two major themes within the conference. The role of the government may generally be very different within the Asian and Euro-American spheres of operation but this hides more general similarities. There are major differences in the details but the global nature of current virus threats is certainly a uniting feature far outweighing any superficial language difficulties. A virus or worm will now often follow the clock around the globe, with preventative measures in one time-zone having an effect for the better on those whose dawn arrives later. The papers presented showed that differences in language cause differences in the specific threats which are most prevalent in a country, but have not rendered the major threats in any way impotent.

In the past there has been a tendency for Asia to be considered very much 'foreign territory'. This attitude can only be unhelpful in the fight against globally spread viruses. One of the great successes of *AVAR 2000* is that it attracted such an international spread of delegates rather than simply a wide range of Asian attendees. The next *AVAR* conference is planned for 2001 in Hong Kong and it is hoped that this international flavour is built upon then.

FEATURE SERIES

The Usual Suspects – Part 2

Andreas Marx
University of Magdeburg, Germany

We continue our look at the problems commonly encountered during the testing of anti-virus products.

Language

For server or mail server systems an English version of the AV scanner is sufficient. However, client systems require a localised version of the program, since not everyone speaks English. The program's implementation method is often to recompile itself with new language strings. This is not a good solution since even minor changes or patches require programmers to recompile everything in several different languages before testing can be carried out.

A better idea would be to provide localisation files for the scanner and make the main program independent from the language. This can be done using language definition files where all the important strings are stored, or DLLs with resource information for *Windows* platforms which are easier to handle.

Bootable Start Disks/CDs

It would be neither expensive nor labour intensive to provide customers with slightly better protection in emergency situations by making the installation CD bootable. Most programs use a small *Linux* kernel and a *Linux* version of the scanner to scan FAT, FAT32 and NTFS volumes and this is completely free of charge for the company. DOS and a CD-Rom driver will work for FAT and FAT32 disks too, but usually licence fees have to be paid. Some companies avoid this by writing their own simple DOS with additional routines to avoid problems with special malware (mostly tricky boot viruses).

Since there are still many old computers around, a bootable floppy disk should still be included in retail products. However, it does not make sense to include up to seven disks with the main scanner program running under DOS. It would be better if the bootable disk worked like the bootable CD and loaded the main program from the CD and additional or updated databases from floppy or hard disks.

Virus Naming Conventions

Very often, different products have different names for one and the same type of virus. This starts with a prefix like Macro.Word97, W97M, WM97, followed by '.', '/', '_' or whatever. Some programs use strings like O97M to show that a virus is able to infect more than one *Office* platform, others use the optional @mm or @m to show that it is a mass-mailer or a mailer. This is where it stops being relatively easy. For Win32 file viruses and worms there are more than 30 different philosophies and suffixes (Win32, Win95, W95, PE, I-Worm, TROJ etc.) and we need a standard supported by the majority of companies.

The same confusion surrounds virus names – the most widespread malware should have one and the same name under all scanners. In emergency situations different names are understandable, but never changing the name after including signatures into the database causes confusion. Another problem is the variant detection of some programs. Some say they have definitely found 12345.A, but it is actually a completely different variant since the identification checks just a few bytes. In this case, a less precise name would show that more than just one variant could be identified and that this identification is generic.

Self-Checks

Every security software product should perform a self-check to make sure it is in the original, unmodified state. However, some AV programs do not perform a self-check at all, neither on the main program, nor the scanner libraries or virus databases. Installation need not be checked, but these three essential parts ought to be.

We have seen several methods of integrity check: starting from an easy 8-byte XOR through a CRC16 or CRC32 up to a strong cryptographic checksum. The last is probably the best idea, especially if the databases contain executable code or p-code, which allows write-access even in 'scan only' mode and not just for disinfection or archive handling. In the past there have been some retro viruses which successfully caused problems with deletion or modification of all scanned files.

The check must be performed before a value of the bases is read for use in the scanner engine, since a wrong value could cause buffer overflows or crashes. If the program or the databases have been modified, an error dialog must be displayed containing all the information needed to clarify the problem and instructing the user on what to do.

This includes messages to the effect that the program must be re-installed or a scan performed after booting from a clean disk. Sometimes only short dialogs like 'ScanInit failed.' or 'Error 128. Reinstall product.' are displayed. This is not good enough. Other programs, while they do not load virus databases if they are corrupt, do not display suitable warning messages.

If the scanner seems really fast it may be because there are no viruses for it to scan for or only a very few. In one case we saw the scanner really slow down because the heuristics had to do everything. If the program is rather old, an

appropriate message should appear advising that the scanner should be updated as soon as possible and maybe how to do it. With server or mail server systems this can be done via email or in other ways.

Encryption

A good encryption of all virus-related parts in the program and its databases helps avoid false positives from other scanners and reverse engineering by virus writers. Since the size of the databases increases fast, they should be compressed, too. It is incomprehensible why some leading companies still only use '+1' or 'XOR 255' encryption of their work – some competitors' X-Ray engines are able to look inside these files! On the other hand, it makes no sense to implement strong AES (Advanced Encryption Standard) routines for protection, since the scanner has to be able to read everything. If someone really wants to decrypt the databases, they will succeed eventually.

In the main program or scanner libraries there are often unencrypted heuristic strings for macro or script virus detection or proper removal, such as modified Registry keys and how to restore them – these should be secured.

On-Access Scanners

A virus guard has to protect its user against the same viruses which the on-demand scanner finds. However, some programs do not allow the specification that all files should be scanned and that not only incoming (writes) but also outgoing (reads) files should be scanned, too.

Under high workloads it is possible to spot really big problems. In these situations, some scanners are unable to scan all files or crash intermittently and cease scanning altogether. The same thing tends to happen to the program displaying alerts and writing them to a log file: after several infections it either crashes or fails to display a full list of all the viruses found.

In some cases, it can be useful to switch off the guard for a period of time, for example while burning a CD. What we do not understand is why some scanners require a *Windows* restart for small configuration changes or after unloading. The virus guard can also help protect the scanner against modifications by retro viruses or Trojans, since it looks at access to all files. So, it is easy to implement a routine that checks if a program wants to modify or delete one of the scanner's program files and avoid it.

Archive Formats

A good scanner should be able to detect viruses in popular archive formats, and at least in ZIP files. A survey for the preparation of our last test showed that customers also want ARJ, CAB, LHA, RAR and ACE-compressed files, as well as Unix formats like TAR, GZ and BZ included in the list. Since TAR files are not compressed, some scanners randomly detect viruses inside this type of file.

Of course, the scanner should be able to scan inside the files recursively (ZIP in ZIP, but also GZ in TAR archives) in both GUI and command-line versions (DOS32 and higher). In some cases, people answering the survey requested that it should be possible to include external unpacker programs if a file format (e.g. ACE) is not supported by the scanner.

It is odd that, even if some scanners can handle archives correctly, the same programs may be unable to scan inside self-extracting (SFX) files of the same type, since the same decompression routines can be used. Other scanners only look for known SFX unpack routines, but will fail on new, changed or different language versions of them. It is essential to support most installation archives (InstallShield, Package for the Web, etc).

While scanning archives in memory is the faster and more secure solution, about half of the scanners we test are unable to do it: they extract the archives into a temporary directory and scan them afterwards. With this method, every file has to be renamed to avoid problems with specially prepared file names including pipes ('|') or other problematic characters like '`' or '". Names of subdirectories have to be ignored, since viruses like BAT/WinRip use '..' constructions to spread and copy themselves into the *Windows* Autostart directory.

Even if a scanner supports many file types, a useful standard setting – for example, 'scan only 5 recursion layers deep' – is important, since unpacking requires a lot of memory and stack space. Very large files can cause problems, too – some scanners will skip them without any notice, while others require time to scan inside them and it looks like the scanner has crashed.

Embedded OLE objects

A good anti-virus scanning engine should be able to scan embedded files inside OLE files numerous times without problems and handle them like an archive file. However, some programs still fail to find an infected .DOC in an XLS or SHS file.

In our tests, we only look at the most significant scenarios – infected COM, EXE, VBS, DOC, XLS and PPT files embedded into DOC, XLS, PPT, SHS and even RTF files. Usually, only about half of these will be found, and RTF files will not be scanned at all. But there are additional formats to these, since *Office 2000* supports the saving of all documents as HTML files, storing macros inside OLEDATA.MSO or EDITDATA.MSO files. Such files should be scanned, too, regardless of whether they include either additional embedded objects or the original document was infected.

Next month's final instalment of this series will focus on the following issues: password-protected *Office* documents, run-time compressed files, disinfection, speed, updates and test strategies.

CASE STUDY

The State of the First Union

Max M. Morris
First Union Corporation, USA

I am the Enterprise Anti-Virus Administrator for *First Union Corporation*. *First Union*, with headquarters in Charlotte, North Carolina, is the United States' sixth largest banking company and a provider of financial services to 15 million retail and corporate customers, including 2 million Internet customers. The company operates full-service banking offices in 11 East Coast states and Washington, DC., with full-service brokerage offices in 45 states and foreign countries and employs over 70,000 people.

Virus Protection at First Union

In 1999, *First Union* was hit by Melissa and suffered significant impact. A portion of our network was down for hours so that dozens of servers and hundreds of PCs could be restored and cleaned. As a result of this incident, an Enterprise Anti-Virus Task Force was created to prepare better us for future viruses. This team consists of representatives from each of the IT departments across the bank, as well as the company's Help Desk and the Information Security Division.

The Task Force's duty is to ensure that there are multiple levels of protection, from initial firewall blocking of possible infected files – either by filtering rules or anti-virus software – through anti-virus software on our email systems, to workstation-based software scanners and finally real-time protection of our file and print servers. Protection measures may also include a temporary disconnection from the Internet or the disabling of inbound email.

Virus threat communications for the Task Force are initiated by either observing a reported incident on one of the AV vendors' or industry Web sites we monitor, an email from one of the AV vendors that we have signed up for alerts, or a direct call from our primary vendor's support team. If it is determined that the virus poses a low risk, an email description is sent out to the Anti-Virus Task Force. Depending on the publicity the virus/worm is receiving, the email may also be sent out to our Help Desk, other support areas and management. If the virus is medium-high risk, as much information as possible is confirmed, a bridge-line opened and the Task Force contacted immediately.

On that bridge-line, the first action is to implement firewall filtering to provide an immediate block of emails containing a new virus. If filtering is not an option, a decision is made around whether or not to shut down inbound/outbound/both email traffic to minimize the risk. All areas are required to put filtering rules in place immediately (or shut down systems and provide confirmation of those rules).

If a virus outbreak warrants it, a corporate broadcast is placed on the central Web sites which internal customers can access for information around new virus/worm threats. All the different areas of the company are then responsible for contacting their own individual anti-virus vendors to determine when a new special definition pattern file will be available for their area.

An impact summary is available for all areas within several days of the outbreak – this details the number of Help Desk calls, actual workstations infected, number of files infected/lost, etc. The Task Force then summarizes the incident, looking for improvements and/or potential changes (specifically those which would help reduce the amount of time required to immobilize the group or to obtain filtering or pattern files in place faster).

In addition, we have provided education for our customers with flyers, emails and broadcasts. It appears that the best method is simply to report recent outbreaks of real viruses/worms and the publicity that they have generated from sources our customers see every day (Web sites, TV news, etc). Even that sometimes does not work – we had people still opening LoveLetter-infected virus attachments two weeks after the initial outbreak.

The best defence does not require the customer to do anything at all. In other words, the responsibility is on our response team and support areas to help ensure that we are doing everything possible to protect our customers against new outbreaks by having solutions and processes in place that do not require their intervention.

Avoiding Significant Damage

We have standardized our AV solutions on our servers and desktops as much as possible, selecting products that provide centralized management and implementing solutions that require no intervention from our customers. In addition, we look at the following to help ensure our damage is minimal.

We ensure that the malware outbreak is a real one and that it will truly impact our company. Just because an AV vendor calls it a 'major outbreak' doesn't always mean it is. We factor in how the malware is spreading (type of email system), the geographic region it is being reported in, the actual number of infections (and how many new incidents are being reported) and the type of industry reporting the infection (compared to our environment).

We have varying risk factors we use to let our company know how critical a new threat is. If we send out email on every new reported virus, customers – and support areas – become (pardon the pun) immune to the communications. So, we try to weigh all the factors before making a decision

and communicate it appropriately.

We ensure that lines of communication, including procedures and service level agreements for our AV vendor and our internal response team are developed, in place and understood before an outbreak. It is important to have an up-to-date list of all contacts, including management, easily accessible. A method for contacting everyone easily (i.e. pager software with a canned message that requires only slight modification and one send) is critical.

We do a mock outbreak test on a regular basis to work out any problems with the process and have regularly scheduled meetings of our response team to talk about what we can improve on and do differently. We also ensure that the data being provided by an AV vendor is accurate (specifically, ensuring that email filtering rules are correct).

The first thing we do is put email filtering in place, even if we have anti-virus software on our email systems or firewalls. No matter how good our anti-virus vendors are, it takes time to get a new patterns tested and pushed out. When a major new virus or worm has been reported and confirmed and we know we are at risk, minutes, not hours, count and the quickest way to ensure something does not get into our environment is to block it. If there is not a specific text string or file attachment which can be used, temporarily shutting down email systems may be the only other way to keep infection and propagation from occurring. However, we always weigh the downside of the impact of blocking mail within the company.

Current and Future Initiatives

A recent initiative included changing the anti-virus solution we use in our corporate environment. We recently selected a popular Corporate Edition as the new standard for our *Novell* file servers and those desktops logging into them, replacing *Intel's* retired *LANDesk VirusProtect*.

The new solution provides us with more robust protection against future viruses and offers new features, including higher scalability at the enterprise level. It has also helped to reduce significantly the likelihood of 'fragments', an occurrence commonly seen within *First Union*.

Currently, the protection of company data is accomplished by undedicated resources in various corporate departments and company subsidiaries. To help us prepare for the ever-increasing virus threat, we have submitted a proposal for the creation of an Enterprise Anti-Virus Administration Team. This Team would allow *First Union* to protect its data and computers effectively against current and future virus attacks with minimal impact. It would also ensure that we have the most efficient methodology in place for anti-virus management.

Some of the major benefits of this team include the ability to govern existing corporate anti-virus protection more efficiently by improving on the processes that are already in place. This means being able to view anti-virus protection for the *First Union* enterprise as a whole, allowing us to map out our current environment from a network topology perspective to determine where there is inadequate protection and where there are possibly holes in our anti-virus defence.

It also allows us to investigate enterprise-wide product standardization. Going to a single product as much as possible provides more consistent virus detection, a common interface for our customers, a single standard for our Help Desks to support, simplified vendor relationships and product consolidations. This means substantial savings through reduced software and maintenance costs.

Getting the Message Across

Viruses and worms, while not new from the perspective of how long they have been in existence, have only recently become a real threat to large corporate environments such as *First Union*. With the advent of the Internet, corporate intranets, global email communications and new types of malicious virus and worms which spread quickly, we are just now beginning to realize the true threat of viruses for our company and the industry as a whole. With that in mind, not only do we expect to see an increase in the number and frequency of new viruses in the near future, we also anticipate seeing new methods used by virus writers to infect computers.

One of the more recent examples of this to receive significant media attention was BubbleBoy, which did not actually require you to open an infected file attached to an email. Simply reading the email itself triggered the infection. In addition, new platforms are now being targeted, such as the recent new virus that infected *Palm Pilot* devices.

In reality, anti-virus protection has become a game between the virus writers, anti-virus vendors and a company's anti-virus administrative capabilities. And it has become a game where even minutes can make the difference between a non-event and a major virus outbreak that would cripple a company and cause substantial impact and financial loss.

The key to winning this on-going battle is to ensure that all aspects of your anti-virus administration are covered and fine-tuned so that there is no chance for error and your reactions are immediate.

OPINION 1

Compiled Trojans – the New Threat, and a Hope

A Padgett Peterson PE, CISSP
Lockheed Martin Corporation, USA

The Melissa virus brought in a new class of concern for corporate America. This was not because it was a virus (in that respect it was just like every other boring *Word* macro virus) but because it introduced the concept of mass-mailing, a new form of denial of service.

Except that this virus was not really new, in fact it is remarkably similar to one of the first problematic viruses, the CHRISTMA.EXE that struck *IBM* in 1987. Back then, when the user executed the email attachment, it displayed a cute little Christmas tree. In the background, it read the user's address book and sent copies of itself to every address listed. Sounds familiar?

In Melissa's Wake

Melissa was just the first (and probably unintentional) of the mass mailers to be directed to the fertile *Microsoft Office* environment. Here it made exceptional use of the then-new 'Collaborative Data Objects' or 'CDO' – a new feature of the *Office* suite.

To be fair, this concept of leveraging attributes as far as possible makes good marketing sense. Since email capability is an essential these days, it was simple to package this capability into the operating system, first as a standalone application which persisted through *Outlook 97*, then as a bundled application which included *Internet Explorer*. Both required the same features so it made sense to utilize a common engine – now an integral part of *Office 2000*.

However, while the use of a common engine for all applications makes sense, it would also seem prudent to keep track of which application was allowed to use which feature set. It appears that this was too difficult – it was much easier simply to use a common application language, Visual Basic, and provide appropriate extensions for all applications to use.

Unfortunately, few users ever figured out how to use the features or even that they existed. Virus writers, however, were quick to realize their potential. Well, not quite. Certain anti-virus researchers were experimenting with these 'features' in 1997 and advised *Microsoft* in early 1998 that some of them might be too powerful for safe dissemination, particularly without an easy way to turn them off.

Just as in 1995/6 when anti-virus researchers pleaded with *Microsoft* to provide a way to disable macros, at least

macros in documents, it was apparently just not the proper business attitude and for almost a year, not much happened. The threat actually surfaced in mid-1995 (with the Concept virus), was recognized by researchers later that year, but did not become a pervasive problem for another six months.

Then in March 1999, the Melissa virus appeared and took the world by surprise using the same construct –CreateObject – that researchers had warned about over a year earlier. In 2000 this escalated yet again with VBS/LoveLetter or the IloveYou virus, which discovered how to do the same thing using a VBScript mechanism to duplicate what had previously required *Word* (or *Excel*). True, *Microsoft* had finally provided a way to detect and turn off macros in documents, but the underlying capability was still there and the virus writers simply kept finding new ways to exploit them.

The Next Stage

November 2000 brought the next logical step – compiled Visual Basic programs. Just as *Windows Scripting Host* had been slipstreamed first into *Outlook 98* and later *Windows 98* through the SCRRUN.DLL, in September 1998 MSVBM60.DLL (the run-time library component of Visual Basic 6.0) was added to \WINDOWS\SYSTEM, in time to be included in *Windows 98 SE*. This DLL included run-time library routines like MAPI, the same constructs which permit the email aspects of CDO. Once again, a program could establish native *Exchange* email connections directly.

So, CreateObject(Outlook.Application) was still available but now it could be compiled. Instead of a signature scanner having to look for a simple text string, it had to examine a compiled binary stream – and that is considerably different.

W32/ProLin was the first example seen in late November 2000, but by the first week of December source code had been published on the Internet for another mass-mailer. Today is the 6 December and all of the ramifications have not yet been seen but now there is a simple means of

creating a mass mailer and large corporate enterprises which are susceptible to server clogging attacks have a new reason to be concerned.

Now, I doubt that this will be a viable means of attack for long, at least for those systems with effective defences in place. Some already block executables from entering but that is a draconian prohibition and one that is not effective for telecommuters. Removing executable attachments at the mail servers is another, possibly more effective, method.

The problem with this approach is that most executables are identified by program extension – the three letters following the period. Once upon a time all people needed to be concerned with were three self-explanatory extensions that were recognized as executable: .COM, .BAT, and .EXE. In recent years this has been added to greatly and executables may be found in .SHA, .HTA, .OCX and many other extensions that you may never have heard of. Exacerbating the problem is the fact that most gateways permit a list of extensions to block, not a list of those to which to allow access. This is an inherent problem that hopefully will soon be corrected.

Still others are adopting 'sting' techniques – dummy mailboxes scattered in the address books that in reality will trigger an alert message and log the connecting machine off. This is not something a virus could anticipate, but it does require a certain amount of expertise to install.

The third avenue of detection is yet to come: effective signature scanners which can detect new attacks. This tends to be difficult since the virus writers are the first to get the latest anti-virus products and there are techniques available to determine what they trigger on. An intelligent and evasive virus is something the medical world does not need to be concerned with yet.

However, there may be a home team advantage here. In assembly language, macro code, and scripting languages it is fairly easy to write self-modifying code that can use encryption for concealment. Compiled code OTOH rarely lends itself to self-modification and the same optimizations which resolve variables before converting to tokens may provide common binary constricts for simple detection of suspicious code. In other words, it may be easy to detect CreatObject(Outlook.Application) in binary. It may even be easier to detect than in VBScript.

New Approaches

It is also possible that this will give rise to something I have been expecting for years – the Integrity Managers. By that I mean programs which can detect not viruses, Trojans, or worms, but rather things that should not be happening. The idea is to determine that *something* is happening and then bring out signature tools and other mechanisms to determine *what*. The fact that so many previously scanner-only products are now including heuristic detection is an indicator that this is changing.

Some personal firewalls are already capable of detecting connection attempts being made by programs that have not been granted permission to access the network. Such programs may well detect viral action before it can develop. It is true that many people already have personal firewalls installed, but these must be configured properly in order to be effective and, unfortunately, not many people take the time to do so.

Moreover, it will require a program that can discern which programs are allowed connections and not merely what protocols are allowed. This means the personal firewall program must not only work at levels 3–4 of the seven-layer IP stack but must extend to levels 5–6 to determine which program is making the request and not merely which request is being made. Some do, some do not. This is not a question the public is asking today.

The other potential point of control is for disk access. This is really the same application just directed differently. Unfortunately, many more programs are allowed to write to the disks than are allowed to write to the disks so this is somewhat more difficult. Today such network and disk management tools are separate products. In the near future they may be combined.

Conclusions

These are all things that can be done today. Individual experts are already setting machines up to protect themselves – the problem is scale. The span of one individual is probably under 100 machines. For an enterprise of 150,000 users, that would require 1,500 SMEs (Subject Matter Experts). These people simply do not exist.

The only viable answer today appears to be automation and single points of control, something many enterprises are wrestling with. How to control an enterprise securely and with trust at a time when the watchword is 'fast/cheap/easy'? Not an easy question to answer.

The difference is that in the post-Melissa world, for the first time there are metrics – measures of attacks with costs involved. Already some corporate organisations are taking steps to avoid those costs in the future, but they are few and far between and there is too much in the way of 'smoke and mirrors' to tell for certain what will be effective and what is just fluff. The real problem is that anyone who can actually see through the smoke probably does not need any of the products in the first place.

This is not a simple world we live in, particularly when it is constantly evolving in unexpected ways. No wonder the best crisis managers have low blood pressure, few others could survive. So what we have here is a new threat, one that could have been predicted but which is, as usual, a surprise, and one that hopefully will be resolved even before this is in print. Then again, we know there will be another Melissa or LoveLetter, it is just a matter of when and to whom.

OPINION 2

The Lawful Truth

Eddy Willems
Data Alert International, Belgium

How a law works depends on what country you are in, although you should be aware that if you commit a crime in a country other than your own, authorities there may be able to extradite you to face prosecution (for example, this happened to Dr Popp over the Aids Information Diskette Trojan). Sometimes I hear people saying that virus writing does not need to be made illegal. If it is not illegal to write viruses, the law should concentrate on the damage caused by the virus, but this is not always easy.

Maybe a virus writer should not be held responsible – unless his virus appears somewhere where it is not wanted. But if it does, then its creator must be prosecuted (if known, of course) – even if he is not directly responsible for spreading the virus.

Naturally, the person who spreads a virus intentionally is even more guilty and should be prosecuted more severely, but the original author should be held responsible too, for letting his creation escape. I overheard someone saying that the proper legal term for this kind of occurrence in Belgium is 'criminal negligence'.

I have been working in the anti-virus business for ten years but it seems that I was one of the first in Belgium to complain about the unbelievably old laws we are still subject to over here. I have enquired of and complained to the Federal Police on a regular basis, but they have not been able to do anything about the virus exchange boards and sites because, until recently, there has been no relevant legislation. This has made me angry sometimes but a change is on the way.

New Legislation: An Improvement?

A few years ago the only way to deal with a hacker or virus writer in Belgium was to prosecute them for 'misuse of electricity'. This was a really old law dating back practically to the Napoleonic ages which was still active. After years and years of long meetings where I and a lot of others asked for new, improved legislation, it happened that some fairly recent viruses and fast-spreading worms shook the Belgium Government itself really hard. At last, this resulted in completely new legislation concerning so-called cyber crime being implemented early this year .

Let us see what came out of the brilliant minds of our Belgian Government. Individuals who intentionally break into a network will now face a 625 Euro fine and/or risk incarceration of anything from three months to two years. The same punishment will be meted out even if you try to break into an area of your employer's network to which you do not have access. It is exactly the same if the break-in was started and did not work out completely as foreseen. So, even an attempted break-in will be punished.

If a hacker actually causes damage or if he used the hacked system to gain illegal entry, then that individual will face a fine of 1250 Euros and three years in a state prison. The person who helps him with hacking tools like password-stealing devices and so on will be fined 2,500 Euros and awarded a spell in prison of between one and three years.

Furthermore, if someone hacks or writes a virus 'by order of…' (i.e. on behalf of or at the behest of someone else) then that accessory will be held responsible too, and faces up to five years in prison and/or a 5,000 Euro fine. Virus authors themselves risk a fine of 2,500 Euros and/or three years in gaol.

Individuals who unleash viruses, worms or Trojans (or other malicious code) onto a system, whether or not it is intentional, will also be threatened with up to three years in prison. If such a virus causes damage such as file deletion, the writer concerned could be locked up for up to five years and have to pay a 2,500 Euro fine. If someone re-attempts the same thing, then the punishment is doubled.

I think this is an improvement. Up until now we have had no formal, legal redress with which to deal with all our virus writers and cyber terrorists. I hope with these laws in place that we will experience a decrease in virus and security problems.

However, it could also turn out that laws forbidding virus writing and malware distribution will not deter virus authors, and in some cases could even spur them on. Despite calls from the anti-virus industry and users for tougher legislation covering the writing and distribution of viruses, this may not be the answer and could even do more harm than good.

Police intervention does not always offer a significant deterrent. Even the well-publicised conviction of virus writers such as David Smith (author of the notorious

Melissa virus) failed to have any impact on the number of new viruses appearing throughout the world. Maybe we could educate people or even children that virus writing is not a 'good' thing in order to prevent a new generation of virus writers developing. But how do you do that?

The e-Security Team: Another Good Move?

After the VBS/LoveLetter outbreak, the Belgian Government wanted to do something special for the Belgian people. Following a day of brainstorming, one minister came up with the idea of putting up a Web site where everyone could find alerts of security issues such as viruses. The goal was to alert the Belgian people before anything else was published on the Internet and before an outbreak could begin. An attractive goal and an interesting new approach – that was the idea, anyway.

In order to run this Web site and its associated alarm system, the minister's cabinet duly decided to set up a security team which consisted of relevant experts. Thus, on 5 May 2000, what is known as the e-Security Team of *BIPT* (*Belgian Institute of Postal services and Telecommunications*) was established.

Despite the fact that Belgium does not boast many anti-virus or IT Security experts, after just two days this team was formed. It included individuals from ISPs, some TV stations, several large corporations and two security companies. So, where are the real experts? *Data Alert International*, the company I work for, volunteered me as the only anti-virus expert within the whole team. The other security company donated a general security expert. I really have my doubts about the people they gathered from the other companies. I also have my doubts about the system being used to alert Belgian citizens. This is how it works.

When one member of the e-Security team hears about a new virus or security breach, he must alert the *BIPT*-system (let us call it *BIPT* for simplicity's sake). This alert can be communicated by email, phone or fax. After that, *BIPT* must attempt to reach everyone concerned by email and mobile phone.

This is done in the following way: an email, with an attachment, is dispatched to everyone on the team. The attachment is a *Word* document in .DOC format, containing details about the facts of the problematic new virus, worm, Trojan or hoax. At that moment, an SMS message is also sent to everyone on the list to ask them to have a look at their email. The document attached can then be used to send additional and more precise information about the virus to *BIPT*. Can you imagine the feeling I had when I saw the first email they sent to me? If someone in the chain has inadequate protection they could send a virus round within the email itself! Immediately, I asked that some other method be used to send this information around.

So, what happens next? After receiving this information our job is to respond within one hour. If this is done properly an article can be published on the Web page within that time, and a national security alert can be prepared – nice, if it works. However, the details of the first message I saw were rather disappointing; it would appear that some members of the team cannot even tell the difference between a hoax and a virus. Neither can they distinguish if something is of low or high importance. In one case I was just in time to prevent a warning for a non-existent virus!

The alerts often seem to be based on personal feelings or thoughts, and are often superfluous. They also change depending on the person who is responsible for editing the site at that specific moment. Indeed, there seems to be a sort of shift system. The articles change too – a few months ago everything was translated. Now, they have stopped doing this and instead, they just add some links to various anti-virus developer Web sites. Would it not be more helpful to point to all AV Web sites and only select genuine alerts? But, how do you define a real alert?

At those times when I am out of contact with the team, members tend to gather information about the 'problematic' virus direct from an anti-virus developer's Web site. It is then consolidated and put on the official team Web site even if it has not been verified. I am afraid that I have lost sight of the original goal. Furthermore, I would say that at such times as that described above the system certainly does not act as an early warning.

Making it Work

Now, do not misunderstand me. The original concept of an early warning system has some potential. I have already explained the problems I see to the team and improvements are being made. And the system has worked in the past! We managed to get an early warning out about the 'Big Brother' hoax even before it showed up on the anti-virus developer Web sites. This, in my opinion, is the best example that shows the system is functioning efficiently.

It occurs to me that the *BIPT* e-Security Team is actually more of a political game than it appears, but nevertheless the idea behind the project should be honoured, because it is a responsive and a helpful one. The question remains, however, could a new virus or worm with all the latest and perhaps even unknown spreading techniques be quicker than this system?

Despite my busy schedule doing consultancy work, giving presentations, analysing viruses and seeing to *EICAR* matters, it seems to me that I have always got several channels open to help the team when it is really needed. I sincerely hope that I will be on time at that moment! You can take a look at the *BIPT* Web site at www.bipt.be, where you will find all the warnings posted in Dutch and French, our national languages. Oops! An SMS message from *BIPT* just came through on my mobile phone as I was finishing this article. It seems that I am working more on behalf of the Belgian people than for *Data Alert International* at this moment!

PRODUCT REVIEW

IKARUS virus utilities millennium edition v.5

Matt Ham

It is becoming a standard opener to standalone reviews to state that the company producing the software is a relatively long-term player in its own country and is seeking to widen its market abroad. This is yet again true of *IKARUS Software*, the Austrian company whose offering is reviewed this month. The software in question is offered in three languages, English, German and Korean, and is part of a suite of security programs. These programs are mostly made up of virus detection software for various platforms, with the addition of a content scanner and security utilities. Although DOS, *NetWare*, SMTP and *Windows 9x* are supported by *IKARUS'* scanners, the *NT* version was singled out for attention in these tests.

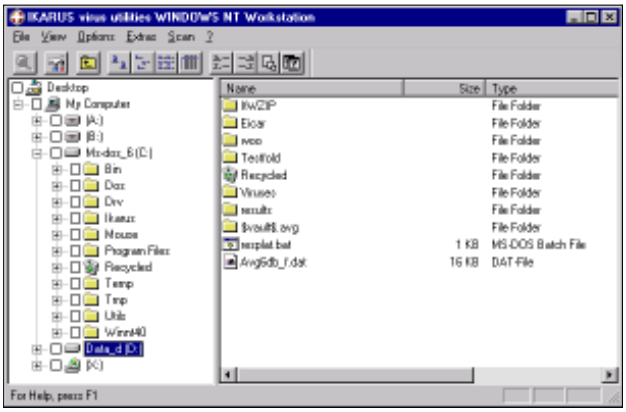
Testing was performed on *Windows NT 4* workstations with Service Pack 5 applied. Scans were performed locally. The timed tests were on the same machine used in the last Comparative, thus where figures are mentioned for scan rates these are directly comparable with those given in that test (see *VB*, November 2000, p.16 and December p.11).

The choice of 'Ikarus' as a company name seems odd to the classics cognoscenti at *VB*, for he is remembered as being not a particularly bright chap. Provided with wings by his rather more illustrious father Daedalus, Ikarus soared off into the skies. Unfortunately, his confidence exceeded his ability to perform, resulting in a disastrous crash, his last sight a great blue expanse.

The fate he suffered does bear certain resemblances to some days at the *VB* test labs, though it is hoped that this was not what the Austrian developers had in mind when they named their software company.

Such flights of fancy were obviously not repeated when naming the product, the two components being named *virus utilities* and *Guard NT*. The former is used by *IKARUS* interchangeably to refer to the on-demand scanner or whole product, while the latter is reserved for references to the on-access component In this review, however, the names are used to refer to the components unless otherwise noted.

There is, however, some sign of the same naming mind at work in the details of the product, as the older scanning engine was named T.H.O.R. and the newer T2. This leads in both cases to images of mass carnage and destruction by blundering dimwits and psychopathic computers respectively. Perhaps this is reading too much into the names – but there was some trepidation experienced as the tests were started.



Boxed Contents

Since this version was very much hot off the presses when the review was started, the box did not appear until some time after the product itself. The documentation in English was not vast and epic in its proportions and showed definite signs of having been translated rather than being an original work. Despite this, it was suitably informative and clarified the workings of otherwise obscure functions in a number of cases.

Installation

Installation of *virus utilities* starts with a bright splash screen, which prominently includes the date of the program, useful for checking that an up-to-date version is being installed. At this point, the product as a whole is named *virus utilities millennium edition* which was confusing at first, as it seemed a possible reference to the *Windows* version of the same designation.

Since the installation program is not the often encountered Installshield, the process is slightly at variance with the usual process. The licence agreement defaults to 'yes', and is not particularly extreme in its stipulations.

Next comes the choice of install location, defaulting to a folder in the root of the hard drive rather than one in the more usual Program Files section. This is also the point where a Standard or User defined installation is selected. Also available, but not active on the *NT* workstation, were the Administrator and Floppy setup choices.

The choice of Standard or User defined setup is one which is less complex than with some other products. The Standard install consists of the *Guard NT* and *virus utilities* components plus the two sets of help files associated with these. The User defined setup simply allowed each of these four components to be installed independently of the others. This does allow for some odd combinations – installing the help files with no related scanners is entirely possible though slightly pointless to all but an amused reviewer.

There follows a choice of whether to make the programs available through the start menu or the desktop and also whether to apply these installation features to the current user or to all users. There is no option for installing to both desktop and startbar and once made, these choices appeared to be unalterable through re-installation. After this selection is approved, setup is complete as far as the installation program is concerned.

There are two further areas where configuration can occur, which might possibly have been better placed in the installation program itself. Upon first running the *virus utilities* program there is a choice of language – English, Korean or German as mentioned above. This does not, however, alter the language within the *Guard NT* program, which must be entered and reconfigured from its original German state for English use, Korean not being available here. Since the default is German this leads to having the need for some understanding of that language. This is probably not too difficult for the guesswork of most English speakers, though German can hardly be said to be widely spoken in Korea as yet.

The help files installed can be accessed only from within the two programs and were only present in German, which made them somewhat useless from the point of view of an English speaker. The programs were thus, generally, operated by a system of experimentation. The translation of labels and dialog boxes was of a high standard, however, with a few typos and missed dialogs being the only barriers to comprehension. Given this, the lack of any English help in this version was not a mighty hindrance, though *VB* reviewers might be expected to have an advantage over most users on the 'what does this do?' front.

Features

The *virus utilities* program, or on-demand scanner, is advertised as having an *Explorer*-like interface, which cannot be argued with. A drop-down menu bar above an icon bar containing selected menu options is at the top of the interface, while the rest of the screen is split. On the left the directory tree is displayed, while on the right the contents of selected folders or areas can be seen. This gives a generally bare look to the interface when it is at rest – with various dialog boxes being popped up in addition to the basic interface when activity occurs.

The menu bar has options for File, Options, Extras, Scan and '?' – the last of which is the symbol for the source of help and information rather than some unknown, unprintable hieroglyph. As is customary, these will be treated in turn. File is, unexpectedly, quite a small menu, containing simply the entries for 'Show log file' and 'Exit'. Both of these are fairly self-explanatory, though the log file structure will be returned to later.

View begins in true *NT Explorer* style with the Tool bar and Status bar toggles, to which is added the Split command. This is quite mystifying, since it allows the lower part of the screen to be split in desired proportions rather than the default setting. Not in itself an odd command, this is performed much more simply by a click-hold-release on the divider, requiring fewer clicks and less thought than entering the View menu. The theme then returns to the *Explorer* style with the Large icons, Small icons, List and Details settings. These are not quite *Windows* standard, however, lacking the blob which designates which method of viewing files is currently selected, though this setting is arguably redundant to any user able to look at the right hand side of the screen.

Also within View, the Arrange icons menu does not come with its customary Line up Icons pairing but does add an option to arrange icons by attributes. This last feature is just downright strange, since sorting is done on the basis of no readily fathomable algorithm. The View menu is completed by the 'Refresh' and 'Move one level up' commands plus the 'Show network drives' toggle.

The options drop-down menu is the next along the bar,and more approaching the heart of *virus utilities'* functionality. Here the settings offered are 'Configure', 'Load default settings', 'Global configuration file' and 'Language'. The latter gives the option to change the base language after start up, 'Load default settings' is obvious in its utility, while 'Global configuration file' enables the current configuration to be saved or others to be loaded. It is the Configure option which leads to the very core of *virus utilities*. Choosing this produces a tabbed dialog box from which there are many configuration settings to be tweaked or selected at will.

Starting with 'General', this configures whether memory, boot sectors or master boot sectors are examined, whether results are displayed automatically and whether the user is prompted before actions are taken. The configuration settings may also be password-protected here, though the security of the scheme was not put to the test.

Secondly, 'Log' gives control over whether a log file is written, the name of the log file and whether the method is to append or recreate the log when additions are made to it. There is also an option to include extended information in the log files. This consists primarily of including scanned but clean objects in the log file. The standard information included within the log is indeed a model of completeness, containing details of almost all the configuration options which are of major relevance. It is, however, surely irrelevant to include in the log the option 'write to log file' as having been selected.

'View' has two options, 'Save View' and 'Show' icons, which allow the setting of a standard startup view for the on-screen folder and the tweaking of the righthand pane display, respectively. The Archive tab proved more interesting. By default this setting is off, and when activated two new selections become available. The first is the location of any extra temporary directory where decompression work in progress may be stored during scanning. The more

innovative is the option to supply a list of standard passwords for archive files, so that the scanner can access these files easily. The option does, however, leave these in plain view of anyone authorised to change scan options within *virus utilities*.

'Actions' is another quite familiar set of options, relating to how viruses are treated when found. The default is no action, with disinfect, delete, move and rename all supported. Disinfection may optionally be preceded by file backup and the location of quarantine and rename choice may also be selected here. 'Exclusions' is devoid of any surprises. Files or directories may be selected, though since browsing is not selectable here there could be a higher than necessary latitude for errors when using this feature.

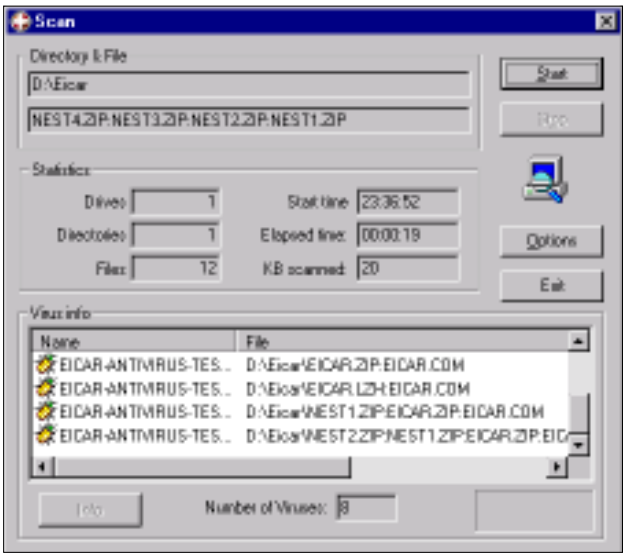
There are also some interesting features present in the 'Extras' section, where global configurations can be enforced. In addition to this, it is possible to set a special notebook mode to mitigate problems with variably non-existent floppy drives on portable computers. Warnings for out of date Virus DataBases may also be activated or deactivated as required.

The last tab is perhaps the most controversial in its settings. This is the 'Extensions' settings tab and is noteworthy more for what is not contained. The non-controversial parts allow the selection of 'normal' or 'special' mode for scanning, the latter being more rigorous, while heuristics may also be changed from their default activated state.

The list of scanned extensions, however, is rather short and missing extensionless files, any form of *PowerPoint* associated extensions and a selection of others which have become notable of late. New extensions can be added, and wildcards are supported, though the addition of extensionless files is always tricky in such circumstances. The effects of this lack of scanned extensions were to be apparent in the scanning test results.

Moving back to the drop-down menus, the 'Extras' menu is next on the list. This contains useful utilities for the recovery of system areas and CMOS – though the latter could not be persuaded to become active. The system areas selection enables areas to be saved, restored or compared with saved versions, though this was untested. More interesting was the utility for placing boot sectors of a selected sort on to media, though this is certainly an option which should fall under the password-protection scheme.

This is coupled with an MBR repairer, which has an integral MBR finder, useful for those seeking MBRs shifted by boot infectors. These are nice to have for the likes of *Virus Bulletin* readers but the use of most of the tools will be limited to a very small group of people indeed. With all these tools the *NT* version is, moreover, likely to be unavailable if system or boot information is damaged or absent. Encryption of drives, folders or files is also supported and again this was neglected in favour of the more interesting matter of virus scanning.



The final drop-down menus are devoted to Scans and Help, the former allowing scans to be triggered and the virus encyclopaedia inspected. The encyclopaedia is marred by an excessive initialisation time and contains a majority of minor variant names which refer the user back to the original virus of that name. Finally, the Help menu offers version information, the input area for user details and the link to the help files where they are present.

After all of this, how does it relate to scanning ? When the scan areas are selected in the trees in the lower panes and the scan triggered, a summary box appears which also gives a last chance to change scan options.

The scan may then be started fully, with a mini summary popping up when finished and the full summary available through a log or the already existing summary box. This does seem a duplication of information, having two summaries on screen at the same time.

As for the *Guard NT* on-access scanner, this is configured from the toolbar by a right-click. By default, heuristics are off for this type of scan, and 'Programs and Documents' the default setting for files to scan, this being an even more limited selection than the standard on-demand extension list. The control over scanned extensions here is, on the other hand, much more refined, with each of the programs and documents selectable independently – all files are available, as is the option for user-defined lists.

Although heuristics are off here they can be activated and the settings of the default Normal scan may be upped to Special. Exclusions too are better handled, with size limits possible and processes selectable.

Scanning here is termed 'supervision' and is applicable independently to Reads and Writes, with this being fine-tunable for a good variety of devices. With messages and user rights also capable of being altered with some degree of accuracy this program showed a degree more refinement than the *virus utilities*.

Scanning Tests

In standalone testing, the detailed results of scanning tests are traditionally less important than the feature set – pure scan results being the preserve of the *VB* Comparative Reviews. Nevertheless, scan testing did produce some interesting results. The tests themselves were not prone to instability but were not devoid of problems.

Towards the end of scans virtual memory warnings appeared, and after viewing logs in the same session these memory problems manifested themselves in unpredictable failures of programs such as *Notepad* and *Explorer*. It also resulted in the cessation of scanning during one of the most strenuous scans and thus scans were performed with a reboot between them out of a sense of paranoia. It is to be hoped that this is a result of the large size of the virus test-sets and the number of infected files contained within them. The false positives tests did not show this tendency to memory problems.

Testing was performed with standard settings, with the additional individual selection of 'special scan', 'no heuristics', and 'all files' respectively. As a 'paranoid' case, testing was also performed with 'special scan' and 'all files' activated together.

The results of the default scan showed the gaps that were expected due to the choice of extensions scanned. This resulted in misses on W95/Babylonia (.HLP), JS/Kak (.HTA), O97M/Tristate.C (extensionless and .P?T), W95/MTX.B (.PIF) and others – though several ItW set viruses with standard extensions were also missed.

Overall misses were similar in their pattern, though *Excel* viruses were a distinct weakness in detection. And several of the polymorphic viruses were not detected at all. Perhaps more surprising was the number of detections which were accounted for by the heuristic scanner. Of the total test-set of roughly 22,000 viruses, 19,000 files were detected as infected on a scan with heuristics enabled.

With heuristics disabled the same scan resulted in one third of that number being detected. This vast change was not as striking as raw figures might suggest – many of the new misses could be attributed to the polymorphic set, where hundreds of samples of each virus exist. There were also, however, many heuristics-only detections within the standard set – with macro viruses far less likely to be detected by heuristics only.

There were also some oddities in the log files for some scans, with files being marked 'could not be examined' – a result, presumably, of the lack of virtual memory for which warnings were issued. This resulted in several scans' results being unusable since detection in this case could not be confirmed or denied from logs. This is potentially a loophole in detection capability on lower end machines. [*Worried developers should note that new test machines will be in place for the* Windows ME *Comparative in the February issue. Ed.*]

The 'special' scan improved detection slightly, though with a severe increase in time taken to scan the virus sets, which was already long. On-access testing using the *Guard NT* application proved slightly irritating, since the on-access monitor alert was impervious to being disabled. After a test on standard settings was completed, another was started with the 'paranoid' settings of 'all files', 'special scan' and 'heuristics on' ('off' by default). These scans gave very similar results to those seen in the on-demand scans – not really a surprise since the scanning engine runs as a service and is presumably accessed by both the on-access and on-demand components.

Overhead testing was not without its oddities. False positives were seen both on-demand and on-access, though in the latter the 'detection' was erratic. Scanning speeds for the product were in general slow at 30 minutes for the standard Clean set executable test and suffered large increases with the addition of more paranoid scanning, rising to 510 minutes.

On-access overheads were in default mode around 100% rising to 200% with the 'paranoid' setup and no false positives. On one occasion when a false positive was triggered this overhead rose to the awesome peak of 4600%, a new record in *VB* testing. Despite the added detection given by the higher settings it seems likely that few users would wish to incur the time problems.

Conclusions

Long term readers of *VB* will be familiar with the picture painted by this review of a product which can definitely detect but which has distinct holes in its capabilities. The configuration reporting and interface are generally easy to use and informative, the main problems being in choice of extensions, speed of scanning and the misses in the ItW and *Excel* areas. Many AV products reviewed in these pages have started off in a similar vein and gone on to achieve VB 100% awards – what will be interesting now is the rate at which the problems are addressed by the *IKARUS* team.

Technical Details

Product: virus utilities millennium edition v.5.

Developer: *IKARUS Software GesmbH*, 1060 Wien, Fillgradergasse 7, Austria; Tel: +43 1589 95-0, fax +43 1589 95-100, email office@ikarus.at, http://www.ikarus-software.com/.

Price: Single user version – £31/US$45, 10 users – £199/US$293, 100 users – £1417/US$2091.

Test Environment: Workstations: Three 166 MHz Pentium-MMX workstations with 64 MB RAM, 4 GB hard disks, CD-ROM and 3.5-inch floppy, all running *Windows NT 4 with SP5* . The workstations were rebuilt from image back-ups, and the test-sets were scanned on local hard drives or CD-ROM.
Virus Test-sets: Complete listings of the test-sets used are at http://www.virusbtn.com/Comparatives/WinNT/200011/test_sets.html. A complete description of the results calculation protocol is at http://www.virusbtn.com/Comparatives/Win95/199801/protocol.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:

VB, 50 Sth Audubon Road, Wakefield, MA 01880, USA

Tel (781) 2139066, Fax (781) 2139067

END NOTES AND NEWS

The UK Security Show 2001, incorporating The IT Security Showcase, is to take place in Hall 2 of the Wembley Arena in London, UK from 14–15 February 2001. The line-up includes interactive product demonstrations and practical installer workshops alongside study-based seminars and debates and more traditional conference-style presentations. For more details about the event visit the Web site http://www.securityshow.com/.

Symantec* announces the release of *Norton AntiVirus Corporate Edition v7.5, a key component of *Symantec's Enterprise Security*. The new release has closed loop automation technology, part of the Digital Immune System. The company also announces the launch of a new range of products for Macintosh users. For more information on either of these product releases, including price and availability visit the Web site http://www.symantec.com/.

The 10th Annual *EICAR* conference, also the 2nd European Anti-Malware conference, is to be held in Munich, Germany from 3–6 March 2001. **For more information see http://www.eicar.org/.**

InfoSec 2001, Europe's largest IT security event, is to take place from 24–26 April 2001 in the National Hall, Olympia, London, UK. See the Web site http://www.infosec.co.uk, or find out more about the event by emailing infosecurity@reedexpo.co.uk.

***TenFour* announces the release of *TFS Secure Messaging-Server v4.61*.** A new feature in the product opens and checks the contents of file attachments. For more information contact UK PR Manager Liz Stewart; Tel +44 1620 810989 or visit http://wwwtenfour.com/.

iSEC Asia 2001 is to be held at the Singapore International Convention and Exhibition Centre from 25–27 April 2001. The conference and exhibition covers IT security topics from anti-virus through encryption to biometrics and digital signatures. For more information and a booking form contact Stella Tan; Tel +65 322 2756 or email stella@aic-asia.com.

***Norman Data Defense Systems* announces the release of *NetBank Edition*.** The new product is an Internet security solution designed exclusively for on-line banking services. Participating Bank customers are to receive protection from virus and Trojan attacks free of charge. For more information see http://www.norman.com/.

InfoSec Paris 2001, the 15th information systems and communications security exhibition and conference, will take place at CNIT, Paris-La Défense, France from 29–31 May 2001. Companies wishing to participate in the exhibition are encouraged to contact the organisers; Tel +33 0144 537220, or email salons@mci-salons.fr.

California-based Internet mail server provider *Stalker Software* has announced a strategic partnership with *McAfee*. Stalker is to embed *McAfee's VirusScan* into its *Stalker CommuniGate Pro* product. For more information see the Web site http://www.nai.com/.

AntiVirus eXpert for MS Office 2000 is now available from *Central Command Inc*. Email ssundermeier@avx.com for more details.

iSEC Australia will take place in Halls 5 & 6 of the Sydney Convention & Exhibition Centre from 6–8 August 2001. For information on how you can be a sponsor, exhibitor or delegate, visit the Web site http://www.isecworldwide.com/isec_aus2001/. Alternatively contact Chris Rodrigues; Tel +61 2 9210 5756.

***Computer Associates Inc* has released a beta download of its flagship anti-virus product *InoculateIT v6.0 for Windows*.** For more information, see the Web site http://www.ca.com/products.betas/.

Following queries from users and developers, *Virus Bulletin* is clarifying its VB 100% award scheme. The full protocol and criteria for this scheme are at http://www.virusbtn.com/100/whatis.html. As far as AV vendors are concerned, little has changed since the inception of VB 100% awards back in January 1998, with the exception of one important new development. As of now, on-access scanners will be tested on 'close' as well as 'open'. As has been the case for some time, the flagging of false positives precludes the winning of an award and all products are tested in default mode, meaning that the detection settings are in their 'out-of-the-box' state throughout testing.

For the benefit of users, VB 100% awards continue to be platform-specific and clearly dated. Promotional material featuring VB 100% awards without dates should be reported to *Virus Bulletin*. As usual, *Virus Bulletin* is here to offer subscribers the best impartial advice about anti-virus security and the products on offer. Please contact us with any problems and queries either relating to VB 100% awards or to general AV problems; Tel +44 1235 555139.