ISSN 0956-9979 FEBRUARY 2000

VIRUS BULLETIN

THE INTERNATIONAL PUBLICATION ON COMPUTER VIRUS PREVENTION, RECOGNITION AND REMOVAL

IN THIS ISSUE:

• **Happy Birthday *Virus Bulletin*!** *VB's* annual conference is ten years old this year. Information about where, what, how and who can be found on p.3.

• **How DOS your scanner rate?** Thirteen products line up in this month's Comparative – starting on p.16 – competing for the first VB 100% awards of the new millennium.

• **New and Improved?** Two recently discovered *Windows* file viruses are examined to reveal ever more complex infection methods. Our analyses start on p.6.

CONTENTS

COMMENT

“ *We survived the information apocalypse …*”

Media: Insight or Incitement?

[*Sarah Gordon was to reflect this month on the impact of David Smith’s guilty plea. However, since sentencing has not yet been pronounced, that commentary has been delayed temporarily. Ed.*]

Millennium madness has come and gone – we have all been exposed to the tidal wave of press coverage. We survived the information apocalypse, but not without some battle scars. The press releases by *Computer Associates* and the rebuttals by *Sophos* (and others) prompt me to reflect on the impact of the media and the information provided by anti-virus vendors and researchers.

The Melissa outbreak seemed to cause an awakening within the on-line community: viruses were big news and could affect anyone, from the average home PC user to the high-tech corporation. As a practitioner supporting organizations in their anti-virus protection strategies and response, I discovered that my customers didn’t get virus information from the anti-virus vendor sites, but rather from the media – *CNN*, *MSNBC* and the like. We rely heavily upon the media to provide us with information that helps us make daily decisions, down to movie times and weather forecasts. I expect that information, when presented as fact, to be as accurate as possible. And I, like many IT practitioners, am also reliant upon the anti-virus vendors and researchers to provide information which helps me make decisions that affect the protection of the organizations I support.

When inaccurate, incomplete, or sensationalized information is provided to the public, whether it is about viruses or the weather, it obviously has an impact. In the case of information relating to computer viruses, the impact has more than just credibility issues for the press or vendors. It results in lost productivity and vast amounts of overtime for the weary customer who doesn’t want to experience ‘another Melissa’ and wants the latest protection, often even before it is available.

Recent virus misinformation situations caused an extreme influx of support calls, necessitated threat/protection briefings to senior management, and required the potentially unnecessary mobilization of deployment teams – preventing people from performing their normal tasks. An example? I heard about BubbleBoy during the very early morning news; so did approximately 90% of our organization’s senior management and customers. By 8am, senior management (and numerous others) had already called the helpdesk repeatedly to establish whether or not they were protected. They were (understandably) very eager for a solution and ready to roll it out as soon as possible – believing their computing environment was in imminent peril. You might think this was an excellent response, and in many ways it was, at least from an awareness perspective. However, it significantly delayed us from getting the information we needed from our vendors and from being able to supply useful information to the network and system administrators (who, incidentally, were *already* on standby) to install the necessary updates.

It does not even require a *real* virus to generate this sort of near panic – reactions to hoax messages are similar. People rush about to warn their friends, co-workers, and seemingly most of the world about the latest virus hoax, ‘get rich quick’ chain letter, or urban legend without actually reading it or checking out its validity – opting for the send key instead. Then we, the support team, spend our time combating a false alarm or misinformation that takes us from our normal duties.

Calculating the cost of virus (and hoax) incidents is difficult. It is, however, safe to say that the cost of an incident rises considerably in the face of inaccurate and/or incomplete information. I realize that marketing departments and the media have tremendous jobs that are sometimes a contradiction – the need to sell a product/story versus the need to present timely, accurate information that may help thousands of people. Since awareness is a key component in the solution to the virus problem, dissemination of information needs to be handled responsibly. When inaccurate, incomplete information is repeatedly presented as fact, people eventually stop listening – making the cure sometimes worse than the disease.

Christine Orshesky, i-secure Corporation, USA

NEWS

Sunshine Conference

VB 2000 is to take place on Thursday 28 and Friday 29 September 2000 in Florida. The conference and concurrent exhibition will be held at the Hyatt Regency Grand Cypress Hotel in Orlando. The traditional Welcome Drinks reception is planned for the evening of Wednesday 27 September, and the Gala Dinner for Thursday 28 September.

Virus Bulletin is currently seeking submissions for papers to be included in the conference programme. Abstracts of approximately 200 words must reach *VB* by Tuesday 29 February. Regrettably, submissions received after this date will not be considered. Please send your abstracts (in ASCII or RTF format) to editorial@virusbtn.com.

There are currently both sponsorship and exhibition opportunities available. Last year's conference exhibition proved enduringly popular, and corporate feedback was extremely positive. For information about sponsorship and exhibition packages or more details about VB 2000, please contact VB2000@virusbtn.com ▮

Nothing if not Predictable

The first and only virus to replicate under *Windows 2000* was discovered in early January. Apparently, the 29A group is responsible for W2K/Installer (also known as W2K/Inta), two variants of which have already been released. There are no reports of this virus in the wild. Experts are not attributing much significance to this virus, despite a unique infection method. Watch this space for further details in the near future ▮

Feeding the Hand that Bites?

A Taiwanese firm which manufactures multilingual *Linux* operating systems has hired the author of the notorious CIH virus to test its hardware. *Wahoo International Enterprises Co* proudly claims to have beaten off several rivals in the attempt to employ 24 year-old Chen Ing-hau. Whether *Wahoo* has been progressive or precipitate remains to be seen. Sentiment and second chances aside, *VB* takes exception to the description of 'a rare computer professional' bestowed on Chen by a company spokeswoman ▮

CD-ing is Believing

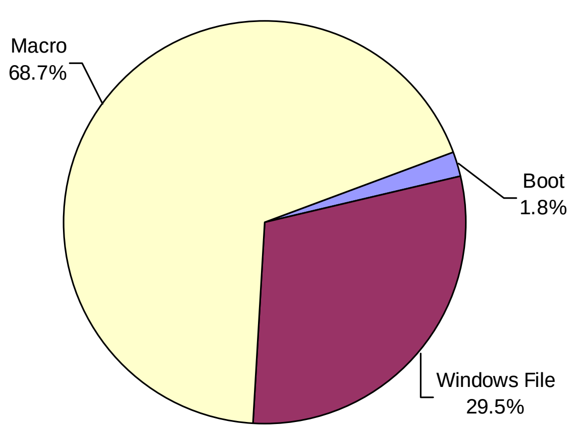
Virus Bulletin is pleased to provide current subscribers with a complimentary copy of its 10 year CD, containing back issues of the magazine since its inception in 1989 up to and including December 1999. Non-subscribers can purchase CDs for £95 or US$150 and there are corporate bulk discounts for regular participants in *VB* Comparative Reviews. Contact *VB* for more information ▮

Prevalence Table – December 1999

Virus	Type	Incidents	Reports
ColdApe	Macro	272	31.0%
Win32/Pretty	File	106	12.1%
Marker	Macro	84	9.6%
Win32/Ska	File	77	8.8%
Laroux	Macro	58	6.6%
Win32/ExploreZip	File	47	5.4%
Class	Macro	35	4.0%
Melissa	Macro	33	3.8%
Ethan	Macro	27	3.1%
Win95/CIH	File	21	2.4%
Cap	Macro	17	1.9%
Thus	Macro	16	1.8%
Tristate	Macro	11	1.3%
Freelinks	Script	10	1.1%
Story	Macro	10	1.1%
Broken	Macro	6	0.7%
Form	Boot	6	0.7%
Form	Boot	5	0.6%
Win32/Fix	File	5	0.6%
Astia	Macro	4	0.5%
Chack	Macro	4	0.5%
Verlor	Macro	4	0.5%
Locale	Macro	3	0.3%
Others [1]		17	1.94%
Total		868	100%

[1] The Prevalence Table includes a total of 17 reports across 13 further viruses. Readers are reminded that a complete listing is posted at http://www.virusbtn.com/Prevalence/.

Distribution of virus types in reports



LETTERS

Dear Virus Bulletin

MMX-cuse Me

With reference to Snorre Fagerland's 'Merry MMXmas!' article in December's issue (p.10), I have a comment on the line – 'The return from this opcode (bit 17) tells the virus whether the CPU supports MMX or not'.

Now, the fact is that CPUID returns stuff in EDX when it is executed with EAX=1. Bit 23 is the actual MMX support bit according to *Intel*. This means the 17h (hex) bit.

However, it is not because of the typo that I am writing this. The polymorphic engine of this virus mutates the CPUID stuff. One possibility is that it checks the 17h bit to see if MMX is presented – this time correctly.

However, it uses TEST EDX,00400000h which happens to be the 22-bit not the 23-bit. The 22-bit is usually 0 and the virus will not detect the MMX in that case. The correct way would be TEST EDX,00800000h.

Thus, the virus will believe that the code does not run on MMX in some cases. Even then, when it should not include any MMX stuff the code might contain MMX instructions in all directions! This was indeed the case in a sample I have investigated.

The article gives the impression that the virus can be faked by not returning the MMX bit in EDX when CPUID EAX,1 is issued. However, in my sample, the executed instruction causes the virus to fail on a non-MMX processor because an MMX instruction gets executed in the non-MMX chain of the code too. I think the virus does not always generate non-MMX and MMX chains, but MMX/MMX chains by accident. It could be a different bug in the virus.

Péter Ször
SARC
USA

Counter Offensive

Daniel Schrader's military man (see *VB*, December 1999, p.4) would not just want a distant perimeter, he would also want a complete perimeter. A fence does not do much good if the attacker can walk round the end.

Similarly, if 80–90% of incidents originate as files attached to email, 10–20% arrive by other means. It is pointless concentrating your anti-virus efforts at the ISP if this results in (a smaller number) of uncontrollable incidents because there is no desktop protection in place. AV at the ISP is an incomplete perimeter defence. It does not prevent viruses arriving on CD-ROM, CD-R, diskette or other physical media. While these methods are slow, once it has arrived a Worm can spread quickly from desktop to desktop via the network (e.g. ExploreZip). Protection at email servers, gateways or file servers will not stop such spread.

AV at the ISP cannot even prevent all viruses arriving from the Internet – end-to-end email encryption and VPNs prevent that. If your ISP can crack your encryption in order to search all traffic for viruses, perhaps you should be looking for a new encryption vendor.

We need a perimeter without holes, the only place we can build that (for all the currently significant viruses, i.e. ones that run on the desktop) is at the desktop, because that is the only place we can be sure of intercepting all incoming routes. Protection at the server, gateway or ISP is a useful supplement to this, and they do deserve more attention, but they are not a replacement for desktop protection.

To address the flaws Mr Schrader finds with the desktop model, namely, users turning it off or not updating frequently enough – administrators should look for products that give them, not the users, control. This includes features like central management and updating, protection that only administrators themselves can uninstall or turn off, and testing for protection and disconnecting unprotected desktops from the network.

As Mr Schrader makes military comparisons, and I'm writing from China, I will finish with a quote from Sun Tzu, 'Know your enemy and know yourself'. This must include knowing all the methods your enemy may use, and the holes in your defence.

Allan Dyer
Yui Kee Co Ltd
Hong Kong, China

What's the Outlook?

In Eric Chien's 'Malware Do You Want To Go Today?' article (see *VB*, January 2000, p.18) he writes 'Microsoft Outlook already has VBScript support with VBA to be added in the near future'. Up to *Outlook 98* did indeed have only VBScript support, but *Outlook 2000* introduced full VBA capabilities. Normally, *Office* applications let you associate VBA code with their documents. As there is no such thing as an 'Outlook document', the code is stored in a single file that serves as global storage (NORMAL.DOT).

The file is called VBAPROJECT.OTM and stored in the appropriate place …\Application Data\Microsoft\Outlook in *Windows NT* and *Windows 95/98* if the user profiles are defined and in …\Application Data\Microsoft\Outlook in *Windows 95/98* if they are not. The new *Outlook* object model supports application level events such as, for example, Application_Startup (occurring when *Outlook* is started) or Application_Newmail (occurring when new mail is received).

This makes *Outlook 2000* a potentially vulnerable target for VBA macro viruses.

Gabor Szappanos
Computer and Automation Research Institute
Hungary

Apocalypse When?

So, Y2K has come and gone without the 'onslaught' of computer viruses some companies were predicting. Did anti-virus companies learn anything from the Michelangelo mania in 1992? It seems not.

For months a small minority of anti-virus experts were telling the press that there was simply no evidence that viruses would be any more of a problem on 1 January 2000 than on any other day of the year, against a herd of AV marketroids who were hyping up the threat as equivalent to a virus 'superbowl'.

Even the much vaunted Dutch virus-writing challenge turned out to be a damp squib, with only one and a half viruses actually released.

My wish for 2000? That anti-virus companies will act more responsibly, present the true facts of the threat to their users rather than damaging the credibility of the entire industry with this kind of hyperbole.

Depressingly, I suspect we will see more of the same. Customers, engage your codswallop detectors!

Graham Cluley
Sophos Plc
UK

Crying Wolf!

[*While Vincent Gullotto makes a valid point,* VB *often picks up the pieces following less than helpful press releases from the majority of the big AV companies. Furthermore, since we received this letter many of them have added a proportion of the viruses listed below to their sites. Ed.*]

What do the following threats all have in common?

Feliz
Inst98
VBS/Lucky
VBS/Tune.A
W32/Crypto
W95/Esmeralda
W95/Lovesong
W95/Plage.worm
W97M/Armigidon
W97M/Backhand.A
W97M/Chantal.B
W97M/Marker.BN
Win2K/Inta
Win32.NewApt.Worm.d
Win95/LoveSong.998
Win95/Spaces.163
WScript/Kak.worm
Zelu

They are *all* low risk threats that *Computer Associates* has, for some reason, chosen to 'warn' users about. In addition, all these 'warnings' have taken place over a two-week period! Let's be serious here for a moment. Now, I admit that *NAI-McAfee* has a rather large, moving marketing machine. However, in all the years of pushing information, *NAI* hasn't even come close to manifesting such a barrage of unnecessary warnings.

So the next question is – why? Well, perhaps they needed to back up a rather bold initial press release, which can be found at http://www.cai.com/press/1999/12/y2k_virus.htm. In this release *CA* noted that it 'helped curtail the incidence of major virus outbreaks'.

How could that be, when most of these were never even seen at a customer's site? To add to this, *AVERT* gave five of these threats to *CA* as a trusted part of the AV Research community's practice and *CA* issued warnings on these as though that company discovered them. We didn't see the necessity to 'warn' the user community, and nor did any other vendor. We simply posted them to our Web sites, and proceeded to give an assessment.

My only guess in attempting to answer the previous question would be that *CA* has decided that it, and no one else in the community, is the all-knowing and all-seeing AV company. As doers of good, the unsuspecting AV Administrator in a company will view *CA* as 'the authoritative word' in anti-virus protection.

Distributing a press release or 'warning' has many significant impacts. The primary of these should be when there *really* is a threat and users *do* need to be concerned. In most, if not all, cases this should be when there are actual customer sightings, or something is discovered that makes a significant change occur.

Warnings like *CA's* misleadingly create the impression that a company has the user's best interests at heart. Messages like these 'warnings' generate thousands of calls to Help Desks around the world. They cost companies thousands of dollars while the threat lies as dormant as a cat on a Saturday night while the mice are at a cheese-feast.

Enterprise-wide companies do not need to be belittled by an organization which makes a decision to warn users of something that it is doubtful some researchers will ever see, let alone a computer user.

Vincent Gullotto
AVERT – NAI Labs
USA

VIRUS ANALYSIS 1

Digital Rivers of Babylonia

Marius van Oers
NAI, The Netherlands

The W95/Babylonia virus was discovered at the beginning of December 1999 (see *VB*, January 2000, p.3). It consists of multiple components and is basically a combination of a *Windows 9x* PE and *Windows* Help file virus, and an email/IRC Worm. This virus' most interesting feature is its ability to change the actual viral code/payload by checking remote 'template' plug-ins posted on a Web site (more details on this later).

Initially, the virus was encountered in a *Windows* Help file called SERIALZ.HLP (40.637 bytes) on an Internet newsgroup called Alt.Crackers. This trap file is supposed to contain registration information for licensed software products but all too soon, the hidden 'functionality' appears at the user systems. Towards the end of the Help file, the virus contains Aplib compressed binary viral code. Upon launching the infected *Windows* Help file a script routine is called to decompress and execute the viral code. The virus installs itself in memory, creating a 32-bit 'PE' file called C:\BABYLONIA.EXE, which is then executed.

When the BABYLONIA.EXE file is run, it copies itself to C:\WINDOWS\SYSTEM\KERNEL32.EXE and modifies the *Windows* Registry so that it is called on each system startup. Note that the file's name is KERNEL32.EXE, not KERNEL32.DLL like the regular *Windows* file. The trick of loading via the Registry and using the slightly modified names of regular *Windows* files is often seen in relation to remote hacking/BackDoor programs. The main reason for BABYLONIA.EXE/KERNEL32.EXE is to get the infection routines on the local system initialized.

Plug-ins

Once running in memory, KERNEL32.EXE is not visible in the *Windows* task list, as it is a registered system service. The process checks at selected time intervals (every minute, approximately) for an Internet connection by monitoring the dial-up networking status. If there is an active Internet connection, it tries to connect to a specific Web site (210.169. 20.21), which turns out to be in Japan. So what is the aim of this Web site check?

The first component to be downloaded is an ASCII file called /VECNA/VIRUS.TXT consisting of these four entries: DROPPER.DAT, POLL.DAT, IRCWORM.DAT, and GREETZ.DAT. These files have a specific marker at the beginning. The file header starts with VMOD, probably meaning Virus MODule – a quick identifier similar to *Word's* DOCF. All four components are downloaded with the 'Get' instruction and launched sequentially.

There are two main motives behind this trick. First of all, the virus can change its viral/payload code by using a 'remote template' to do 'real-time virus/payload updating'. Secondly, if the virus part (EXE/HLP) is cleaned but the 'BackDoor' part of BABYLONIA.EXE/KERNEL32.EXE is still running, then it could download the viral code from the Web site and start to re-infect the local system. This is why DROPPER.DAT is used; it can install/re-install the viral part by using a temporary file called INSTALAR.EXE (Brazilian/Portuguese for INSTALL.EXE). After being run it gets deleted.

Briefly, (more details in the 'Email Worm' section) this file INSTALAR.EXE is the same (except for the name/icons etc) as the files that are used as attachments in the email Worm component, with files named X-MAS.EXE etc.

The IRCWORM.DAT file is meant to get the virus spread by IRC (Internet Relay Chat). Firstly, the file checks for MIRC.INI. The script file will try to send a file called 2KBUG-MIRCFIX.EXE to other on-line users at various IRC channels:

```
[script]
n0=run $mircdir2kBug-MircFix.EXE
n1=ON 1:JOIN:#:{ /if ( $nick == $me ) { halt}
n2= /dcc send $nick $mircdir2kbugfix.ini
n3= /dcc send $nick $mircdir2kBug-MircFix.EXE
n4=}
```

Luckily, in this case, this routine does not work very well, but IRC should always be set to prompt the recipient whenever another IRC user wants to send over a file with, for example, 'DCC send'. IRC users should constantly be aware that it is not always a genuine picture file or a harmless program being sent!

The POLL.DAT file was probably designed by the virus writer to monitor its 'popularity'. It gets the computer name and will send an email to babylonia_counter@hotmail.com. Here it will appear as mail from Babylonia@rasta.net with the message 'Quando o mestre chegara?' ('When will the master arrive?'). The GREETZ.DAT file is used to 'say hello' to the unsuspecting user. It uses the GetLocalTime function and will modify the AUTOEXEC.BAT on 15 January, inserting the following text:

```
W95/Babylonia by Vecna (c) 1999
Greetz to RoadKil and VirusBuster
Big thankz to sok4ever webmaster
Abracos pra galera brazuca!!!
–
Eu boto fogo na Babilonia!
```

The Brazilian/Portuguese section translates as 'Hugs to the Brazilian guys' and 'I put fire in Babylonia'. Incidentally, the VECNA.HTM file on the Web site indicates the Brazilian origin of the virus:

```
<META NAME=”Generator” CONTENT=”Microsoft
Word 97">
<META NAME=”Template” CONTENT=”C:\ARQUIVOS DE
PROGRAMAS\MICROSOFT  OFFICE\OFFICE\html.dot”>
```

The virus writer most likely comes from Brazil and abused a commercial Japanese Web site provider. After some tracking the provider was identified and apparently took appropriate action, so the hosting threat was stopped early. Luckily, no real payload, such as file deletion, triggered but the ability to change virus/payload code by using an Internet ‘template’ is scary. Security software is able to block IP addresses so this, once again, indicates the shift from desktop solutions towards integrated solutions towards to firewalls etc. The Japanese Web site not only hosted the plug-ins for Babylonia, but appeared to be a virus/info exchange site. Entries were found from 29A, VicodinES and NoMercy.

EXE/HLP Virus

The memory resident virus infects 32-bit Portable Executable (PE) EXE files as well as *Windows* Help files (HLP). The infection routine works in *Windows 9x*, not in *NT*, as the virus makes calls to *Windows 9x* virtual device drivers (VXD). The virus reserves VXD memory and becomes a system driver. By performing a trick with the Interrupt Description Table it is able to move its code into the low-level area of system drivers (Ring0, another proven technique, used by the ‘ringzero’ Trojans).

The infection routine will run upon file access using File Open, File Rename or when setting file attributes. The PE file’s entry point is not modified, probably to fool some AV scanners’ heuristic code analysers which would get suspicious if the initial control flow were set to the end of the file. Instead, the virus inserts a call to its viral code. The viral code is usually located towards the end of target files.

In most cases the viral code will be put into the last section of the PE EXE files and infected files will grow in size. That is not always the case. PE EXE files are nicely structured/aligned and sometimes have enough empty space for the virus to store itself with no file size increase. Constant file size increase is actually a remnant from the 8-bit DOS COM/EXE days – usually, file size increases in PE EXE files are variable. Babylonia-infected PE EXE files see a file size increase of up to 10 KB, with minimum size for clean target files of 8 KB.

Not all infected PE EXE files run properly, so crashing PE EXE files could easily arouse suspicion in the user. Infected Help files get a small script routine and the viral code added. The routine uses the *Windows* Help system to launch the data as a binary *Windows 9x* program.

Email Worm

If WSOCK32.DLL is not in use the virus will try to patch it with a short routine, targeted at the application-programming interface to catch the send email trigger routine. Each time the user sends an email, an attachment is added. The name of the attachment depends on the month – in December it is called X-MAS.EXE, a 17,020-byte PE file.

So far, only the X-MAS.EXE file has been encountered and not the other names. This is most likely due to a small bug in the viral code. Anyway, what’s in a name?

The attachment is sent not as an uncompressed PE EXE file, but in mime format, the protocol used by many Internet mail clients. This attachment-adding is reminiscent of the email Worm W32/Ska (Happy99). Babylonia will infect the already attached files if they are PE EXE or HLP files. If the email attachment is executed by the recipient, two dialog boxes appear. The first one shows a caption ‘Loader Error’ and a message ‘API not found!’. The second displays (under *Windows 9x*) the caption ‘Loader Error’ and the message ‘Windows NT required! This program will be terminated.’. The third has (under *NT*) the caption ‘Loader Error’ and the message ‘Windows 95 required! This program will be terminated.’. The C:\BABYLONIA file is created and run, followed by a copy which is inserted into C:\WINDOWS\SYSTEM\KERNEL32.EXE and so on.

Closing Thoughts

Babylonia is a pretty complex virus/Worm ‘cocktail’. The components it uses enable it to spread on local systems, attach itself to email messages, send itself by IRC and change/re-install the virus if the ‘BackDoor’ component is still active. It is a complex mix of separate existing techniques and it does not rely on all its components in order to be successful – they can run independently. Although Babylonia’s fixed Web site was taken down at an early stage, the virus could still propagate by email and IRC. Furthermore, what if the Web site components were harder to track down (IP masking) or not limited to one? The appearance of Babylonia indicates again the shift from the single target desktop environment to integrated packages. AV software should cover the full gamut from desktop towards gateway scanners/firewalls etc.

W95/Babylonia

Alias:	None known.
Type:	Memory resident *Windows 9x* PE EXE and *Windows* HLP file infector, email/IRC Worm with plug-in components from Web site.
Infects:	PE EXE and Windows HLP files.
Removal:	Use current AV software to identify and clean infected PE EXE files, restore infected HLP files and WSOCK32.DLL from backup. Remove Registry entry.

VIRUS ANALYSIS 2

Cryptomaniac

Adrian Marinescu
GeCAD, Romania

Windows viruses have evolved rapidly over the last year. We've seen many new techniques adapted from DOS viruses, but also many new methods.

Out of the large number of viruses received in the last year, Win32/Crypto is one of the most complex and remarkable Win32 viruses I have ever seen, raising many questions about its disinfection. The virus is written by a guy with the nickname Prizzy, from the infamous 29A group, the same author who wrote the first polymorphic engine able to generate MMX instructions inside decryptors.

Many of you know the One_Half virus, and what is so unusual about it. One_Half makes an infected system depend on the virus code in order to run properly. Using the very same idea, a brand new virus – Crypto – makes a Win32 system 'need' the virus code in order to work properly after infection. It does this by encrypting the DLL files from your disk and then decrypting them on the fly when they are needed. To make the cleaning process much harder, the virus uses strong cryptographic algorithms (provided by the Crypt API included in *Windows*) to encrypt files.

Running Infected files

When executing an infected file, the virus code receives control from the polymorphic decryptor loop. After setting an exception handler, Crypto tries to import an impressive number of APIs, 82 in all, from the KERNEL32.DLL, ADVAPI32.DLL and USER32.DLL libraries. To make infected files less suspicious, names are imported using checksums calculated on the API names.

At this point, Crypto tries to fool heuristic analysers by splitting the infection process into two different threads. The first is responsible for infecting the KERNEL32.DLL file, and the second will erase the virus code from memory. Thus, Crypto will prevent anti-virus programs that can scan the process memory from detecting the infected programs running in the system. The first thread will disable several resident anti-virus programs: *Avast*, *AVP*, *AVG* and *Amon*.

Then, it will check if the current program is executed under a debugger by calling the IsDebuggerPresent API. Since the virus does not check if the API address is a valid one, it will not work under *Windows 95*, where this API is not implemented. Also, Crypto checks if the *SoftIce* debugger is present. If found, the virus will simply set the stack pointer to zero, causing the fault handler to be called. In such a case, the virus code will jump to the original program code.

In order to be able to use the *Windows* Crypt API the virus needs to create a new key, with the container name set to 'Prizzy/29A'. First, the virus checks for its existence, and if the key is not present the dedicated API is called in order to create a new one. Then, the virus will need to store the generated key using the system Registry.

At this point the virus contains a limitation – it assumes that the …\Microsoft\Cryptography\UserKeys\Prizzy/29A key will be created after the CryptAquireContext API call. The virus will set the value 'Kiss Of Death' to the newly generated key.

To be able to spread, Crypto will modify the exports of KERNEL32.DLL in order to filter the following APIs: CreateFile, OpenFile, __lopen, CopyFile, MoveFile, MoveFileEx, LoadLibrary, LoadLibraryEx and FreeLibrary in both ANSI and UNICODE forms.

Crypto will copy the KERNEL32.DLL file to the *Windows* folder, infect it and then force *Windows* to swap the old file with the infected one at the next boot. To make sure that the creation of Registry keys and infection of the kernel are not carried out more than once, before calling those routines the virus will create a mutex called 'Crypto:Mutex'. If the mutex already exists, the virus will skip the two tasks mentioned before. Now, the second thread will overwrite the virus code with nops, cleaning the infection signs from the process image. After all this is set, the virus calls the original program code.

After the infected kernel is loaded, the virus filters several file-related APIs. Then, the virus body receives control. Unlike other KERNEL32 infectors (such as Win32/Kriz), Crypto will leave the virus image unencrypted in the KERNEL32.DLL file.

When the virus body receives control it will search for victims on all the disks from drives C to Z. All archive files from all the targets will be infected. To search for files, the virus will create a thread and use a synchronization mechanism based on mutexes. In order to make the scanning process less suspicious, Crypto will wait for three seconds before each drive scan. On each *Windows* reboot the virus will try to infect 20 executable files.

Infection of EXE files

Files smaller than 4,096 bytes are not infected. To verify if the file is already infected, Crypto uses a very unusual method – calling a routine that uses FPU operations such as 'exp' and 'ln'. If that routine returns the carry flag clear, the files are assumed to be already infected. In fact, the whole routine checks if the number is a multiple of 117. The virus will enlarge the last section and reserve space for its code, then write the encrypted code along with the polymorphic decryptor in there and set the program entry point to the decryption routine. To be fully Win32-compatible, Crypto uses the dedicated Win API to compute the new file checksum after infection.

Since the virus uses the GetTickCount API, it will not generate many different forms in a short period of time. If the infected file is in the KERNEL32.DLL library, the virus will also hook several APIs. The virus is aware of many *Windows 2000* features, but does not check to see if the file is protected by the System File Check mechanism.

Files with names starting with the following patterns are not infected: TB, F-, AW, AV, NAV, PAV, RAV, NVC, FPR, DSS, IBM, INOC, ANTI, SCN, VSAF, VSWP, PANDA, DRWEB, FSAV, SPIDER, ADINF, SONIQUE, SQSTART (most of the anti-virus utilities are avoided this way).

Archive Infection

Crypto is able to add droppers with the following names: ‘install’, ‘setup’, ‘run’, ‘sound’, ‘config’, ‘help’, ‘gratis’, ‘crack’, ‘update’, ‘readme’ beginning or/and ending with a ‘!’. The extension is .EXE. The method used to insert the droppers inside an archive is new – the virus creates a dropper and executes the external program needed to process the respective archive type. Using this technique, the virus is able to append the dropper (compressed) with a randomly selected method, depending on the archiver program. Affected archive types are ACE/RAR (also SFX files) and ZIP/CAB/ARJ.

Retro Functions

Crypto contains many routines that are meant to disable the anti-virus programs running on an infected computer. First of all, it disables several memory resident protections, as mentioned before. When scanning the disks, the virus also looks for: CHKLIST.CPS, AVP.CRC, IVP.NTZ, ANTI-VIR.DAT, SMARTCHK.CPS, SMARTCHK.MS, AGUARD.DAT, CHKLIST.MS and AVGQT.DAT. Also, if the LGUARD.VPS file is encountered, the virus will patch its internal structures, avoiding detection by the anti-virus product using that database.

Filtering the LoadLibrary

If the cryptographic keys are created in the installation part of the virus, Crypto is able to encrypt the code of the DLL files loaded with the LoadLibrary API and decrypt them on the fly. Any DLL with the name starting with one of the following patterns are excepted: SFC, MPR, OLE32, NTDLL, GDI32, RPCRT4, USER32, RSASIG, SHELL32, CRYPT32, RSABASE, PSTOREC, KERNEL32, ADVAPI32, RUNDLL32, SFCFILES.

Also, DLLs listed in: …\SessionManager\KnownDLLs and …\SessionManager\Known16DLLs are excepted. The most important aspect is that the encryption key and the encryption algorithm are unique for each infected system. WinCrypt supports custom encryption algorithms making disinfection from systems other than *Windows* impossible. The encryption of the DLLs will consume many time/CPU resources – the virus will read the keys needed from the Registry each time.

Polymorphism

The polymorphic engine of Crypto displays several old techniques from the DOS viruses successfully applied in *Windows* infectors. First, the polymorphic decryption loop does not hold the encryption key. Instead, the virus uses a brute-force attack to compute the decryption key, in the same way the DOS virus IDEA does. This is not the first time this has been done under Win32 – the first was Win95/IHSix.3048, discovered several weeks ago.

Secondly, the non-linear encryption of the virus body makes x-raying an obsolete technique for detection. Considering the vast number of instructions used by decryptors and the very large number of executed instructions per decryptor, with an average of about 50 million instructions, we can say that this is the most advanced polymorphic engine written for Win32.

Epilogue

Even if it was not considered vital in the past, disinfection is, and will increasingly become, one of the most important parts of an anti-virus product. Viruses that spread very rapidly inside a company need to be handled with a fast and radical solution. Virus authors try to make this process harder, in order for their viruses to gain enough time to spread. Due to several limitations and infection bugs, Win32/Crypto is unlikely to be found in the wild in the next few months, but its ideas could be used with great success in other viruses.

Win32/Crypto.21458

Aliases: None known.

Type: Memory resident, polymorphic Win32 infector.

Infection: PE files, inserting droppers to ACE ARJ/CAB/RAR/ZIP archives.

Self-recognition in Memory:
Since the KERNEL32.DLL file cannot be infected more than once, the virus does not need to check its memory residence.

Self-recognition in Files:
Last DWORD in file is a multiple of 117.

Removal: Boot from a clean floppy disk, delete infected files and restore from backups. Locate all affected DLL files and restore from backups.

FEATURE SERIES 1

Malware Do You Want To Go Today? Part 2

Eric Chien
Symantec, The Netherlands

In last month's first instalment I discussed both the current and the upcoming threats in terms of malware. In this second and final part I present a brief overview of a few of the key technologies that are in development for combating these threats, keeping anti-virus vendors and customers ahead of virus writers.

Automation Systems

The whole practice of taking days to react to new virus threats is being thrown out of the window and for good reason. In order to prevent fast spreading infectors such as network-aware malware, two things must occur. Reaction times need to be quicker than spreading times, and systems need protection *before* seeing the actual threat (more on this later). The realization of the former is already here today and can be seen in the developments made by several anti-virus vendors.

At the 1999 *VB* conference Steve White and colleagues demonstrated the *IBM*/*Symantec* Digital Immune System. During the presentation, John Morar mimicked an average user who attempts to access a document infected with W97M.Melissa.A (without suitable definitions).

The document was identified by heuristics and automatically forwarded to the *Symantec AntiVirus Research Center*. There, automation tracked the submission, replicated the sample, created a signature, built the signature into definition files, performed quality assurance, and sent back the fix to John's computer, which repaired the file and then allowed him to access it.

This entire procedure took place in 42 minutes without any human intervention, and all along the way the Administrator could easily see at which stage the submission was engaged. More importantly, this was a live demonstration – it really happened!

The idea behind such automation is that anti-virus vendors can spread the fix for a virus utilizing the same means by which the virus spreads (via the Internet), but even faster. Someone in England can submit a virus to the *Research Center* and an hour later have a fix. Moreover, not only will *they* have the fix, someone in California will have it, as well as everyone else, all over the world. Network-aware malware may be able to spread quicker than 'classic' infectors, but with this automation we can spread our solution quicker than ever before.

Heuristics and Emulation

Heuristics seems to have a bad name. It is associated in many people's minds with false positives and the inability to repair. When Administrators hear about heuristics they often roll their eyes stating the idea sounds great, but how come the heuristic missed the Foobar virus? Granted, heuristics are never 100% accurate (if they were, I could stop writing and go home), but besides some missed viruses here and there, they truly do help. What is better, they can detect more than before, have lower false positive rates than ever before, and can even repair.

We developed macro heuristics that emulate VBA code in macros inside *Word* documents and *Excel* spread-sheets. Emulating such code allows our anti-virus product to watch the macros copy themselves from one document to another in a virtual environment. This reduces false positives since the heuristic looks not at lines of code that may cause the macro to copy itself but at the actual copying. In addition, repair can be performed, as it is possible to identify the macros that comprise the virus (copy themselves from one document to another).

The idea of using emulation for heuristics produces fewer false positives and higher detection rates. While macro viruses are still the leading class of virus seen today, *Windows* malware, as described last month, is a big threat. Anti-virus vendors are beginning to apply similar techniques for *Windows* viruses. There are AV products that now perform emulation of *Windows* files in order to look for viruses. Emulation can be combined with traditional string scanning to uncover encrypted viruses, but also with heuristics to find new viruses.

The number of products with *Windows* virus heuristics is small, but on the rise. There are already products that examine *Windows* programs for network-aware code such as opening ports. Techniques such as these, as they become more robust and refined, will allow us to detect *Windows* network-aware malware before it is even able to infect. Expect to hear about more such technologies before the year is out.

New Engines and Architecture

Anti-virus companies are finally beginning to understand the dynamic nature of the virus world. New classes of virus are appearing every month, covering the spectrum from the first macro virus to the first Java virus. Highly complex viruses continue to appear ranging from the Bolzano *Windows* viruses to Win32/RemoteExplorer.

Initially, these types of virus presented a challenge to anti-virus software. Anti-virus products are developed to handle the viruses of the day. When a new class of virus appears, a new engine needs to be developed. Worse, this engine must be distributed across thousands of computers, potentially causing IT Administrators to reinstall or upgrade their existing anti-virus solution.

Luckily, anti-virus vendors realize the temporal and financial impact of having to re-roll out an anti-virus solution. *Network Associates*, for example, has recently announced *SuperDats* which attempts to alleviate this problem by not only being able to ship out data definitions, but also potentially special code to handle new types of virus. *Symantec* relies on its *NAVEX* technology to predict such a need. This separates the engine from the product, allowing it to be completely 'updatable' via the normal definitions that are shipped every week.

In addition to changes in the underlying architecture of anti-virus products, we can expect to see additional engines from *Windows* memory scanning to secondary effects engines. *Windows* memory scanning is a clear need, as discussed in the virus problem last month.

Although theoretically simple, practically speaking *Windows* memory scanning is not the easiest thing to do. Without it, however, anti-virus products may accidentally aid infection. By the end of the year, anti-virus products will need *Windows* memory scanning and disinfection or they will fall prey to many viruses.

Malware, unfortunately, also affects the Registry, system files, and INI files in order to help their spread. This type of damage is called 'side-effects'. Side-effects are rarely removed by anti-virus products. This has an obvious cost and time impact as IT Administrators may have to visit desktops to remove Registry keys or replace system files from known, clean backups. However, again, in the works are side-effects engines that will kill processes running in *Windows*, remove Registry keys, and repatch system files, bringing a system back to its original, uninfected state.

Other Security Solutions

There is more than anti-virus software. As predicted by Mikko Hyppönen (see *VB*, October 1999, p.2) it is only a matter of time before someone creates the true security suite. While security companies explore more than just anti-virus, they understand products such as content security and intrusion detection systems will aid existing products in preventing malicious content from entering the enterprise. Even home PC users can find personal firewall products in retail boxes these days.

Vendors understand the shifting trend, especially towards network protection, and this understanding is leading to better products designed to target such threats specifically. Already, these products exist and they will only become better over time, especially when combined with other key products to provide an overall security suite. By creating products directed towards the future trend we stay proactive rather than reactive.

Awareness and Education

The awareness of viruses and their effects is growing every day. One may be able to draw such conclusions from the record attendance at VB'99 in Vancouver [*our biggest conference yet! Ed*.]. Despite this, the genuinely damaging effects of W97M/Melissa.A, W95/CIH and the ExploreZip Worm are being felt in business. There is no doubt that Administrators are having an easier time explaining why spending money on anti-virus software, as well as other security solutions, is warranted.

While having more new users on-line exposes more people to possible threats, it also leads people to understand safe computing practices. As they spend more time on-line, people begin to understand that the common sense we use in the street every day should apply to when we use our computers, too. We do not eat a piece of candy we find in the street and likewise, we should not run programs we find on some unknown Web site, newsgroup, or in email. Again, while malware spreads itself via the Internet, so can education about safe computing. While education is probably the hardest battle in the war against viruses and we still have a long way to go in educating the average user, we are headed in the right direction.

Policies and Procedures

Education leads to solid policies and procedures. No doubt corporations understand that anti-virus products are not the final solution. Policies and procedures must come first. By implementing solid policies and procedures, corporations build a dam. Then, to protect against the eventual holes, they utilize anti-virus software and other security products. Awareness about the real threats has caused corporations to review and update, or even create for the first time, security policies and procedures.

Such policies and procedures can completely eliminate the threat of viruses that make use of security exploits. For example, VBS/BubbleBoy or the Internet Worm would have been complete non-threats had companies simply updated their products with the latest patches. A patch to prevent BubbleBoy had existed for over three months.

If you do not allow the malware in to begin with, then you do not have to worry about fending it off once it gets inside your gates. IT Administrators understand this and that in itself could potentially eliminate a large percentage of future virus problems.

Summary

So, while the next year will continue to bring about faster and more dangerous malware, anti-virus vendors will also bring about quicker and smarter technologies. Combined with the better educated computer user and secure corporate environment, customers will remain ahead of malware writers. Despite popular opinion, I believe this is one arms race in which we will remain ahead of the game.

FEATURE SERIES 2

Lotus Notes and Email Risks – Part 2

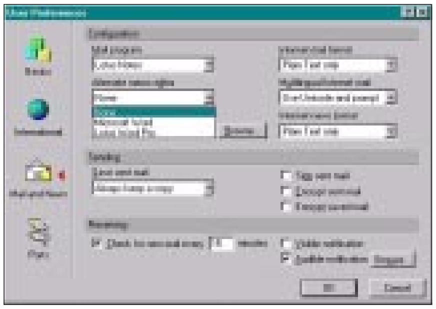
Martin Overton
ChekWARE, UK

Last month I covered most of the email risks posed by malware to *Lotus Notes* in its ‘out-of-the-box’ state, i.e. the worst case scenario. This second part of the series will address how to use the in-built security features in *Lotus Notes* to neutralise (where possible) or minimize the identified threats.

Addressing the Risks

Lotus Notes offers a few options to help minimize the threat from existing classes of viruses. I will look at these briefly below, and cover the *Notes*-specific functions later.

Let us look at macro viruses first, as they constitute the largest percentage of outbreaks each month. Currently, *Lotus Notes* does not allow you to stop attached OLE compound files which are infected from being launched by a user’s intervention (loaded into *Word*, *Excel* etc). What you can do is encourage the use of the View option on the attachment dialog. This will allow you to read the *Microsoft Office* file without running any macros or VBA within the document or spreadsheet. Also, ensure that the default Document Memo Editor is set to None, rather than *Word* or *Lotus WordPro* (see picture below).



Encourage the use of portable document formats that cannot contain VBA code, such as *Adobe Acrobat*. I would have suggested Rich Text Format (RTF) but this can easily be subverted (as illustrated by WM/Cap) and therefore, unless you are prepared to inspect the file format with a hex or ASCII editor, you cannot be 100% sure that the file really is an RTF file. Even if it is, please be aware that while macros are stripped, any embedded objects (which may contain viral content) are not.

Apart from banning executable attachments (like COM and EXE) which, unfortunately, is looking more and more attractive and even good policy, there is little more that you can currently do to reduce the risk of file infectors. Let’s put this into perspective though – this risk (until recently) only accounted for around 5% of reported virus outbreaks each month. I do not mean that it is a non-threat, but it does need to be taken in context against the preponderance of macro viruses.

Disk images are not generally passed around but the associated risk, while small, needs to be understood. Prevalence tables indicate that Boot Sector viruses average around 10% of all monthly virus outbreaks. The most prudent solution is to ban disk images in much the same way as executable attachments.

1999 appears to have seen the revival of file-type malware, especially Worms like W32/Ska (aka Happy99) and the many W32/Explore.Zip variants. In fact, the former appears to have caused the reports of file-infecting/affecting malware outbreaks to jump to over 16% in April of 1999 and average out at around 12% for the rest of the year. There seems to be little that *Lotus Notes 4.6/5.0* security features can do about them.

Any Good News?

With *Notes 4.6* or *5.0* there is a security facility known as the ECL (Execution Control List). The ECL – in *Notes 5.0* there are three distinct ECLs – allows you to restrict access to specific functions that code embedded in the *Notes* email (or other *Notes* document/form) can use, if allowed. The ECL is controlled via the use of digital signatures which allow you to restrict/grant access to functions by a specific signature or lack thereof.

This is best thought of as a type of Access Control or Behaviour Blocking. It does *not* mean that the signed code is safe, just that it is signed. All this gives you is proof that the code was signed by its owner and has not been changed since its signing, nothing more.

ECLs *only* affect the email’s embedded auto-run (auto-execute) code, which may be used to auto-launch an attachment or any number of other functions. This does *not* stop a user running an attachment or clicking on a button or hotspot, thereby launching a Trojan, Worm, virus or other code behind these functions. When an ECL rule is triggered (i.e. a signature or lack of signature exceeds the bounds of its authorized security settings) an ECL pop-up box will appear (see below) which offers the option to trust the code and the signatory of it.

This, at least, gives you non-repudiation as a stick to beat the signatory with in case of unwanted effects, such as a Worm, virus or other malware. But, and it is a big but, this dialog box still allows the user to accept/run the embedded code/action (trust the signatory, once or always). Haven’t we seen this type of approach somewhere before? If in doubt, delegate the responsibility to the user, then it is their mistake!

ECL Setting

Lotus says: '*By default, no scripts or formulas, whether signed or unsigned, can execute on your workstation without displaying a warning message. However, scripts or formulas run from any database created with a template that ships with* Notes *are signed "Lotus Notes Template Development/Lotus Notes", and this signature has complete execution access* [including the mail database template].'

Workstation security limits the following:

Access to the file system

Access to the current database

Access to environment variables

Access to non-*Notes* databases

Access to external code

Access to external programs (this option affects the ability to create or modify OLE objects)

Ability to send mail

Ability to read databases other than the current one

Ability to modify databases other than the current one

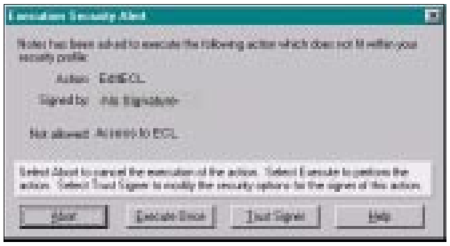
Ability to export data

Access to the Workstation Security ECL

Using wildcards in the execution control list: *'You can enter a wildcard in a name in the execution control list, thus extending access to everyone whose hierarchical name contains a particular element. For example, you can enter */Acme to extend access to all users whose hierarchical names end in /Acme.'*

As you can see, the key here is that the ECL settings only affect agents, scripts and macro functions included in databases, forms and fields, *not* attachments.

The dialog box below shows what happens when an ECL setting is tripped, in this case, to *Edit ECL* for the workstation. This would allow the workstation (clients) security level to be altered to anything the author intended!



Execution Security Alert

Notes has been asked to execute the following action which does not fit within your security profile:

Action: EditECL

Signed by: -No Signature-

Not allowed: Access to ECL

Select Abort to cancel the execution of the action. Select Execute to perform the action. Select Trust Signer to modify the security options for the signer of this action.

Get the feeling that you could be looking at the macro warning dialog in *Microsoft Word 97* or *Excel 97*? I wonder just how many users would just click 'Trust Signer' without thinking, just as they do for *Office* Macro warnings? This risk can be removed by the Administrator locking the clients' access to change their ECL.

Interestingly, *Lotus* added the further option to ECLs in *Notes 5.0*. You can also restrict access to signed Java applets and JavaScript applications. Select either 'Java applet security' or 'JavaScript security' in the Execution Control List and go through the list of access options you want to give to each signatory.

A user who shares a computer with others can set up his or her own ECL. The ECLs are unique to each person's User ID. Yet *Lotus* still offers no option to restrict the launching or detaching of file attachments by the users themselves. One wonders why not? Surely offering such a facility would ultimately help to negate some of the risks posed by sharing *Office* files?

Of course, I am playing devil's advocate here. No current groupware/*Office* Suite offers anywhere near the level of security that *Notes*/*Domino* offers. Nevertheless, I would like to see this feature added.

Let us take a look at this proposed option. Say, for example, we decide that users may 'View' attachments, but not 'Launch' (execute) or 'Detach' them. This would kill the risk from VBA macro viruses dead when sent as an attachment to a *Notes* client protected in this way. The *Lotus Notes* viewer can handle many file types (including *Microsoft Office* formats) and they do not run VBA macro code when the attachment is being viewed via the internal *Notes* viewer.

I am confident that I am not the only one who would like to see the following extension to the ECL implementation. It would be nice to see the ECLs extended to allow the optional blocking of attachments, so that these can only be 'Viewed' (in the in-built viewer), rather than 'Launched' (run) or 'Detached' (saved to disk).

The Bottom Line

Lotus Notes is, in my opinion, the package with the tightest and best-integrated security facilities of those groupware products available for use within most corporations. *Lotus Notes* security is multi-layered (with seven distinct layers in all) and in many ways can be likened to an onion – even if one layer of security is attacked and defeated there are other layers still to be bypassed.

The key to ensuring *Notes* is secured against targeted attacks is simply good, solid administration. Ensure that clients only have the minimum access rights necessary to perform their jobs. Proper use of the ECLs can minimize or neutralize such an attack.

Finally, there are products on the market that can be used to improve the level of virus protection in *Lotus Notes*. This is true for existing classes of virus and there are a few products which include features to help protect against *Lotus Notes*-specific threats. Scanning for viruses in *Notes*/*Domino* servers is required because otherwise *Notes* databases/email can become foxholes for viruses to hide out in, waiting to strike out again.

I hope I have given you a few things to think about as well as made you aware of some of the risks and solutions for *Lotus Notes*. More information can be found in a paper I presented at the 1999 *Virus Bulletin* conference in October. This paper is available at http://www.arachnophiliac.com cmindex.htm.

INSIGHT

Nick, Nick – Who's There?

Nick FitzGerald
Computer Virus Consulting Ltd, New Zealand

I was born in 1961 in the back-blocks of the Nelson province of New Zealand. This was the heart of New Zealand's tobacco- and hop-growing area. Perhaps fortunately, I have never been much interested in products made from the former, however, certain imbibations [*the quintessentially FitzGeraldian habit of coining words has not waned since he left his post at* VB*! Ed.*] prepared from the latter have been known to pass my lips over the years.

My family moved to Christchurch – the main city in the South Island – shortly before I began school. I don't recall anything profoundly significant from my school days, but did enjoy pulling things apart (and putting them back together, mostly successfully) in my attempts to better understand how things worked.

Crazy About Computers

My interest in computers developed relatively late compared to what I've read of the many who have been profiled in previous *Virus Bulletin* Insight columns. At highschool I missed selection for the 'advanced' mathematics class that got to use the school's first computer (I think it was a retired PDP from the local university) and my friends weren't into computers at the time.

Part way through university, it seemed clear that computers were going to play an important role in anything I would end up doing. Cheap *IBM* clones had just started flooding the local market and, despite knowing little about them, I bought one. My first computer was an *XT* with a 4.77MHz 8088 CPU, 640 KB RAM, two floppy drives and a mono *Hercules* graphics card and monitor. It cost pretty much what an 'entry level' 500MHz *Pentium* costs today, and within less than a year I bought my first upgrade for it – a 32 MB hard drive priced at about 25% of the original cost of the whole machine.

Initially, I mainly just used it for simple word-processing and spreadsheet tasks, but early versions of programs such as *Norton's Utilities* and my interest in pulling things apart soon had me delving deep into DOS file system internals and the like. This later paid off later as I did small data recovery jobs, getting files off 'corrupted' diskettes and doing a few hard drive recoveries too.

As I increasingly lost interest in my Masters Thesis, I found more and more computer-related work around the campus to occupy me, eventually landing a job in the computer centre there. Over the next several years I became expert in most things PC-oriented that were useful in supporting departmental computing within the university. One of several interests – a 'luxury' of working there – was early access to email and Usenet. This allowed me to search out, and keep up to date with, computer virus information.

The Good Guys

Perhaps inevitably, given my location in New Zealand, my first virus experience was with Stoned. I do not recall any of the specifics – just another task in a busy support schedule, I guess. Written and released in the capital, Wellington, this simple Boot infector spread around the country fairly quickly and then spread offshore. We were well aware of its existence due to media coverage and contact with support staff at Victoria University in Wellington, who were the first to uncover it.

Although very commonly seen around campus, it was mostly found on floppies, and as the first generation of 'mass computers' (*XT* clones similar to my first machine) were replaced, Stoned's prevalence dropped away as 386's with 3.5-inch (primary) drives became more common.

Most of the very high prevalence viruses were seen around the university, at least in passing, with AntiCMOS and Junkie probably being the most common after Stoned. We also saw some of the first Hare infections (the joys of encouraging use of the 'Net) and I'm still at a loss as to how Boot-437 made it to the Christchurch area as early in its life as it did.

Over the years at the computer centre, I was a regular reader of, and occasional contributor to, the Virus-L mailing list. As Ken van Wyk – the list's founding moderator – gradually succumbed to the increasing workload of his real job, he asked me if I would take over as moderator. I did and enjoyed running the list until I moved to the UK as the Editor of *Virus Bulletin*, where I enveloped myself in work and did not find (nor make) time for the list. Several attempts to resuscitate it have failed for various reasons, but I am hopeful the latest restart (coinciding with this interview) will succeed. My good friend and another long-time independent anti-virus expert, Bruce Burrell, has agreed to co-moderate the list and help 'keep me on track'.

I really enjoyed my time at *VB*. However, eventually the distance from family and friends in New Zealand – and particularly from my partner, Jessica – became too great and I decided to move back, or at least much closer so more regular visits were plausible. Jessica was really supportive of my decision to leave the university and move half way round the world to take up the *VB* Editorship, and I am very grateful for her understanding and support of my work. Currently I am building a small anti-virus consultancy and am very busy working on a contract to supply material for *Computer Associates'* virus encyclopedia.

No Leisure for Pleasure

As for free time – what's that? I used to really enjoy theatre technical work. I have not done any work of this nature for between three and four years now, but I used to relish it as a break from the computer centre. I'll do pretty much anything backstage, though I have a penchant for 'the flies' – the rope and pulley stuff making flats, scenery, drops and drapes 'fly' in and out from the roof. There is something about the challenge of playing your part, perhaps moving (or controlling) tons of equipment, in such a way that the audience doesn't really notice your role. Apart from special effects, much of the art of theatre technical work is in doing it so it is not noticed.

This is quite the opposite of my troubleshooting role at the computer centre, where being noticed was part of the deal. Of course, as a System Administrator as well, you tried to minimize the need to troubleshoot in the first place, but the university was full of diverse people with diverse needs and legacy systems, and most people wanted their machines to talk to as many other machines as possible.

Shape up or Ship Out

As to the future of the anti-virus industry, I hope it sees a huge change. To some extent we are seeing this already with the recent buzz-phrases being 'content scanning' and 'content management', but I think it has to adopt an even broader security focus than that.

As I have often said lately, scanning is a poor technological solution when malware can spread at the speed of light, and do so independent of human intervention. When we mainly transferred files by copying them to diskettes and putting the diskettes in physical mailers, the 'cons' of scanning were acceptable because of limitations in the computer systems themselves. We now have a corporate environment where those 'cons' have been heavily outweighed by ubiquitous desktop computing environments and a 'run everything' user (and Administrator!) culture. In such environments, the former 'pro' of scanning – its user convenience – is now all but a 'con' as well.

Anti-virus has to re-invent itself, in the corporate market, as the code integrity management sector of the security industry. If it does not, some other segment of the security market will fill this niche. In doing so, it will break the tyranny of the scanner and its unending upgrades, cutting the update lifeblood from anti-virus' most lucrative market.

Then the big names will be unable to survive on the paltry returns (after paying support costs) from the small user/ home user market which will not be able to use integrity management systems. Scanners will hopefully come to be seen as the second-class 'solution' they have always been.

The Bad Guys

We'll see more of both viruses and their writers, but we'll also see something of an increase in the current move away from traditional virus malware. Ubiquitous networking and desktop email client software means that if your goal is to write code that can spread itself around, you need not go to the trouble of writing a traditional virus.

Self-mailing Trojans or 'email Worms' are clearly on the increase and it may not be long before other attack strategies that do not depend on the distribution of unwittingly infected files are seen prospering. The increasing trends of the large OS – and sub-system developers mixing data and code blithely and not providing foolproof mechanisms to prevent the 'viewing' applications from running or interpreting the code elements – makes many of these things so much easier for the bad guys. Maybe it is time the corporate buyers put their collective feet down to stop things becoming worse in this regard?

Recently, there seems to be something of a groundswell of support for tougher penalties for cyberhoons, and while this will not directly solve anything, I think it is a positive move. For far too long, the easily influenced but inexperienced in the real ways of the world have seen the media especially, and much of society, treating 'hackers', virus writers and their ilk as 'techno-heroes'. A solid dose of realism, in the form of a stream of media coverage of the arrest and sentencing of these vandals, painting them as they are, will not hurt.

As it is an untried approach, I wonder where those who criticize it draw their beliefs that it could be harmful from? I agree that we should be working to alter perceptions at grassroots level, but that sort of change takes quite some time – perhaps generations – to become effective. That we have not been teaching computer ethics to the current (or next) generation of the Information Age from their earliest exposure to computers is a travesty, but it is not a reason to absolve today's cyberhoons of responsibility for the outcomes of their actions.

COMPARATIVE REVIEW

Prescribing the Right DOS

This month's Comparative Review comprises the annual peek at the most elementary of anti-virus species, the command-line DOS scanner.

The line-up of products is fairly small compared to the sixteen featured in the last DOS Comparative (see *VB*, January 1999, p.10). With the continued increase in the *Windows 9x* monoculture, the priority given to DOS scanners has diminished remarkably. Some of the products submitted (notably those of *Computer Associates* and *Symantec*) are the 'emergency' scanners supplied as part of the *Windows* product package. Nonetheless, such products provide the same detection capability as those on other platforms, and have been tested as usual.

Test-Sets and Procedures

The customary *VB* test-sets were used for testing – Standard, Macro, Polymorphic and In the Wild (ItW) sets. Importantly, the ItW set (both Boot and file virus components) was aligned to the October 1999 WildList (see http://www.wildlist.org/WildList/199910.htm).

The product submission deadline was 1 November 1999, a couple of weeks after the announcement of the WildList. Products which successfully detected all the ItW file and Boot virus samples have been awarded the now familiar VB 100% award.

The usage of DOS anti-virus scanners is far removed from that of their *Windows* brethren. They are typically used to perform scheduled on-demand scans, or for incident recovery. To reflect this, two important changes to the VB 100% criteria were introduced for this Comparative (and *only* this Comparative). Firstly, each product was set to scan all files even if this setting was not the default mode. Secondly, since the DOS scanners are designed for on-demand scanning from the command-line, the need for complete on-access ItW detection was removed.

Additions have been made to each of the test-sets since the last round of testing, (see *VB*, November 1999, p.16). Additions to the Polymorphic set include samples of the Win95/SK virus (see *VB*, January 2000, p.7) as well as samples of the E and F variants of W97M/AntiSocial (October 1999, p.6).

Recent months have seen the discovery of numerous *Windows*-specific file infectors, a selection of which have been added to the Standard set. Such samples include Win32/Oporto, the B and C variants of Win32/Bolzano and the *NT*-specific WinNT/Infis. A large number of macro viruses were introduced to the Macro test-set – samples include recent variants of W97M/Melissa, O97M/Tristate, W97M/Wazzu X97M/Vcx and W97M/Chack. Complete listings of the contents of each test-set can be found at the URL specified in the technical details section at the end of this review.

All the detection tests were conducted on identical machines, with the test-set stored in a read-only directory on a *NetWare* server. The scanners were run from the command-line whenever possible, as opposed to the menu-driven interface that some of the products offer. Importantly, each of the scanners was set to employ heuristics if available, the sensitivity of which was set to the lowest setting (irrespective of the default setting).

The speed of each of the scanners was tested by scanning the traditional *VB* executable and OLE2 Clean file sets. These tests also double as false positive tests, since no viruses should be detected in either. The speed test scans were performed with the products in identical configurations to those used for the detection tests – that is, scanning all files, with heuristics employed if available.

Alwil LGuard v7.70.34 (01/11/99)

ItW Boot	100.0%	Macro	96.3%
ItW File	99.8%	Standard	98.9%
ItW Overall	99.8%	Polymorphic	91.6%

The usual solid performance from *Alwil's* DOS scanner was marred slightly by its failure to detect three *PowerPoint* files infected with O97M/Tristate.C in the ItW set. It therefore missed out on a VB 100% award.

This lack of attention to files in *PowerPoint* format (the analysis of which was introduced some months ago in *Alwil's Windows* product) is responsible for some of the misses in the Macro set. Here, files infected with other Tristate variants, P97M/Vic.A, P97M/ShapeShifter and P97M/ShapeMaster were also missed.

PE samples infected with Win32/Oporto were, unfortunately, missed from the Standard set, as were three variants of VBS/First in both their VBS and JS incaranations. Misses in the Polymorphic set included the E and F variants of W97M/AntiSocial and the complex Win95/SK.8044. A selection of macro viruses, predominantly *Word*-based, were missed from the Macro set.

LGuard scooted happily through the executable Clean set, delivering a throughput of over 2000 KB/s and positioning the product at the speedy end of the field. Performance was slightly poorer in the OLE2 set, the throughput dropping to approximately 500 KB/s – at the other end of the field. No false positives were recorded in either set.

On-demand tests	ItW File		ItW Overall	Macro		Polymorphic		Standard	
	Missed	%	%	Missed	%	Missed	%	Missed	%
Alwil LGuard	3	99.8%	99.8%	123	96.3%	91	91.6%	11	98.9%
CA Vet Anti-Virus	0	100.0%	100.0%	60	98.4%	264	94.4%	1	99.9%
Command AntiVirus	0	100.0%	100.0%	3	99.8%	62	97.1%	0	100.0%
Data Fellows FSAV	3	99.8%	99.8%	30	99.1%	0	100.0%	2	99.9%
Dialogue Science DrWeb	0	100.0%	99.9%	11	99.6%	0	100.0%	6	99.7%
Eset NOD32	0	100.0%	100.0%	60	98.3%	21	97.2%	8	99.7%
GeCAD RAV	23	96.3%	97.0%	92	97.2%	8792	43.3%	236	85.0%
Grisoft AVG	0	100.0%	100.0%	52	98.4%	355	86.1%	90	96.4%
Kaspersky Lab AVP	0	100.0%	100.0%	19	99.3%	0	100.0%	0	100.0%
NAI VirusScan	0	100.0%	100.0%	12	99.6%	17	97.7%	0	100.0%
Norman Virus Control	0	100.0%	100.0%	11	99.7%	195	94.4%	6	99.7%
Sophos Anti-Virus	0	100.0%	100.0%	73	97.7%	191	94.9%	18	99.3%
Symantec Norton AntiVirus	0	100.0%	100.0%	34	98.9%	305	88.8%	1	99.7%

CA Vet Anti-Virus (01/11/99)

ItW Boot	100.0%	Macro	98.4%
ItW File	100.0%	Standard	99.9%
ItW Overall	100.0%	Polymorphic	94.4%

Currently, *Vet Anti-Virus* is not shipped as a standalone DOS product – instead a command-line program is supplied as standard with the other product packages. Nonetheless, the command-line scanner (RESCUE.EXE) has all the detection capabilities of the other *CA Vet* products.

Detection rates were as high as we have come to expect from *Vet*. Once again, all the ItW file and Boot sector viruses were successfully detected, earning *Vet* its third consecutive VB 100% award. A single sample remained undetected in the Standard set – one of the five PE files infected with the polymorphic Win32/Parvo (one of the first viruses to utilize socket communication in order to propagate itself). The bulk of the remaining misses were against the Macro test-set, where a variety of *Excel* and *Word* macro viruses were missed.

According to percentages, the poorest performance is against the Polymorphic set. This was due to *Vet's* failure to detect both the A and the B variants of ACG. However, on the upside, *Vet* was one of only four products to detect all the samples of the newcomer, Win95/SK.8044, thus deserving some credit irrespective of the percentages.

The scanning speeds observed were perhaps not as high as those typified by *Vet* in previous Comparatives, although they were sufficient for *Vet* to remain amongst the faster of the products tested.

Command AntiVirus v4.57.4 (31/10/99)

ItW Boot	100.0%	Macro	99.8%
ItW File	100.0%	Standard	100.0%
ItW Overall	100.0%	Polymorphic	97.1%

After picking up their first VB 100% award for more than 12 months back in November (in the *Windows 98* Comparative), the developers at *Command* will be pleased to see their DOS product reproducing the achievement this time round.

The clean sheet earned in the ItW sets was maintained throughout the Standard set, and was only lost thanks to misses in the Macro and Polymorphic sets. Three *Word* documents infected with W97M/Astia.Y account for the misses in the former, and samples infected with ACG.A and Win95/SK.8044 those in the latter. The detection of these polymorphics has been implemented in the product since a proportion of each of the sample collections was detected. However, the results suggest that further work is needed in order to detect all the samples – whether their detection is implemented more successfully in the next product version, time, and the next Comparative, will tell.

	Floppy Diskette Scanning speed		Hard Disk Scanning Speed					
	Clean	Infected	Executables			OLE2 files		
	Throughput (kB/s)	Throughput (kB/s)	Time (min:sec)	Throughput (kB/s)	FPs [susp]	Time (min:sec)	Throughput (kB/s)	FPs [susp]
Alwil LGuard	15.6	11.5	4:18	2119.9	0	2:25	547.1	0
CA Vet Anti-Virus	24.9	14.7	4:32	2010.8	0	0:54	1469.1	0
Command AntiVirus	19.5	24.9	7:31	1212.7	0	0:23	3449.3	0
Data Fellows FSAV	23.2	23.7	5:12	1753.0	[2]	1:02	1279.6	0
Dialogue Science DrWeb	15.1	12.3	19:06	477.3	1 + [17]	1:30	881.5	[1]
Eset NOD32	32.2	25.6	2:02	4483.1	0	0:20	3966.7	0
GeCAD RAV	14.7	13.8	31:08	292.8	1	1:09	1149.8	1
Grisoft AVG	11.3	19.9	2:31	3622.1	0	0:23	3449.3	0
Kaspersky Lab AVP	16.1	23.7	5:13	1747.4	0	1:19	1004.2	0
NAI VirusScan	20.3	14.4	8:24	1085.2	0	1:04	1239.6	0
Norman Virus Control	20.8	19.9	3:57	2307.7	0	0:24	3305.6	0
Sophos Anti-Virus	19.2	14.9	7:52	1158.8	0	1:09	1149.8	0
Symantec Norton AntiVirus	21.7	18.8	5:55	1540.7	0	1:15	1057.8	0

Command AntiVirus sped through the OLE2 Clean set at a rate far removed from that observed in the executable set. Happily, no false positives were observed in either set.

Data Fellows FSAV v3.0 (31/10/99)

ItW Boot	100.0%	Macro	99.1%
ItW File	99.8%	Standard	99.9%
ItW Overall	99.8%	Polymorphic	100.0%

As reported in last month's *VB*, the *Data Fellows Corporation* have recently changed their company name to *F-Secure Corporation* – a name more in tune with that of their anti-virus product line. However, since the product for this Comparative was submitted prior to this name change, it is referred to as *Data Fellows F-Secure Anti-Virus* (*FSAV*) throughout this review.

The *FSAV* incarnations for *Windows* have traditionally achieved high detection rates, thanks partly to the product's use of two engines, those of *F-Prot* and *AVP*. The DOS product submitted to this review only featured one engine – that of the latter. In fact, the product was an *F-Secure* badged version of *AVP Lite*, the stripped down DOS scanner from *Kaspersky Labs*.

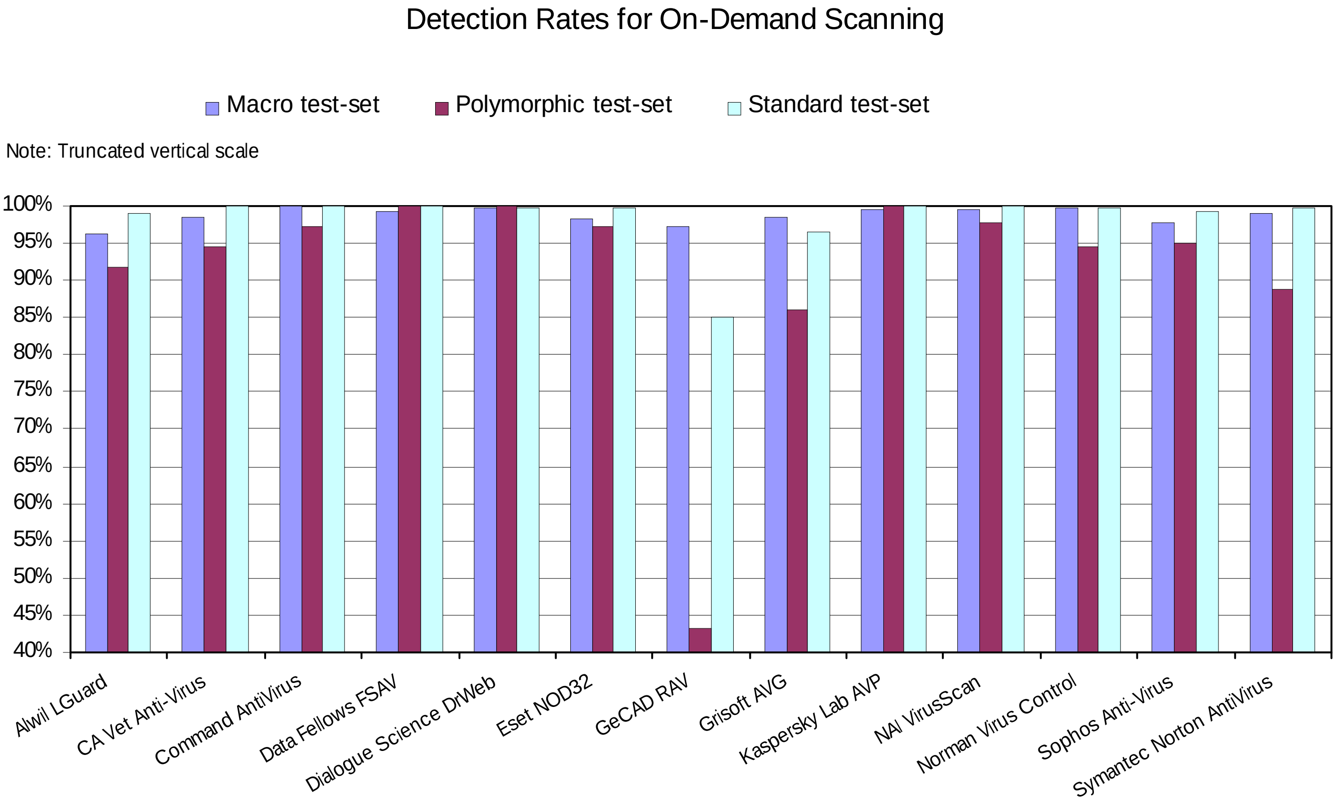
Despite not utilizing the *F-Prot* engine, *FSAV* still returned high detection rates across all test-sets. Thanks to the Russian virus engine it was one of only three products to detect all of the samples of Win95/SK.8044 in the Polymorphic set successfully – a worthy feat in itself. In fact, the only non-Russian product to achieve the same result was *Computer Associates' Vet Anti-Virus*.

Unfortunately, VB 100% award glory was prevented due to the failure of *AVP Lite* to cope with *PowerPoint* files. Thus, three samples infected with O97M/Tristate.C were missed in the ItW set, and a host of others in the Macro set.

The effect of *FSAV* only using one virus engine is perhaps most notable in terms of scanning speed, a field in which, traditionally, the product has not excelled in the past. The speeds observed during testing put *FSAV* somewhere in the middle of the pack when scanning both the executable and OLE2 Clean sets.

Dialogue Science DrWeb v4.14 (26/10/99)

ItW Boot	100.0%	Macro	99.6%
ItW File	100.0%	Standard	99.7%
ItW Overall	100.0%	Polymorphic	100.0%



As has been noted in previous Comparatives, one of the main strengths of *Dialogue Science's DrWeb* has traditionally been its detection of polymorphic file infectors. This was in evidence once more during this review – *DrWeb* being one of only three products to cope successfully with the entire contents of the Polymorphic set.

Unfortunately, a minor bug in the product (evident when the 'continuous running' – /GO – switch was employed) led to *DrWeb* attempting to disinfect certain infected files, despite the fact that the 'no disinfection' switches had been included on the command-line.

Initial results suggested that *DrWeb* had missed the extensionless O97M/Tristate samples, thereby missing out on the VB 100% award. However re-running the scans without the command-line *.* mask resulted in such files being scanned and detected as infected. Performance elsewhere was impressive, with misses few and far between. In fact, the average detection rate (across all the test-sets) was second only to *Kaspersky Lab's AVP*.

As ever, the overkeen *DrWeb* heuristics triggered on a few innocent files during the speed tests. In the executable set, one file was triggered as infected and 17 as suspicious. In the OLE2 set, no definitive false positives were registered, although one *Word* global template was reported as possibly infected. With the introduction of a 'no false positive' criterion into the VB 100% award requirement, it will be interesting to watch *Dialogue Science* re-tune the detection of their product to eliminate false positives, while maintaining (or at least minimizing the sacrifice in) detection rates.

Eset NOD32 v1.27 (29/10/99)

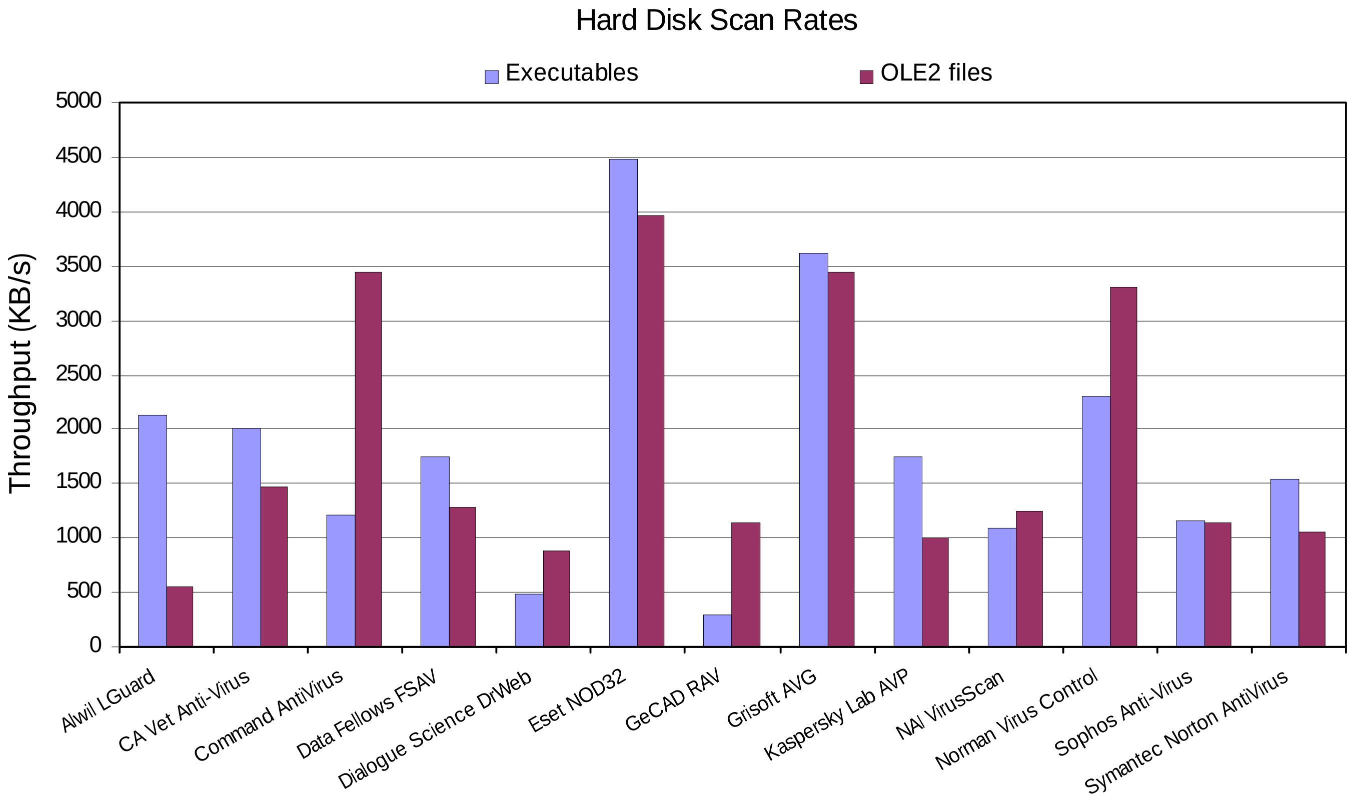
ItW Boot	100.0%	Macro	98.3%
ItW File	100.0%	Standard	99.7%
ItW Overall	100.0%	Polymorphic	97.2%

Eset's NOD32 starts the new year as it ended the last – in fine fettle, earning yet another VB 100% award. As it happens, *NOD* has earned a VB 100% award in each and every Comparative to which it has been submitted (since March 1998) except for those on *NetWare*.

The bulk of the missed samples were in the Macro set, where samples infected with XM/Soldier, W97M/Astia.Y, W97M/Marker.Y and the L, M, U and V variants of W97M/Melissa were missed (amongst others). Complete detection of the Standard and Polymorphic sets was prevented by eight DNA.1206 samples in the former, and all the samples of Win95/SK.8044 in the latter.

The observant reader may notice that *NOD32* missed some samples that it has detected successfully in previous Comparatives. This is due to the fact that this Comparative was run with each product's heuristics in their lowest setting. Had the product been run with its default level of heuristics, then a number of the missed samples listed here would have been flagged as possibly infected.

To round off a fine performance, the Slovakian offering also delivered the greatest throughputs during scanning of the Clean sets and floppy diskettes, returning scan rates of 4000 and 25 KB/s, respectively.



GeCAD RAV v7.50

ItW Boot	100.0%	Macro	97.2%
ItW File	96.3%	Standard	85.0%
ItW Overall	97.0%	Polymorphic	43.3%

GeCAD's Romanian Anti-Virus (*RAV*) has set some high standards in the last few Comparatives. In fact, it has received the VB 100% award in the last two reviews. Unfortunately, this success has been short-lived, and not repeated this time round.

The detection rates observed are significantly lower than have come to be expected – a factor attributable to a bug in the DOS4GW extender. Despite the developers at *GeCAD* suggesting that the bug would only manifest itself on a machine without HIMEM and EMM386 installed, this was not the case during testing. The resulting detection rate was the same for all DOS configurations upon which the test was repeated.

Grisoft AVG v6.087 (database 47)

ItW Boot	100.0%	Macro	98.4%
ItW File	100.0%	Standard	96.4%
ItW Overall	100.0%	Polymorphic	86.1%

While never awarded the VB 100%, *Grisoft's AVG* has put in strong performances of late. The Czech developers will no doubt be delighted to see that complete detection of both the ItW file and boot virus samples has managed to earn *AVG* the VB 100% award this time around, however. Detection rates in the other test-sets were slightly lower, especially in the Polymorphic set where *AVG* failed to detect samples infected with Win95/SK.8044, ACG.B and the E and F variants of W97M/AntiSocial.

Kaspersky Lab AVP v3.0.132 (23/10/99)

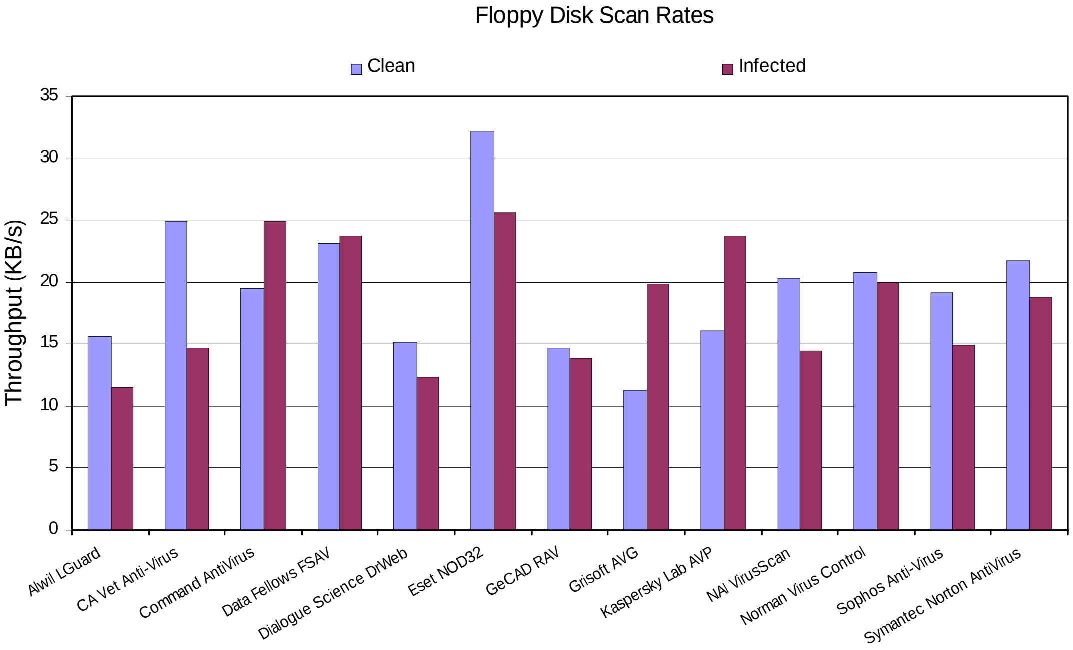
ItW Boot	100.0%	Macro	99.3%
ItW File	100.0%	Standard	100.0%
ItW Overall	100.0%	Polymorphic	100.0%

Unsurprisingly, *AVP* scoops yet another VB 100% award this month, detecting all the ItW boot and file viruses (unlike *AVPLite*, which failed to cope with *PowerPoint* files). A motley selection of macro viruses were missed in the Macro set, including the *Excel*-infecting X97M/Clonar.A and X97M/Vcx.D, and the *Word*-infecting W97M/Astia.Y and W97M/Mck.H.

Speedwise, there is little to report for *AVP*. Throughputs of approximately 1,750 and 1,000 KB/s were observed for scanning of the executable and OLE2 Clean sets respectively, positioning *AVP* amongst the bulk of the products.

NAI VirusScan v4.0.4.4049 (27/10/99)

ItW Boot	100.0%	Macro	99.6%
ItW File	100.0%	Standard	100.0%
ItW Overall	100.0%	Polymorphic	97.7%



Another strong performance from *VirusScan*, missing only 33 samples encompassing five viruses over all the test-sets. Complete ItW detection earns the product its first VB 100% award since March of last year.

Four macro viruses were missed in the Macro set, namely X97M/Clonar.A, W97M/Astia.Y, W97M/Venus.A and one of the four samples of W97M/Walker.B. This latter macro virus employs on-the-fly encryption and decryption of its code, perhaps explaining *VirusScan* having missed one of the samples (although the same sample has been detected successfully by other *VirusScan* product versions since early 1999).

In terms of speed, *VirusScan*, once again, surprises no-one. Fairly middling scanning speeds were observed in terms of executable and OLE2 scanning, and the product, happily, registered no false positives.

Norman Virus Control v4.72 (01/11/99)

ItW Boot	100.0%	Macro	99.7%
ItW File	100.0%	Standard	99.7%
ItW Overall	100.0%	Polymorphic	94.4%

Another product which did not disappoint is *Norman's Virus Control* (*NVC*). Scooping its eleventh VB 100% award since January 1998, *NVC* continues to deliver the detection rates with which it has come to be associated.

Over the entire test-set, 216 samples were missed. The bulk of these were registered in the Polymorphic set, where all the samples of ACG.A and Win95/SK.8044 were missed. In the Standard set, *NVC* joins three other products in failing to detect any of the samples infected with the PE-infecting Win32/Oporto.

NVC delivered extremely respectable throughputs during scanning of the Clean sets. Throughputs of almost 2,500 and 3,500 KB/s were returned during executable and OLE2 file scanning respectively. True to *NVC* tradition, no false positives were observed during the speed tests – a useful factor given the ‘no false positives’ condition soon to be added to the VB 100% award criteria.

Sophos Anti-Virus v3.27 (01/11/99)

ItW Boot	100.0%	Macro	97.7%
ItW File	100.0%	Standard	99.3%
ItW Overall	100.0%	Polymorphic	94.9%

Having taken something of a winter break from VB 100% awards, *Sophos Anti-Virus* (*SAV*) continued where it left off in May 1999, and detected all the ItW samples successfully. The VB 100% award, absent for the past three Comparatives, is back on the *Sophos* mantelpiece.

Elsewhere in the test-sets, a number of the recently introduced macro viruses were missed (including, the D and F variants of X97M/Vcx, X97M/Manalo.E, W97M/Astia.Y, and a few variants of W97M/Chack), in addition to a small number of missed samples from the Standard set. Interestingly, only four of the Win95/Sk.8044 samples were detected – the complex polymorphic engine successfully managing to elude *SAV*. Additionally, all the samples of ACG.A were missed from this set.

As ever, *SAV* produced no surprises in the Clean set, delivering scanning speeds characteristic of the bulk of products, and registered no false positives.

Symantec Norton AntiVirus (25/10/99)

ItW Boot	100.0%	Macro	98.9%
ItW File	100.0%	Standard	99.7%
ItW Overall	100.0%	Polymorphic	88.8%

Alphabetically the last contender in this Comparative, and the final recipient of the VB 100% award, *Symantec’s Norton AntiVirus* (*NAV*) picks up its eighth award. As with a couple of the other product developers, *Symantec* does not produce a specific DOS version of *NAV*. Instead, the version tested was *NAVDX* – the 'emergency' command-line scanner shipped with the *Windows* product.

Detection rates across all test-sets were high – that in the Polymorphic set was the lowest. This was due to all the samples of ACG (A and B variants), Win95/SK.8044 and W97M/AntiSocial.F being missed. In the Macro set, only one of the three P97M/Vic.A samples was detected, as were all the samples of the B, C and D variants of W97M/Lys.

Conclusions

Apart from the obvious glitches, once again all the products have exhibited impressive detection rates. Ten of the thirteen products detected all of the ItW samples successfully during on-demand scanning, earning themselves the VB 100% award – so congratulations to *CA Vet*, *Command AntiVirus, Dialogue Science DrWeb, Eset NOD32, Grisoft AVG, Kaspersky Lab AVP*, *NAI VirusScan*, *Norman Virus Control*, *Sophos Anti-Virus* and *Symantec Norton AntiVirus*.

Samples of the complex polymorphic Win95/SK.8044 (the sample set consisting entirely of infected EXEs for this review) posed problems for the products. Five of them did not manage to detect any of the infected files. Of those products which had implemented Win95/SK.8044 detection, three managed to detect a fraction of the sample set (*Command AntiVirus*, *NAI VirusScan* and *Sophos Anti-Virus*). Only four offerings managed to detect all of the samples – *Vet Anti-Virus*, *AVP Lite* (submitted by *Data Fellows*), *DrWeb* and *AVP*. It will be interesting to monitor how the future versions of these products cope with other variants of the polymorphic Win95/SK as they are added.

Plans are afoot for the addition of further requirements to the VB 100% award. As from the June 2000 review (*Windows 98*), the criterion of no false positives during scanning of the *VB* Clean sets will be introduced.

Another point of interest in future reviews will be how well the products cope with archives containing infected files – an area which will be investigated in the next review (*NT*) for the April 2000 issue.

Technical Details

Test Environment: Server: *Compaq Prolinea 590,* 90MHz Pentium with 80 MB of RAM, 2 GB hard disk, running *NetWare 4.10*. Workstations: Three 166 MHz Pentium-MMX workstations with 64 MB RAM, 4 GB hard disks, CD-ROM and 3.5-inch floppy, all running MS-DOS 6.22 and *Novell* ODI/VLM drivers. The workstations could be rebuilt from image back-ups, and the test-sets were stored in a read-only directory on the server. All timed tests were performed on a single machine that was not connected to the network for the duration of the timed tests, but was otherwise configured identically to that described above.

Virus Test-sets: Complete listings of the test-sets used are at http://www.virusbtn.com/Comparatives/DOS/200002/test_sets.html.

A complete description of the results calculation protocol is at http://www.virusbtn.com/Comparatives/Win95/199801/protocol.html.

OVERVIEW

Exchange of Ideas

Fraser Howard

As has been indicated in recent issues, the testing of groupware anti-virus products has already started at *VB*. In fact, the first two reviews will feature in the March 2000 issue. This brief article is intended to serve as a prelude to these reviews. As such, it provides a brief insight into the principles of the *Exchange* messaging system, including the intricacies involved with message routing.

First of all, exactly what are we referring to when using the term 'groupware'? A convenient definition (and the one used at *VB*) of groupware applications is 'those applications which provide environments that support communication and collaboration between users or groups of users'. The concept of Information Sharing is central to any groupware application, be it sharing within an organization, between separate sites of an organization, or between different organizations. So, where do viruses come in? Well, facilitating the sharing of data also facilitates the sharing, and hence propagation of, computer viruses. A document stored in a publicly accessible folder presents the potential to infect workstations throughout an organization if no anti-virus measures are in place.

So now we know what the term groupware represents, and why anti-virus measures are a necessity in a groupware environment, let us take a closer look at *Exchange* itself.

The Exchange Server

In order to be able to review anti-virus products for *Exchange* meaningfully, it is first essential to build up an understanding of the principles underlying the operation of the *Exchange* messaging system itself.

Exchange provides a groupware solution through the use of a series of databases, allowing users to access and share information or exchange messages using a variety of network protocols. It is the manipulation of these databases that forms the backbone of the messaging system.

Four main components provide the functionality of *Exchange* – Directory Services, Information Store, Message Transfer Agent and System Attendant. The basic role of each is outlined below. Directory Services (DS) is responsible for creating and managing the storage of all objects (be they mailboxes, servers etc) within the *Exchange* organization. All of the objects are stored in the directory database (DIR.EDB). The Information Store (IS) stores mailbox and public folder data. It consists of two constituent databases, the private and public IS (PRIV.EDB and PUB.EDB, respectively). When a user composes an email, a message is created in the relevant part of the IS. The next *Exchange* component, the Message Transfer Agent (MTA), is responsible for the routing of messages between servers. As such, it is only needed when the mailboxes of the people that the message is being sent to are not situated on the local *Exchange* server.

The final component of *Exchange* is the System Attendant (SA). This can be conveniently regarded as a form of monitor, checking on the status of *Exchange* services, and maintaining the system logs. In addition to this, the SA is also responsible for building the routing tables, and for generating the email addresses of new users.

Message Routing

The path that a message follows once composed and sent, is principally determined by the location of the designated message recipients. However, as will become apparent, the exact route that is followed is also determined by the relative costs of each of the possible routes.

The simplest scenario to consider is that of message transfer between two users in the same *Exchange* site, whose mailboxes reside on the same *Exchange* server (i.e. they share a 'home' server). In this instance, message transfer is handled entirely by the IS service and the private IS. The IS service resolves the recipient address through polling the DS, and, assuming all are local, places a single copy of the message in the private IS. The appropriate recipients (if they are using MAPI clients) are then informed of the new message's arrival.

The next scenario assumes that the mailbox of the message recipients is located within the same *Exchange* site, but on a different *Exchange* server. In this instance, when the IS service determines that the recipient is not local, the message is sent to the MTA on the local server, which in turn determines the name of the remote server that hosts the recipient's mailbox. The local MTA then opens an RPC connection to the MTA on that remote *Exchange* server, and transfers the message. Assuming that the remote server is that which hosts the recipient's mailbox, then the message is delivered to the IS service, which subsequently handles writing the message to the private IS.

The final scenario to consider is that involving message transfer between users located on servers in different sites. In this instance, as in the preceding case, the IS service on the sender's server determines that the message recipient is located in a different site, and passes the message to the MTA. The MTA then uses the Gateway Address Routing Table (GWART) in order to determine message routing details (all possible routes to the destination), and then select the specific route (based upon costs). The MTA searches the GWART for a match to the recipient address, and then determines the routes available. Subsequently, a route is chosen based upon the costs which have been assigned to each route. The message is then transferred to a remote server (e.g. the MTA of a foreign *Exchange* server, or an SMTP server etc).

Email Clients

In order for the *Exchange* server to be of any use, users must be able to connect to it and access the information it manages. The email client products that are used to connect to the Exchange system fall into one of two categories – those which use the MAPI interface and those which use Internet protocols (e.g. IMAP4, POP3) in order to talk to the server. The functionality that each product provides varies enormously. At one end of the scale is *Outlook* (either a MAPI or Internet client), which provides numerous functions from calendaring and scheduling through to the usage of public folders. At the other end of the scale are POP3 clients such as *Pegasus* or *Pine*.

Exchange Anti-Virus Products

Now that we have considered message routing within *Exchange*, let us look at the integration of anti-virus products into the *Exchange* environment. Putting on-the-fly email scanning to one side for a moment, an essential requirement of an *Exchange* anti-virus product is to be able to scan within the IS – including both its public and private constituent databases. It should be noted here that although not the default location, it is possible to configure *Exchange* to store certain user mail on the local, client machine.

The above requirement fulfilled, perhaps the most important function of such a product is to provide scanning of all incoming and outgoing email, ensuring that all file attachments are clean. The implementation of such functionality is far from simple, and the final product has to meet certain requirements. For example, the product must be able to cope with a high throughput of both incoming and outgoing mail. Thus, issues of scanner overhead become a concern.

One such method is to use notification events as the message is routed between the IS and the spooler. However, in such a scenario there exists the possibility of a delay between receipt of notification and scanning. Thus, under heavy server load, it might be possible for a message to reach its destination prior to being scanned. In order to avoid such a scenario, an alternative principle is to intercept mail as it is routed through the server (for example using a hook provider or pre-processor). In this way, messages are intercepted as they are routed between the message store and the spooler, queued for scanning. The messages are not released from the queue for subsequent routing until scanned, irrespective of server load.

Summary

This article serves as an introduction to the principles behind the *Exchange* messaging system. Future issues of *VB* will feature full reviews of anti-virus products for *Exchange*, in which the products are put through their paces as issues from installation and configuration, to detection rates and imposed overheads are investigated. In this way, the reviews will serve to aid the choosing of an anti-virus product for the *Exchange* messaging system.

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

SUBSCRIPTION RATES

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

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

Editorial enquiries, subscription enquiries, orders and payments:

Virus Bulletin Ltd, 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 ninth annual *EICAR* conference, also known as the first European Anti-Malware Conference, takes place in Brussels, Belgium, from 4–7 March 2000. For further information, to place your booking reservation or to order a timetable of events visit the *EICAR* Web site at http://www.eicar.dk/.

The latest anti-virus firm to suffer PR embarrassment is Korean-based *Dr Ahn's Laboratories*, which shipped infected software from their Web site at the end of December 1999. A self-extracting archive file containing the latest version of their anti-virus engine was infected with the Win95/Lovesong virus – a fairly new, prepending *Windows* PE file infector.

A two-day course entitled Practical Anti-Virus will be run by *Sophos* on 21 and 22 March 2000 at the organization's training suite in Abingdon, Oxfordshire, UK. A one day training course called Best Practice for Sophos Anti-Virus will take place on 23 March. For further information, or to reserve your place, contact Daniel Trotman; Tel +44 1235 559933, email courses@sophos.com or visit the company's Web site http://www.sophos.com.

Now in its fifth year, InfoSecurity Europe is the largest dedicated Information Technology event in Europe. **InfoSec 2000 will take place at the National Hall, Olympia, London from 11–13 April 2000.** The show includes exhibitions and talks on various subjects including virus protection, firewalls, network security, e-commerce and Web security. There will also be a series of 46 free, on-floor seminars on topics such as *Windows 2000* and *Linux*. For more details or to make a booking contact Yvonne Eskenzi; Tel +44 2084 498292 or email yvonne@eskenzi.demon.co.uk.

Symantec and *Azlan Training* have launched an independently certified programme to train anti-virus engineers. ***Symantec's* Certified AntiVirus Engineer course** lasts five days with a final examination. For details contact Lucy Bunker; Tel +44 1628 592222 or email Lucy.Bunker@symantec.com.

F-Secure Corporation* has introduced version 3.0 of *F-Secure Anti-Virus for Firewalls, ensuring higher and faster throughput when scanning large amounts of data for a distributed workforce. For details contact; Tel +1 408 938 6700, email Pirrka.Palomaki@F-Secure.com or visit the Web site http://www.F-Secure.com/.

***Oxford Solutions Ltd*, founded by two ex-*Dr Solomon's* managers, is to market and support *Kaspersky Lab's AVP* in the UK.** For more details contact Phil Watts; Tel +44 1844 210300 or see http://www.oxford-solutions.co.uk/.

The fourteenth annual Vanguard Enterprise Security Expo 2000 will be held at the Atlanta Hilton and Towers, Atlanta, Georgia, on 15 and 16 May 2000. For further information contact *Vanguard*; Tel +1 714 9 390377, or see http://www.vipexpo.com/.

NAI* has launched *Virus Interface for Protective Early Response* (*VIPER*) for *Linux to create anti-virus solutions for e-business applications, such as the scanning of Internet email traffic. For more information contact Caroline Kuipers; Tel +44 1753 827500 or see http://www.nai.com/.

***Norman Data Defense Systems* has released *Norman Virus Control* (*NVC*) for *MIMEsweeper* DLL.** *MIMEsweeper*, from UK firm *Content Technologies*, is a gateway for email, ftp and Web traffic. System requirements are *Windows NT 4.0* server or workstation, *MIMEsweeper 3.2*, *NVC 4.70* or later. For details, contact Dawn Cooke; Tel +44 1908 520900, email dawn.cooke@normanuk.com or visit the Web site http://www.norman.com/.

Panda Software* has announced the release of *Global Virus Insurance 24h-365d for *Lotus Notes* (v 4.5–R5). The Spanish company claim that theirs is the first anti-virus program which is installed on the *Notes* client not on the server, thereby leaving system performance unaffected. For more information contact Maria de Vera; Tel +34 913 013016 or see http://www.pandasoftware.com/.

The *Computer Security Institute* (*CSI*) has released details about its 10th annual Network Security conference and exhibition this year. **NetSec 2000 will be held at the Hyatt Regency Embarcadero in San Francisco from 12–14 June.** For more details contact *CSI*; Tel +1 415 9052626 or visit http://www.gocsi.com/.

UK-based security firm *CenturyCom* is offering consultancy for business corporations wanting to assemble an internal security policy and policing mechanism. **For more details and advice on how to 'build your company's corporate security A-team'** visit the Web site http://www.centurycom.co.uk/.