ISSN 0956-9979 SEPTEMBER 2000

# VIRUS BULLETIN

THE INTERNATIONAL PUBLICATION ON COMPUTER VIRUS PREVENTION, RECOGNITION AND REMOVAL



## *IN THIS ISSUE:*

• **Prima donna:** a certain email-borne worm has been hogging centre stage (groan!) far too much lately. A full analysis is featured on p.6.

• **International departures:** we hear from both Greek and Indian readers on the state of their respective nations in the fight against viruses. Feature 2 starts on p.14.

• **Be the best:** submission numbers have slightly improved on July 1999's *NetWare* Comparative Review, but has performance? Find out on p.17.

## CONTENTS

# COMMENT

*“ Microsoft-bashing was the sport de jour ”*

## Thank you Microsoft

I am no apologist for *Microsoft*, as anyone who has worked with me will confirm.

It is a large corporation producing more software than any other, at least in sheer bulk if not diversity. Much of that software is mediocre but marketed well, which seems to be what counts these days. It is easy to argue either (or both) that the resulting development of a computing monoculture dominated by one company’s OSes, products and visions is a bad thing (the seeding of an ‘Evil Empire’) or a boon for users (who no longer need to become nerdy computer geeks to obtain some of the benefits this supposedly ‘enabling’ technology promised for so long).

However, media coverage of virus incidents over the last few months whipped up a storm of protest and ill-informed comment driven by advocates of the first position above. The commentary I am talking of focussed on claims that *Microsoft* and its ‘insecure’ products caused the LoveLetter virus ‘problem’ and many other (then recent) virus outbreaks. Largely started by those with vested interests in denigrating *Microsoft*, this commentary tended to ignore the flip-side of the equation. As a user-advocate, I feel compelled to challenge such loud and provocative anti-user posturing.

Bill Gates certainly did not help matters with his ludicrous statement that things would be even worse on the virus and security front if *Microsoft* was broken up. Whatever planet Bill has retired to, Steve Ballmer (as CEO) should ensure that Bill stays there and remains incommunicado with the press! Regardless of Bill’s comment though, within a few days of the initial LoveLetter outbreak, I was interviewed by many journalists, ostensibly interested in the virus.

In several cases though, the questioning quickly turned to rather blatant badgering. When I did not proffer the opinion freely, I was asked for my confirmation that ‘much of the blame for this must be laid at Microsoft’s feet’ or similar. Refusing to play ‘put the journalist’s opinion in the expert’s mouth’ (you never know where used opinions have been), I found my views were apparently less expert or otherwise less press-worthy than they have been previously.

Hard on the heels of the anti-trust case, *Microsoft*-bashing was the *sport de jour*. And, if it was unfashionable to choose not to play, it was apparently reprehensible to take a contrary position. I am still no apologist for *Microsoft*, but *Microsoft* is not at fault here – at least, no more than any other large software maker is or would be were *Microsoft* not the dominant player in the desktop systems and application software market. *Microsoft* is guilty of making software that more people choose to use, but it is the people who choose to use it that are responsible for that use.

LoveLetter, like Melissa and ExploreZip before it and NewLove and Stages after it, required its recipients to choose to run an unknown program. Despite some confusion in the general media, LoveLetter is not like BubbleBoy and Kak, whose existence and progress can justifiably be attributed (at least partly) to *Microsoft*. Those viruses inveigle their way into victim systems because of poor judgment and/or implementation errors made by *Microsoft* developers and missed by *Microsoft* software testers. However, that so many users decided to run the LoveLetter email attachment simply because they recognized the sender’s name or email address, or because they were beguiled by the simple message and/or attachment name, cannot be laid at *Microsoft’s* door.

Neither *Microsoft*, nor anyone else, forced such a large chunk of the world’s computer users to use its products. Neither *Microsoft*, nor anyone else, forced such a large chunk of the world’s computer users to double-click that attachment. If you don’t like that it happened, by all means do something about it, but don’t hop on the ‘Microsoft allowed this to happen’ bandwagon. Users allowed it to happen, therefore users (or their managers) should accept responsibility for that and work on preventing a re-occurrence. Jumping on the ‘Microsoft is evil’ bandwagon may feel good in the short-term, but could result in much worse long-term solutions, such as stifling legislation.

*Nick FitzGerald, Consulting Editor*

# NEWS

## Taking Liberties

The major anti-virus companies are reacting in their own characteristically colourful ways to the emergence of PalmOS/LibertyCrack, a rather banal new Trojan for PalmOS. First prize for creative hype and the special 'let's confuse the hell out of our users' award must go to *Trend Micro*. Not only does *Trend's* headline announce 'the first virus' targeted at PalmOS, but in its headlong frenzy to claim first place with a quick fix in its software, *Trend* elevates this particular piece of malware to the improbable status of 'the Palm_Liberty.A Trojan virus'. Now that surely *is* a first! ▮

## Love Won

By now it is common knowledge that all charges against the man long suspected of writing the infamous LoveLetter virus have been dropped. At one point Onel de Guzman, a former student at the Filipino AMA Computer College, was charged with 'traditional' crimes such as theft and the violation of a law which normally covers credit card fraud while the authorities struggled to find legislation to fit his crime. While it is reported that he admitted that he may have released the virus 'by accident' – he refused to admit that he had created it.

Saddened as we are by the flagrant ineptitude of Filipino laws with regard to computer crimes – one hopes that the ruling passed hastily and too late by President Joseph Estrada in June will have its day in court – isn't it time to stop applying platitudes about horses and stable doors to our Far Eastern cousins and start checking the bolts a little closer to home? ▮

## EVAC-uation Proceedings

British businesses have taken the initiative in the wake of the LoveLetter and Stages epidemics by establishing an informal group known as the Enterprise Virus Alert Community (EVAC). Set up by a posse of ten corporate systems administrators, the outfit has been running for a couple of months and boasts 17 prominent members including several High Street banks.

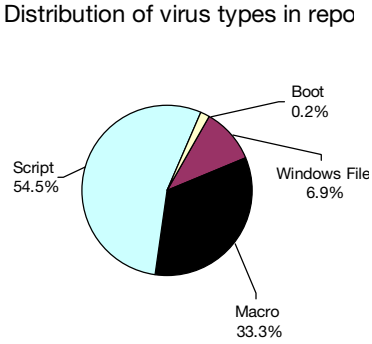
*Virus Bulletin* is encouraged to hear that the group is confident that it will not waste time on hoaxes and that this is not cyber vigilantism. EVAC expects businesses to take evasive action on learning of a new corporate email-borne threat and then wait for the fix from established AV companies to come through. It's a beautiful theory but cynics here at *VB* are sceptical about the notification methods EVAC employs. In the face of a serious threat, sys admins receive a short text message on their mobile phones – only a matter of time, perhaps? ▮

### Prevalence Table – July 2000

| Virus | Type | Incidents | Reports |
|---|---|---|---|
| Stages | Script | 387 | 30.0% |
| Kak | Script | 177 | 13.7% |
| LoveLetter | Script | 125 | 9.7% |
| Marker | Macro | 93 | 7.2% |
| Laroux | Macro | 69 | 5.4% |
| Win32/Ska | File | 61 | 4.7% |
| Win32/Pretty | File | 52 | 4.0% |
| Ethan | Macro | 35 | 2.7% |
| Thus | Macro | 27 | 2.1% |
| Tristate | Macro | 26 | 2.0% |
| Divi | Macro | 21 | 1.6% |
| Smack | Macro | 20 | 1.6% |
| AntiCMOS | Boot | 17 | 1.3% |
| Class | Macro | 16 | 1.2% |
| Myna | Macro | 12 | 0.9% |
| Yawn | Macro | 12 | 0.9% |
| VCX | Macro | 10 | 0.8% |
| Win32/Fix | File | 9 | 0.7% |
| Cap | Macro | 8 | 0.6% |
| Proverb | Macro | 8 | 0.6% |
| Melissa | Macro | 7 | 0.5% |
| Panther | Macro | 7 | 0.5% |
| Pica | Script | 7 | 0.5% |
| Total | | 1288 | 100% |

[1] The Prevalence Table includes a total of 82 reports across 35 further viruses. Readers are reminded that a complete listing is posted at http://www.virusbtn.com/Prevalence/.

In order to avoid a distortion of the figures, data for the 'self-reporting' W97M/ColdApe virus (totalling 574 reports in July) have been omitted from the table this month.



# LETTERS

## Dear Virus Bulletin

[*At VB2000 in Florida in September, two papers address the subject of viruses for* Linux. *Feel free to join in the debate! Ed.*]

### Show Me the Virus!

In *Computer Weekly*, 29 June 2000, a letter from *Sophos* claimed that viruses exist for *Linux*. I, the director of netproject, UK (www.netproject.com), have asked *Sophos* to demonstrate these viruses on a *Linux* computer that netproject supplies. *Sophos* has refused.

We have been working in the Unix area for over 20 years. During this time we have never encountered a Unix or *Linux* virus nor have heard of any organisation that has been infected by a Unix/*Linux* virus. We need to stop the fear, uncertainty and doubt that the anti-virus companies are trying to create around *Linux*.

I do not doubt that viruses can be written for any operating system. What is different about *Linux*, compared with *Windows*, is that there is no need for anti-virus software because controls exist to ensure that only authorised software runs on a correctly configured and administered *Linux* computer. These controls do not exist for *Windows*. At *netproject* we believe that *Linux* is pretty much bullet-proof and if *Sophos* are able to infect a well configured *Linux* box then it will uncover an implementation defect rather than a design flaw. Anti-virus software simply treats the symptoms and does not address the fundamental design weaknesses that allow viruses.

The challenge to *Sophos* is to send an email with attachments. These will be read and attachments opened. There will be no anti-virus software involved in this demonstration. The demonstration can take place in the laboratories of *Sophos*. I believe that this is a fair challenge and it is not one that *Microsoft* would be prepared to offer. Viruses are a fact of life with *Windows* because of the design defects and the complacent attitude *Microsoft* has to security.

So far *Sophos*, despite having claimed that viruses exist for *Linux*, has refused to demonstrate them on any *Linux* computer that *Sophos* has not configured. The response received from Graham Cluley at *Sophos* is: 'I don't have any response other than that which I have already given you. I'll give the same response to any journalists who might call me up. I don't have any more to say on the matter. I think it will be a waste of our mutual time if you email/phone Sophos on this matter again.'

*Eddie Bleasdale*
netproject
UK

### Practise the Theory

The letter *Sophos* wrote to *Computer Weekly* states the simple fact that there are a handful of Unix viruses in existence (note – not circulating) and that Unix users should not think they are invulnerable to malware attack.

*Sophos's* letter to *Computer Weekly* wasn't saying there was a huge threat from Unix viruses – just that they existed. We were correcting a previous correspondent who said they didn't and couldn't ever exist. The original academic research into viruses was done by Dr Fred Cohen on Unix systems long before there was a PC virus threat.

Some devotees of non-*Microsoft* operating systems believe viruses are impossible on their favoured operating systems just because there are very few, or they are rarely encountered. However, this is not because their operating systems are more secure. Viruses do not require security loopholes to operate, nor do they need to cause damage. All a virus needs to do is copy itself. And what usable operating system doesn't have that facility?

But the *Linux* devotees don't want to see this simple truth. And trying to get them to listen to the facts from respected experts in the security field is as easy as nailing blancmange to the ceiling. The irony is that there are operating systems which don't have any viruses for them; *Novell NetWare*, for instance. But no-one questions the importance of running anti-virus software on their *NetWare* servers (hint: it's not to stop the non-existent *NetWare* viruses, it's in case you use the server to store *Wintel* viruses).

*Sophos* is in the business of protecting computers, not infecting them. Therefore, we have no interest in responding to challenges to attack systems.

Maybe Mr Bleasdale should attend VB2000 and picket the talks being given about viruses and Unix there?

*Graham Cluley*
Sophos Plc
UK

### In Defence of the Enterprise

I'd like to comment on the August Book Review. Granted I am biased, I am one of the people who was mentioned in the 'Introduction' and helped check through the drafts of the chapters. While the book may have benefited by 'Professional Proof Readers', I think the reviewer totally missed what the target audience is. The target audience is the *Enterprise* support personnel. In a majority of larger enterprises, anti-virus is the job of either one individual, or a staff. In many cases, these individuals are not anti-virus experts, and this book is to help them in their quest for

knowledge. It gives them a set of what the author and his consultants felt were real-world issues that face enterprise level anti-virus support and deployment personnel.

The view of the reviewer that the author attacked *Virus Bulletin* is greatly mistaken. The author *did* point out the relationship between *Virus Bulletin* and *Sophos*, and he *did* point out what are felt by many to be shortcomings in current testing procedures by *all* current testing facilities.

Again, the target audience needs to be given information so they can make the best decision for their environment. The criteria outlined are a suggested criteria and are questions that the user can accept as is, detract from, or add to, and are based on several people's years of support to enterprise environments.

The attitude of the reviewer shows the classical breakdown in the Academia/Vendor Lab environment and field support personnel. Often those in the Academia/Labs consider Enterprise support as not part of the industry, and Enterprise/Independent Consultants/Supporters consider Lab and Academia out of touch with customer needs and desires. In no means is the book perfect, but especially after the mass mailers we've been assaulted with, more and more companies/enterprises are making anti-virus a dedicated responsibility, if they haven't before. I really feel the reviewer missed several major points of the book, and seemed to go out of his way to find things he felt
were wrong.

*Kenneth L Bechtel II*
Team Anti-Virus
USA

## Valuable but Élitist?

I agree in principle with Stuart Taylor's Comment 'Divided We Fall' (August issue) that organizations like REVS and *CARO* are a good idea. It is true that the current methods of sample distribution are too slow and as Stuart discussed, are reliant on key people being present day and night to send and receive urgent samples.

However, to ensure closer working practices throughout the AV industry, the whole process of gaining membership to bodies like *CARO* and REVS needs to be addressed. *MessageLabs* has been attempting to join REVS since it started, but we are still at least several months away from being accepted as a member.

This seems very strange to me – *MessageLabs* has a proven track record for being one of the first companies to come up with new virus samples. This is not surprising given that we scan mail at the ISP level, we invest heavily in heuristical detection methods and we house a massive knowledge base of virus characteristics – this all puts us very much at the forefront of the AV industry when it comes to being the first to detect new viruses. As an example, the post-LoveBug doom and gloom, which featured in the June issue of *VB*, could have been avoided and instead, been a story of triumph and success for the AV industry. Had *MessageLabs* been a member of REVS (and of course, if REVS had existed then) then the damage caused by the LoveBug to customers could have been minimal.

We intercepted our first sample of LoveLetter on 3 May 2000, 23:43 GMT. The first public signature was not released from the AV vendors until 09:00 GMT the following day. We had already protected all our customers by this time. If *MessageLabs* had been part of an AV body like REVS, then a lot more customers would have been saved.

So, in summary, I think organizations like REVS and *CARO* are key to the AV industry and more importantly, to our customers. However, serious consideration needs to be given to membership and overall awareness worldwide – we do not want to continue to promote an élitist entry structure to membership of these organizations.

*Alex Shipp*
MessageLabs
UK

## A Question of Motivation

The whole anti-virus industry upsets me. They charge for a product which I don't want to use, and in an ideal world I wouldn't need, much like a fire extinguisher (on-demand scanning). They also sell smoke detectors (on-access scanning) but what do you get? Where's the value? How is the common man to gauge the efficacy of one product over another? Who is the potential loser every time a new virus is created by some malicious party? The common man.

As I understand it, REVS is a mechanism for anti-virus companies sharing samples of new viruses via email. This confuses me. As I see it, an anti-virus company has two main assets – its body of knowledge about the subject (really a succinct way of saying the number of viruses it detects and the variety of media in which it detects them), and the service it gives its customers.

The former is easy to understand – the more it detects, the better the product. The latter is important for two reasons. Firstly, the computer industry, and the economy as a whole, is becoming service-orientated. Secondly, anti-virus software must be kept up to date. This means buying more products each and every month.

If many anti-virus companies are participating in REVS, does it mean that they detect all samples transmitted through the system? No, of course not. But it does mean that all anti-virus companies involved in REVS are starting from the same position in the number of viruses they are aware of. Surely the big multinational AV companies would want to differentiate themselves from one another as much as possible. Instead, REVS is simply a check-box which some IT Manager will want ticked.

*Anon*
UK

# VIRUS ANALYSIS

## All the World's a Stage

*Gabor Szappanos*
*VirusBuster, Hungary*

A well-known Argentine virus writer called Zulu created VBS/Stages, which appeared in the wild in June. A closer look at the contents of the worm (there are some advantages to the fact that worms use the same carrier) reveals that it was created in a Spanish version of *Windows*.

### Shell Scraps

This worm proves once again that whenever AV experts think that they know just about every possible executable file type, *Microsoft* comes in and proves them wrong. Until recently I had no idea that there are objects called 'shell scraps' that can encapsulate an executable program, document or script and release it on double-clicking.

A shell scrap is basically a special disk representation of a clipboard object created by the OleSaveToStream function. The object could be anything created by an OLE drag and drop-capable server application, for example a text selected in a Word Processor or a VBScript file packaged by the Object Packager, as in the case of Stages. When such an object is dragged to the desktop or an *Explorer* window (an intermediate container object) the shell scrap is created. When the user double-clicks on it later, the internal object is unpacked from the container, and gets passed to its registered host object – in the case of Stages this is the *Windows Scripting Host*.

### Worm Activation

The worm arrives in infected email messages with a randomly selected subject and body. The attachment contains LIFE_STAGES.TXT.SHS and the (usually) 39,936-byte long worm body. For reasons known only in Redmond, the SHS extension (along with .PIF and .LNK) always remains hidden even if the user selects to display all extensions. I guess 'all' has a different meaning in their dictionary. Consequently, at first sight the attachment always looks like an ordinary text file (except for its icon).

VBS/Stages.A consists of three major components: the shell scrap file (carrying a VBScript inside) responsible for installing and reinstalling the worm, and two additional VBS files. The worm is activated whenever the user opens the attachment, or double-clicks on the saved disk file. Upon activation, the worm displays a joke (in the application registered to open .TXT files – usually *Notepad*).

Then it creates several copies of itself and drops a couple of VBS scripts that are used for self-reconstruction in case the main copy of the worm is deleted. The main copy will be under the name LIFE_STAGES.TXT.SHS. The default icon for the .SHS files will be changed to the icon of plain .TXT files to make its presence less visible.

An additional copy named MSINFO16.TLB is placed in the *Windows* system directory along with the two scripts, SCANREG.VBS and VBASET.VLB. If on any available network drive a Windows\Start Menu\Programs\Startup directory is found, the worm will be copied there to infect LANs. Stages moves the program REGEDIT.EXE into the Recycle Bin, renaming it RECYCLED.VXD and then redirects the association of the .REG files to point to this location. This way, install scripts will work but the user will not be able to disable the autostart of the worm on startup.

The worm then copies itself to the bin under the name MSRCYCLD.DAT, and the two above-mentioned 'satellite' VBS scripts under the names RCYCLDBN.DAT and DBINDEX.VBS with respective lengths of 14,559 and 2,543 bytes, all with read-only, system and hidden attributes. Both satellite VBScripts are responsible for restoring the worm in case of corruption or incorrect disinfection. The worm uses a fixed-key encryption for the string constants in the text of all three VBS files with the encryption key being one of the characters in its own VBS code. Stages tries to locate the programs MIRC32.EXE or PIRCH98.EXE on each local drive. Two search mechanisms are used: *Microsoft Word* via automation if it is installed, or alternatively the FileSystemObject method. If any of the above clients are found, an appropriate INI file is dropped that sends out the worm on open IRC channels.

On each execution further worm copies are created with random names. These are combined by randomly selecting one of the following: IMPORTANT, INFO, REPORT, SECRET, UNKNOWN, appended with '-' or '_' and a random number between 0 and 999 (with a 1:3 chance the appendix is omitted) with file extension .TXT.SHS resulting in file names like SECRET_735.TXT.SHS. The files are placed in the root directory of each local or mapped network drive, and in the 'My Documents' (or *Dokumentumok* in Hungarian *Windows* versions) folders of the system drive. These files are created without any checking of presence; therefore, their number will increase with each execution of an .SHS file containing the worm.

The installer registers the first, longer script by creating the HKLM\Software\Microsoft\Windows\CurrentVersion\RunServices\ScanReg Registry key to point to the SCANREG.VBS script so it will be activated each time *Windows* is started up. On startup, this VBScript first restores the 'main worm copy' located on the Desktop. If it is not found, the script restores it from MSINFO16.TLB or from MSRCYCLD.DAT. In either case, the worm recognizes itself by the length: a 39,936-byte .SHS file will be

considered a valid worm copy. In the second step, the VBScript looks for the second VBScript. If it does not find it, it copies the main worm copy from the desktop to the Startup folder, in order to reinstall the worm properly.

The second VBS is registered to run on each ICQ client startup. It does the same as the first one: restores the main worm copy, then checks if the first VBS is present. If not, it copies the worm to the Startup folder for reinstallation.

The installation-reinstallation cycle of the worm is summarized as follows: when first executed, Stages copies the main copy of LIFE_STAGES.TXT.VBS to the desktop and additional backup copies all over, then registers the two 'satellite' VBScripts for auto-execution. These two scripts first delete the optional reinstall copy from the Startup folder, and then synchronize the copies of the worm from the backup copies. If no backup copy is found, LIFE_STAGES.TXT.SHS is copied to the Startup folder to reinstall the worm. The reinstall procedure will create and register the backup copies and the two VBS scripts, which in turn delete the install copy – and we are back where we started; the cycle goes on and on.

Most of the worm code uses numeric constants and the GetSpecialFolders function to reference the various *Windows* folders – except for the Startup folder, which is referenced by name. As a result, the reinstall feature of the worm will not work in nationalized versions of *Windows*, which use different names for the Startup folder.

## Propagation

The worm uses two, by now standard, propagation mechanisms. The first uses *Outlook* to send the worm out to the recipients found in the currently logged-in user's address book. Stages picks 100 random names from the address book (if it has less than 101 tems, it will use them all). The exact number is slightly below 100, depending on the number of address book entries. The reason for this is that the worm performs a check for multiple entries in the pick list, re-filling the list with random recipients, then re-runs the check and removes the occasional reappearing multiple entries. The resulting list will contain fewer than 100 elements.


The subject of messages is randomly assembled: it starts with either 'Fw:' or a blank space with equal probability. Then comes either 'Life stages' or 'Funny' or 'Joke', and finally, with 50% probability, 'text' is appended to the end. The body itself is also random: it contains (with 33% probability) the text 'The male and female stages of life', then with another 33% probability the text 'Bye' is appended. Interestingly, the mail body is either created as a text body or as a HTML body with equal chances. The worm itself is attached to the emails. To cover all signs of its presence, the outgoing messages are removed from the 'sent' items folder. Stages uses the Registry key HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\OSName to ensure that it is spread only once from a PC. If its value is 'Microsoft Windows' the *Outlook* propagation is skipped.

The second method of propagation is by using the *MIRC* or *PIRCH* client programs. This procedure has some protective measures: it will not be sent to 'friendly' IRC channels like nohack, backorifice, dmsetup and will not be sent out to people who use keywords during the conversation like virus, worm, stages, virii, worm, dccallow, stages, .SHS, trojan, spread, infect, unload or remote.

## Peculiarities

Here we have a worm on our hands which does nothing but copy itself. One would say that there should be absolutely no problem with detection – Stages is supposed to have the same carrier whenever it is found. Nothing could be further from the truth. I was shocked to notice that the activated copy of Stages has different lengths after execution – 38,912 bytes length in the Hungarian *Win98SE* or 40,149 in *Windows 2000*.

The response from the shell development team at *Microsoft* is that when you activate a shell scrap, it is reloaded and put back onto the clipboard (or whenever it is located on the disk) using the OLE functions (StgCreateStorage). They suspect that OLE is secretly updating fields like 'Last accessed by'.

Anyway, the result is that the worm will change its size and content after execution depending on the operating system and possibly depending on the user settings. As noted, Stages uses the file length to recognize itself. If an executed copy is re-executed before the synchronization at the next startup, the variants with new length will not be able to recognize themselves, which could result in endless reinfections. It is possible that the complicated reconstruction and reinstall procedure is motivated partly by the need to remove the changed worm copies.

## Conclusion

What else is there left to say after Melissa and LoveLetter? Do not, under any circumstances, open email attachments. In *Microsoft* operating systems only the Almighty knows what can be considered an executable file. So, be on the conservative side and consider everything as a potential virus carrier. This is the only way not to be proven wrong.

# OPINION

## Chopping Off the Tail

*Peter Morley*
*NAI, UK*

Creeping bloat, the disease first postulated by Ian Whalley in April 1999 (p.2), now, to a greater or lesser extent, threatens many anti-virus products. How? Well, it threatens to make them too big, or worse, too slow. The classic response to this situation is to sit back and say 'Hardware speed-up and operating system development have always coped in the past, so they probably always will.' Will they? Well, in the near future they might. Although I see nothing helpful in the latest operating systems (*Windows 2000*, *Windows ME*, and countless *Linux* and Unix versions), the GigaHertz processor is not far from becoming the standard, and that will help a lot. Unfortunately, it helps less in marketing to those parts of the world where the use of slightly older hardware is normal.

The GigaHertz processor will be followed by a fairly monstrous change – the introduction of 64-bit processing. This is already in the public eye viz *Intel's Itanium* and *AMD's* latest announcements about future designs of *Athlon*. As with all monstrous changes, take-up will depend on how easy the vendors make it for the public to change. Time will tell, but I am not optimistic about speed, and whether they like it or not, the anti-virus vendors will have to handle customers using *Pentiums*, and the present *Athlons*, for several years to come.

### The Spectators

It is perhaps worth noting, that some anti-virus vendors will not have a problem – those who have followed a policy of only detecting viruses on the WildList, plus a few 'talking-point' viruses, plus a few oddments like viruses generated by construction kits. Such vendors will happily detect less than 10,000 of the 55,000 or more viruses known to exist. Their customers may be happy too, particularly if they mask their inadequacy by bending over backwards to provide a high level of Customer Support.

The 'hide behind the WildList' approach leaves me cold, because I know what happens to vendors who take it. They get far too many new viruses direct from the wild, and they have to process them with the customer ringing up, asking 'How are you getting on?'. I believe that the cost of handling the Support far outweighs the cost of processing the viruses properly in the first place.

### Another Solution

There is another solution, which at least postpones the creeping bloat crisis, even if fails to solve it – strive to make your detection and repair capability as generic as possible. The *NAI* Virus Lab has been doing this consistently for several years. The effect of this policy is that we now detect and repair correctly over 50% of the viruses we receive, which we have never seen before.

Customers love it. They do not have to ring us up and hassle us for an answer, so they do not. We love it too, because we do not have to do much work on the viruses we already detect and repair correctly. There are two downsides to this policy:

i) The virus count becomes a gross underestimate, because there are so many viruses we do not get, because we already handle them.

ii) The virus library we swap monthly with nine other vendors will gradually become less comprehensive than it used to be.

We accept both these consequences and are still going generic. However, the policy does not solve the creeping bloat problem, so we must look for another solution.

### The Inevitable

There is only one other solution – remove the detection of some of the rubbish which we already handle, including some of the viruses which are long since dead. Easy, isn't it? Well, no it is not. And I believe our strong competitors also find it difficult, judging from their lack of progress in this area.

Let me tell you a story. Back in '87 there was a major field outbreak of a virus called Cascade. It is a relatively well-behaved and fairly harmless virus, but it made its presence known by causing the screen contents to 'cascade' to the bottom of the screen. Alan Solomon, who in those days processed all our viruses, had no difficulty in detecting and repairing it, and his product (*Findvirus*) received a spectacular boost.

Six years later, in '93, when the total number of viruses we detected was still less than 3,500, Alan formed the view that Cascade was dead, based on the fact that we had had no calls about it for over a year. Aware of creeping bloat (although he did not use the term), he decided to do an experiment. He removed all disinfection of Cascade, to see what happened. The result was that we were inundated with calls and complaints, and we had to put it back. And, seven years later, I can assure you that I would not take it out again, even now.

I think there are three conclusions to draw from Alan's experiment.

i) Some of our big customers keep a small Virus Collection containing every virus which has ever infected them. They never prune it, and they

occasionally test against it. If you fail to detect something in it, however old, and however dead, you risk getting a black mark!

ii) The fact that you get no calls about a virus may not mean it is dead. It may just mean that you handle it so well that customers do not need to call. These customers get rather miffed if you fail to handle a later outbreak. So …

iii) Do not *ever* remove disinfection of a 'Field' virus, or a 'talking-point' virus, or you risk making someone unhappy, then you may become unhappy.

**Homing In**

So, what disinfection can we remove? I will suggest a few categories. If, wearing your user hat, you strongly disagree, then I am sure *VB's* Editor would love to hear from you. And I can assure you, she *will* let me know!

1. *Non-virus generators, and the non-viruses generated*

The only reason we detect these at all, is that they appear in Review Suites, against which we are evaluated. Although we detect them, we do not count them, so there is no numerical implication. My instinct is just to take them out, but I have a feeling that I should talk to a few reviewers first, and seek their agreement to exclude them, on the basis that they are not viruses.

I suppose some would argue that the generators are Trojans, but even if they are, they are the type of Trojan you have to run deliberately. And even then they will do no harm, apart from filling up some disk space with junk. Incidentally, *Virus Bulletin* is one of the few reviewers I would *not* need to talk to. *VB* does not include this type of rubbish in their Review Suites.

2. *Virus Construction Kits*

We currently detect viruses made from these kits, but we do not count them. The argument for detecting them was given to me several years ago, by the IT manager of a large customer. He said 'If one of my machines has a Virus Construction Kit on it, I should like to know about it!' – I could not argue then, and we have detected them since.

However, I now hold the view we could remove detection of some of the old ones, particularly those which generate OFFVs (Old Fashioned File Viruses) which no one cares about any more. What do you think?

3. *One-generation wonders*

These are files which will not run, because they have been misinfected, because someone has run a dropper which misbehaves. We usually detect them, and repair them, because the repair is easy to do.

However, customers needing this facility are thin on the ground, and I suspect we could remove the facility, and see who screams. The screamers will be those who have to replace their damaged files with the originals. If they run the damaged files, a reboot is necessary, but other damage is unlikely.

4. *Real viruses which ask whether they should infect*

Before you say 'Of course. Take them out!', please let me tell you why they are in. They are in when the same detection and repair will still work, if the author (or a hacker) modifies the code so they no longer ask.

Again, my view is that we could remove detection of the old ones.

5. *A special case – Trivial.18*

For a short time, in 1996, this 18-byte, overwriting virus held the distinction of being the shortest virus we had ever seen. It was a willing infector, too, but only if you ran an already infected file, and gave it the name of the file to infect, as a parameter!

I have already removed detection of this one, and await the holocaust.

6. *Appenders*

These occur in two ways:

a) A virus appends itself, and then fails to change a jump, so the file will run normally and the virus will never be executed.

b) As a result of a half-cocked repair (not by us!), where the initial jump is fixed, but the appended virus is not chopped off. Again, the file will run, and the virus is harmless, because it will never run.

I tend to remove detection of these cases whenever I think I can get away with it. The snag is that competitors who still use 'Grunt Scanners' will still find the impotent virus code, and questions result.

7. *Boot Sector virus droppers, where the virus has never been in the wild*

My instinct with these is to leave them a couple more years, and then to take them out, together with detection of the viruses they drop.

**Summary**

I am up to category number seven and I have still ducked the question of removing detection of real viruses which no one cares about any more. There are a lot of them and I am in a strong position to know which ones members of the public do not care about.

However, the anti-virus using public has clearly demonstrated that it *does* care about bad review detection rates, so before I can embark on the pruning, I need reviewers' agreement to remove some specifically agreed viruses from their review suites.

# INSIGHT

## Chien Reaction

*Eric Chien*
*Symantec, Netherlands*

Somewhere between the mountains of Costa Rica and Southern California, everything changed. The year was 1995 and it was my last year at UCLA. I was studying Molecular Genetics and Electrical Engineering and continually found myself trying to explain how I thought the two were correlated. I had just returned from hiking the jungles of Central America and was looking forward to graduating with a Bachelor of Sciences degree.

After graduation, I was planning on joining the Peace Corps, which involves performing two years of hard labour for a third world country and being reimbursed a hefty sum of US $2,000 at the end of the term. For some reason, I thought this is what my life's calling entailed – digging ditches and swatting malaria-filled mosquitoes, for free.

Soon, all my plans would completely change and it would all start with a virus. Well, actually it was a fungus. *Histoplasmosis* is not recognised by the *Microsoft Word* spell checker, or most doctors. Luckily, after three weeks of non-stop fevers, the diagnosis was in. I was to live. However, the time away from school would change everything.

After an extra year in school, I dumped the Peace Corps idea and decided to flip 180 degrees and step into Corporate America with the intent eventually to return to university and strive for a PhD in Genetics. After choosing to help develop 'boring utilities' (in the words of friends), rather than the latest Playstation game, I found myself in a Dilbert-size cube at *Symantec* in Santa Monica, California. Three years later, I am now in Leiden, Netherlands and I could not be happier.

So, in a haphazard way, a malicious infector did influence my desire to work for *Symantec AntiVirus Research Center* (*SARC*), but to most people's surprise, it was a biological one and not a digital one. In fact, I personally have never contracted a computer virus. The secret is 'Safe Computing Practices' which I find myself preaching to as many people as will listen. Do not be surprised if you find me at Speaker's Corner in Hyde Park screaming about the evil email attachment.

While not in front of a computer or at Hyde Park, I am usually in a plane, train, or automobile and even then I am usually in front of a computer. I have fielded computer virus enquires from the beaches of India to the slopes of the Alps, even with a snowboard strapped to my feet. Computer viruses do not go on holiday so unfortunately sometimes, neither do I.

So, in my pseudo-free time I enjoy being encapsulated in a steel tube that is either hurtling up to 200kmph on two thin metal strips or defying gravity at 30,000 ft. Granted sometimes it is a steel box that is only achieving 120kmph, but one must avoid similar boxes rolling at the same speed. Sometimes this is not an easy task. For this masochistic nature, I blame my parents.

Born on October 25, 1973 at 1:48 am, in Ann Arbor, Michigan in the United States, before even starting school, I found myself in Europe and eventually Libya, Africa. Now, I like to pretend my father was there on secret missions for the *CIA*, but in fact he was building pipelines or some other boring structures. However, the travelling I did when I was young, stuck. I set a goal to visit all continents before I turned 28, and I should be completed by this year including Antarctica. I have a feeling my mobile phone will not work in Antarctica. I have travelled to about forty different countries and hope to see many more while not listening to the daily hum of computer fans.

While I find it impossible to describe 'A Day in the Life of SARC', there is one commonality – each day is different. My official title is meaningless (Program Manager) and the one on my business card (Chief Researcher – EMEA) is a bit arrogant, but it sounds nice. Currently, I am researching malicious code for *EPOC32* (*Psion*), *PalmOS*, and also *WindowsCE* operating systems including possible solutions. Unfortunately, there are already Joke programs (some classify them as Trojans) for these platforms and it is just a matter of time before someone writes something more malicious.

In the last three years, I have worked on many projects including disassembling countless viruses, developing the current automation system utilised by 'Scan And Deliver', *Symantec's* automated method of allowing people to submit suspect files, and even naming the ExploreZip worm. Today, I speak to the press, research new threats, develop new technologies, and still do virus analysis (like, for instance, VBS/LoveLetter), but I do not plan on researching viruses for life. I enjoy the fact that each day brings new challenges and that technology evolves so quickly; however, I will probably eventually be in the jungles of South America without an email address and fighting biological viruses. For now, 'A Day in the Life of SARC' suits me just fine although the industry may be here longer than I may be alive.

Of course, in the strictest sense, the whole anti-virus industry is already gone. Today, we fight malicious software and not just viruses and as each day passes the scope appears wider and wider. Every researcher has to be aware of not only the latest virus, but also the latest security exploits and possible vulnerabilities that may be harnessed

VIRUS BULLETIN **©2000 Virus Bulletin Ltd, The Pentagon, Abingdon, Oxfordshire, OX14 3YP, England. Tel +44 1235 555139. / 2000/$0.00+2.50**

to distribute malicious code. The problem will never go away in its entirety. It just changes face – mainframe worms, DOS file viruses, boot viruses, macro viruses, network aware malware.

The problem will continue to change face as new technologies emerge – WAP, smart phones, communicators, and other digital devices. For the future of viruses, one must look no further than the future of technology in general. Any platform that can execute code, can execute malicious code. Clearly, the solution of known threat scanning must evolve and some of the oldest methods, but most effective (e.g. behaviour blocking, integrity checking) are returning in evolved states.

Unfortunately, there is an unlimited future of new technologies for the virus writer to exploit. As long as we do not revert to carvings in stone, the arms race between malware writer and security industry will remain. I am certain the industry will continue to respond (proactively and reactively) to threats with novel solutions, if not out of altruistic desire, then out of the desire for further revenue. The computer is clearly here to stay and appears to be the foundation of the future of industry in general, and for that the security industry will remain. In that sense, there is no grand solution to the problem, but the ability to minimise the effects.

Of course, minimising the problem requires everything from technology to politics. Laws should be enforced and in many countries have yet to be created. Software developers should realise the risks especially in a connected environment. The functionality versus security balance should shift towards security, but needs to be driven by consumer demand.

Security vendors and consumers need to realise the necessity of many solutions (intrusion detection, firewall, content scanning, behaviour blocking, backup software, etc.) similarly to how biologically people fight AIDS with a cocktail of drugs. Unfortunately, there is no silver bullet. The closest thing to a silver bullet is education.

When on-line, people often refer to 'in real life' in both chat rooms and emails. That the nature of our on-line presence is somehow virtual obviously affects how users interpret the threats from malicious code. The understanding that millions of computers are now interconnected within seconds from one another and are running in the 'real life' world has not quite hit. After all, the simple understanding of questioning unsolicited email attachments could have stopped W97M/Melissa and VBS/LoveLetter.

But people do learn. Unfortunately, people often learn slowly and new people are getting on-line everyday – even cats apparently have email. So, in addition to safe computing practices, general attitudes must change. The notion of 'I don't care what happens to my computer, it is just a game anyway' is no longer valid in the interconnected world. While the user may not care what happens to their data,

they may care when they cannot cast their bid for the White House on *eBay* because their computer helped in a denial of service attack. And surely, others will care if you cause them to be unable to send email because you have graciously filled their email box with a thousand love letters.

The question of how to educate so many users becomes problematic. I attempt to address safe computing issues in every press interview and we have even visited schools to try to educate students. Pessimistically, the problem of user education will probably only be resolved when each individual learns for himself. When someone does lose valuable data or experience downtime, only then will they realise the impact of a simple double-click.

A friend once told me we should simply let malware run wild. His point was that the Internet and computers were inherently unreliable anyway. Computers tend to crash and email goes down all the time for other non-malicious reasons and we should trust neither computers nor the data held within to be reliable .

However, by ensuring their reliability and security we can create a foundation for unforeseen future industries. E-commerce and chatting on-line are only the beginning of technologies that require a secure computing infrastructure.

I am happy to be part of the industry that aids in making computers secure and hope to continue to do so. While contracting a fungus was not the best thing, it did lead me to a way I can help people and still make more than US$2,000 every two years.

# FEATURE 1

## Tools of the DDoS Trade

*Aleksander Czarnowski*
*Avet, Poland*

Every once in a while we see a 'new technology' emerging which turns out to be nothing more than a bunch of old ideas in new packaging. This is the case with Distributed Denial of Service tools.

Like with viruses, the first versions of these tools were relatively simple – we had to wait a few months to see something more advanced, like TFN2000. Due to the widely available source codes of these tools, we can assume the existence of many 'private' versions which have never been used in the wild. This fact also made it easy to port tools like trinoo to the Win32 platform.

### The Mechanism

Denial of Service (DoS) attacks are based on a very simple assumption that every system has limited resources. On the other hand, we tend to run more and more complicated network applications. More complicated applications usually mean more bugs and design/implementation flaws which can be exploited by an attacker.

It is worth mentioning that although it is possible to perform a DoS attack as a local user, in most cases such attacks are performed remotely. More often than not there is no point in crashing your machine locally while you are logged on to it.

### DDoS Evolution

The first DoS tools were simple programs that exploited the misconfiguration of network services or bugs in the network application or TCP/IP stack. The next phase of their evolution was to create a shell from which an attacker could run few different DoS attacks. This method made it easier to perform an attack, but an attacker was still required to compile all the DoS attack tools.

Then came another generation of DoS tool – Targa. Targa was a compilation of a few different DoS attack tools integrated into one application. This made it easier to compile and run and to perform an attack. At this point it appeared that nothing more could be done to such tools except perhaps adding other types of attack and integrating them into one piece of code.

However, evolution must go on. The next logical step was the use of a distributed model to perform attacks. Again, this is nothing new. Distributed models have already been used in many security applications like intrusion detection systems or automatic virus analysis systems.

### The Advantages of Distributed Models

Why would anyone want to use a distributed model to crash a machine remotely or to render the network connection unusable? The simple answer is that nowadays an attacker can bring down almost any network connection. Let me go back a little. To make a machine unusable remotely one has two choices; exploit some bugs in the configuration on the network application or flood the connection so that the machine or connection will exceed its limit.

The first choice is often used as it is very easy to find an exploit for a chosen platform. However, it can happen that there are no known bugs or that the machine is configured correctly. Sometimes there is quite simply no exploit to use against the machine. Even if there *is* an exploit, an attacker needs to identify the system platform that the machine is running. Not every exploit will work on every version of software. Some attacks can be blocked on an external router so it will never reach any machine in the Demilitarized Zone (DMZ) or LAN.

The second choice will always work: you only need to generate a stream of packets. If you can generate a number of packets greater than the target machine can handle, you have just performed a successful DoS attack. This can be seen sometimes with sites that have a high volume of traffic. They can be slow, or sometimes it is not possible to get a connection with the site. From an attacker's point of view the only challenge is the amount of network traffic that he must generate in order to bring the site down. Even in the case of middle-sized e-commerce sites it is not possible to flood the connection or external routers using a modem connection. More importantly, by using a distributed model, an attacker can remain unknown to the victims.

It is extremely hard to trace the origin of an attack but it is possible with the use of spoofed source IP addresses that are inserted into flooding packets. IPv4 allows such tricks. So, without the help of ISPs and telecommunications companies it is almost impossible to trace the real route of incoming packets. Since the attacker does not need to communicate with the victim machine, he also does not care about the packets received. An attacker can certainly spoof IP addresses, but from time to time he will need to use a real one.

### Construction of DDoS Networks

To perform DDoS attacks an attacker needs to build a DDoS network. Such networks are built from three different types of component. The first component is an 'attacker'. Each network needs only one attacker. The attacker can then communicate with a 'handler' and an 'agent'. The attacker machine can *only* communicate with the agent through the handler – there is never any direct

communication between attacker and agent. This is significant because it means that an attacker does not leave any information about his identity on an agent's machine.

First, the attacker finds a handler machine. It must be sufficiently vulnerable that he can illegally achieve root privileges. Then the handler application can be copied and compiled. The next step is to find an agent machine – again this means breaking into it and installing the appropriate software. Once the attacker has gained root privileges he can install rootkits to hide his presence on the system. He can also install a sniffer to monitor network traffic and to move further into the network.

It might look very time- and energy-consuming, but in fact almost every part of this process can be automated. Finding handler and agent machines can be achieved by scanning a wide range of IP addresses. It is time-consuming but it does not need any human interaction. Then a list of vulnerable hosts (one can look for vulnerable versions of BIND or *SendMail* for example – identification of those is relatively simple and fast) must be generated.

The next step is to exploit a given vulnerability. With Unix, all this, plus installation and compilation on target machines can be performed using scripts. Almost the same functionality can be achieved under *Windows NT*.

### Ready to Attack

When the network is set up, the attack can be launched. By issuing the appropriate command the attacker tells his handler to start the attack. The list of handlers is kept on the attacker's machine. Subsequently, the handler sends the same command to its agents. The list of agents is kept on the handler machine. Agents are always the ones who actually perform the attack. It is the role of an agent to generate a stream of packets with spoofed IP addresses.

Handlers do not communicate with each other, they only communicate with the attacker and their agents. Thanks to such a set-up, even if we were to find and isolate one particular agent we would still not know anything about the associated handlers.

### Defending Your Network

A stream of packets will come from different places and all of the packets are sent to one IP address. Usually, the stream kills the external router long before it can reach a firewall. In such cases the firewall mechanism in place to protect from the attack is rendered useless.

If the stream can reach the firewall it is possible to defend the network by using techniques like dynamic host blocking, but this will only work in the case of several packets with the same source IP address. If every packet is sent with a random IP source address this technique will not work. Even if dynamic host blocking or any other technique is implemented on the firewall, it is still possible that all the firewall's resources will be consumed, again rendering it unusable. This problem is easy to solve: simply limit the number of connections being serviced.

Another method of defence is based on routing. The trick is to switch between two networks. One will be flooded by packets and the other one can be used normally. The drawback to this method is the need of ISP support.

If we can manage to disable the handlers, the agents will become useless. So, the first task is to isolate and remove the handlers. There are already tools like ZombieZapper that can send a signal to the DDoS network remotely to stop an attack. The current version of ZombieZapper works against trinoo (and its Win32 version), TFN, Stacheldraht and Shaft. Unfortunately, it will never work against TFN2K due to the way the TFN2K network communicates with each particular component.

### Host Level Detection

While it is hard to detect such attacks at network level, it is easy to do it at host level. Furthermore, anti-virus software is ideal for the job due to the use of powerful and advanced scanning engines and wide infrastructure. Most scanners should not encounter problems even if DDoS tools employ the stealth techniques used currently by viruses or worms, or if/when polymorphic engines are built into them.

The recent development of worms and other malware for the Unix platform could pose a real threat. It is simple to hide a potential DDoS attack in a large chunk of virus code. If the virus code is analysed by some automatic analysis system its hidden 'weapon' will probably be missed. Even if a human took on the analysis, it is still possible that it would be missed.

What is even worse is the fact that viral worms can spread very quickly. Worms and viruses can be used as a very powerful method of building huge DDoS networks. The best case scenario would see a firewall filtering out all the control requests for DDoS network components like handlers or agents, rendering them unusable.

What is more important is that most machines are still running under *Windows 9x* which is very unstable and not as powerful as other Unix-based systems. While it is true to say that *Windows 9x* is insecure by default, its suitability for building DDoS network is somewhat limited.

### Conclusion

We still do not have one proper method of dealing with DDoS attacks, and what we have seen up to now might not be the end of it. The danger from Unix malware is growing. More and more hacking tools are being ported to Win32. In too many environments a lack of security policy or incident response plan is the norm. Security in educational environments is an almost impossible task and computers will always have limited resources allocated to them.

# FEATURE 2

## Greece is the Word

*Josmaarten Swinkels*
*Inter Engineering, Greece*

The picture that most people get when they think about Greece is one of sun, sea and vacations. For those who have not lived here or at least done a business trip, it might not be clear what those Greeks do who are not involved in tourism and what the level of automation in this lovely vacationland might be. Let alone the virus situation.

### Business in Greece

The major part of the country's national product comes from agriculture, tourism and sea transport. In agriculture, understandably, little or no computer use is seen but the other two categories are very well equipped and heavily dependent on their automation systems. Furthermore, of course, all kinds of other business can be found here. Until recently, few really large corporations were evident, but during the last few years this is rapidly changing with mergers and takeovers, accompanied by significant activity on the Greek stock market.

Today in Greece every business with a chance of survival is computerized with up-to-date equipment with all the usual vulnerabilities involved. Most companies are small or medium-sized. Their interest concerning their IT system is that it improves the effectiveness of their day-to-day business. Few really care about the security of their IT system and even fewer are willing to invest in it. The situation in the larger companies is different. They have systems administrators with a budget large enough also to cover preliminary security measures, and investments are being made. What I observe, however, is that very few companies have a security policy. Furthermore, I am pretty sure that the majority of the people involved are not really aware of the vulnerabilities and do not take them seriously enough. There is quite a wide field as far as security-awareness training goes!

### History

The undersigned has lived in Greece since 1989, and this is also pretty much the time at which the first incidents with computer viruses occurred. [*Pure coincidence, I take it? Ed.*] The first ones I can remember having spotted in the wild over here were Stoned, Ping-Pong and Dir-II. It is hard to get a picture of the volume of virus incidents because a lot of people (including companies) were and still are making illegal use of software and will not report it when hit by a virus. As far as I can comment, the evolution of virus appearances in Greece is about the same as in the rest of Europe. The availability of the Internet may have come a little later but today we can say with certainty that any virus which goes around will also hit some users in Greece.

### Virus Writing

Yes, there are viruses which originate in Greece. The ones of which I am aware are:

- Angus – a *Word* macro virus which encrypts documents on the 23 October and first appeared in 1997.
- Imposter – a *Word* macro virus with several variants. This one does not have a dangerous payload and first appeared in 1997. Angus and Imposter were probably written by the same author – NAENBGOURSG (meaningless, even in Greek).
- Armagedon infects COM files and tries to call a specific phone number in Crete through the modem. It first appeared probably around 1992.
- Athens, also known as Trojector, infects COM and EXE files. Most infected files end up corrupted. Within the file the Greek surname 'Koufidis' can be read. First appeared in 1992. Armagedon and Athens were probably written by the same author because inside the code of the Athens virus, the text 'Armagedon Utilities' can be found.
- Karnavali is a multi-partite virus which infects both boot sectors and EXE files. After having performed 60 infections it manifests sound and video effects: hearts are displayed on the screen together with the message

```
A lovely heart fell from the sky !!!
KARNAVALI OF PATRAS !!!
*** PATRAS H/Y ***
```

Karnavali was most likely written by someone in the Computer Science department of the University of Patras, the city of which also happens to celebrate a large carnival feast (recommended!) annually. I do not recall the exact date but the virus must first have appeared around 1996.

Of these five viruses 'only' Imposter, Athens and Karnavali were widespread. The Athens virus in particular was very effective. From the number of incidents we are aware of, my conclusion is that almost every computer user in Greece must have had an infection of the Athens virus. Lately it has been awfully quiet amongst the local Virus writing community. Its members are probably occupied with the stock market, have joined the army or are busy making a career. Undoubtedly, however, there will be new ones!

### The Anti-Virus Business

I can most probably call myself a pioneer in the fight against computer viruses in Greece. When I started with this professionally back in 1992 I was considered a

madman by everyone else in the IT business over here. It was clear (at least to me) that viruses signified an increasing problem without end, but at that time most of the software used was illegal. And why would anyone pay for anti-virus software or services if they would not even legally buy their business software?

And so for years we were the only company offering professional anti-virus services and solutions. I must say that today is not much different. Computer users have become much more aware of the necessity for AV software and many of them are willing to pay for it. Most distributors of AV software in Greece, however, have a 'box-moving' mentality. They sell AV software *because* people order it and *when* people order it. Most of them do not offer any technical support for it and certainly do not have any knowledge about viruses. Pretty often we find ourselves answering support calls from competitors' users.

From the above we can conclude that the average opinion of the computer industry here in Greece is that security is not important enough to put resources into. And there is proof of this! In March of this year the VBS/Links worm was unintentionally spread by a sales-employee of one of the largest IT distributors here in Greece. Within a day numerous large companies, educational institutes, IT dealers and press offices were infected. The person in question, after having being informed of his achievement, only sent an email out with the message 'sorry, we hope it will not happen again'. No guidelines of how to act after an infection, no acceptance of responsibility, nothing. It has to be mentioned that this distributor also provides one of the major anti-virus products on the market.

### The Users

Greeks are very spontaneous people. They also do not worry too much about anything until the time comes when they really need to. Of course, this mentality is also reflected in their attitude as computer users. If no security problems occur then people tend to forget about them (stop making backups, remove the annoying AV software etc). That is the reason why worms especially, like Melissa and LoveLetter, are a guaranteed 'success', especially in large companies and organizations. When under attack, solutions are sought anxiously and at any cost. After the storm has vanished, the whole incident is soon forgotten and the habitual negligence returns. As mentioned before, a lot of security-awareness has still to be engendered!

### The Media

For years the only Greek media mentioning computer viruses were the IT magazines. Once in a while a superficial article about viruses would be printed with nice pictures of bugs or other insects. The worst nightmares were the articles where IT magazines attempted to do comparative tests between AV products, going to great lengths criticizing the user interface without a word about detection rates.

Over the last few years virus incidents have become more spectacular and widespread – a reason for the rest of the media (other magazines, newspapers, television news) to exploit the phenomena. The first virus I can recall having been mentioned on the news, was Michelangelo. Naturally, as in every other country, a great fuss was made about the Melissa and LoveLetter worms. I myself was interviewed on TV during the Melissa incident and am regularly interviewed by reporters of national newspapers. All of this shows that the media are really after computer-security incidents. From one viewpoint this is good because it creates security-awareness. On the other hand, the media tend to exaggerate and provide incomplete information, provoking technophobia amongst many people.

### Law and Government

At this moment there is no specific legislation against creating computer viruses in Greece. I expect the bureaucratic system in this country will impose quite a delay before all people involved understand what they are talking about and create the relevant laws. To my knowledge, so far no virus writer has been caught here and the first court case sentencing a computer-criminal is yet to come (I pity the poor judge who will do the honours of conducting the inaugural one!).

As far as its own protection is concerned, theoretically the government is doing quite well. For years now every one of its public tenders includes anti-virus software. The question is, however, whether or not this software is being used and maintained correctly after its initial installation.

With the Olympic games ahead of us for the year 2004, the Greek government will also have the challenge of protecting the, as expected, impressive IT system running this very important event. Since attacks specifically targeting this system are more than probable, this is going to be an interesting project!

### The Future

I am sticking with the same statement that I made years ago. Computer viruses will be a persistent and even growing problem. New species and new infection channels will appear in the near future.

In Greece more companies will attempt to protect themselves better by investing in AV software and other security products. People's mentality, however, cannot be changed from one day to another. It will take a long time and a lot of evangelizing before an acceptable level of security-awareness is reached. New and unknown malware in particular will have a good chance of success because of people's curiosity and negligence. With anxiety and respect I await the impact of the first WAP virus in this country where everyone has a mobile phone – which fits ideally with the thinking pattern: it enables you to deal with situations when they occur, without having to worry about them beforehand …

# FEATURE 3

## Indian Cyber Laws

*Nadir Karanjia*
*N&N Systems and Software, India*

To date, the IT industry's focus has been on competitive pricing, technical savvy, knowledge trafficking etc, and while that still consumes considerable 'mind-width' of Indian IT gurus, of late there has been equal interest in securing this large, potentially multi-billion dollar industry. In order to be taken seriously in the long term, the base infrastructure which covers the fundamental requirements of cultivating a secure business environment need to be addressed. This means rules, laws and regulations which lend predictability, operational integrity and a degree of continuity to assist players in planning several steps ahead for the future within a secure legal framework.

It is very encouraging to see that our technology kings, lawmakers and, uncharacteristically, rabble-rousing politicians have all banded together to supervise a genesis of sorts, one that signals to the world that India is ready to usher in a new era in the age of digital economy.

The Indian IT Bill sets a framework for an e-commerce regime, consisting of 93 sections divided into 13 chapters. It includes four schedules which lay down the relative amendments sought to be made to the Indian Penal Code, the 1872 Indian Evidence Act, the 1891 Banker's Evidence Act, and the 1934 Reserve Bank of India Act. The 13 chapters deal with various subjects including digital signatures, electronic governance, acknowledgment and dispatch of electronic records, penalties and adjudication, the cyber regulations appellate tribunal, network service providers limitation of liability etc.

The IT Bill 2000, among other things, covers several aspects of digital communication, authentication, security and transaction integrity. Some of the notable highlights show how the Bill:

- provides for a legal framework so that information is not denied legal effect, validity or enforceability solely on the grounds that it is in the form of an electronic record.
- aims to facilitate electronic trade and commerce, eliminate barriers to electronic commerce resulting from uncertainties over writing and signature requirements and promotes the development of legal and business infrastructure necessary to implement electronic commerce.
- provides for appointment of certification authorities for the purpose of licensing, certifying, monitoring and appointment of the Controller to oversee the activities and regulation of Certifying Authorities.
- includes penalties in the form of compensation for computer crimes, such as unauthorized access to computer networks and computer databases, computer virus damage, disruption of computer services, copying of software. Offences such as tampering with the computer source documents, electronic forgery and other types of computer crimes have also been included in this Bill.
- seeks to empower government departments to accept filing, creating and retention of official documents in digital format. Similarly, unless otherwise agreed, even a contract represented in electronic form will be accepted.

It has also been recommended by a review committee that the fine for creating and causing damage by computer viruses be raised from the earlier amount of 1 million rupees (US$23–25,000) to 10 million (US$230–250,000) which is a hefty deterrent to would be 'ILoveYou' types. Then again, who says virus writers think of consequences; and what can be done to virus writers outside India, where most of the viruses originate in the first place?

Section 73a and 73b of the Bill, which required cyber cafés to keep records of users and logs of sites visited, has been rejected. This sweeping section also included the liability of anyone building and owning a Web site within India to register and disclose all details thereof to the controller or authority in charge. This was in addition to the mandatory registration of domain names required to have a Web site in the first place. This was a ridiculous suggestion, as the administrative costs alone would shut a cyber café down.

Also under this section, punishments included one year in the cooler as a guest of the state and a fine of up to 500,000 rupees (US$12,000). Quite creative, but as rightfully pointed out by *NASSCOM*, what would happen to a 10 year-old who forgot to register his home page?

Clause 79 permits a police officer of the rank of Deputy Superintendent or above to enter any public place, search and arrest without warrant any person suspected of having committed or of being about to commit any offence under the Act. Such a law is already part of the Indian Penal Code in general. Dropping this clause would permit any officer, even of the rank below Inspector, to wield tremendous power and the potential for megalomania looms supreme. Looks like this one we are stuck with. However, in keeping with the visionary trend, the recommendation has been tabled that a special task force be created to educate the police about IT and its surrounding issues. This would amount to empowerment at the enforcement level with a degree of *education* – critical to avoid untoward disruptions and the misuse of power.

The above is a sure step in the right direction. With this piece of legislation and the will to pass it and make it an Act, the Government of India and the country's IT industry join a small group of *élite* nations that have cyber laws in place. The Act itself is a gesture of confidence in India's IT industry, and a re-affirmation of a national commitment to become a digital nation.

# COMPARATIVE REVIEW

## Slipping through the NetWare

*Matt Ham*

In the last Comparative of *NetWare* products (July 1999) it was noted that although *NetWare 5* was the current version of that operating system, the tests were performed on an older version, since few products had active support for the new features of version 5. Time rolls on and not only would it be expected that these features might by now be supported to a greater extent, but also *VB* could be considered to be living in the past if *NetWare 4.x* were to be used again.

These reasons were very nearly ignored at the sight of *NetWare 5.1's* 240 MB Service Pack waiting to be installed in all its vast glory. The trials and tribulations of *NetWare* installation duly followed, though with these being familiar to all those who have had contact with *Novell's* products, the exact details can be glossed over. Of the products submitted for testing, two – the *RAV* beta and *VBuster VBShield* – proved unable to operate on the *VB NetWare 5.1* server. The former proved a casualty beyond help and will probably be featured in a standalone review soon. *VBuster* rallied eventually, and is included in the proceedings.

### The Test-Sets

So, the operating system is all new, what about the test-sets? Detection tests were performed on a *VB* test-set aligned to the July WildList with, due to a reviewer holiday, no non-WildList additions to the other sections. Any reader who has not spent the last few months on the moon will realise that the .VBS extension is the big new appearance in the WildList since *NetWare's* last testing at *VB*, and the numbers of such malware in the WildList have soared since the last (*Windows 98*) Comparative.

The viruses and/or worms in question have also introduced into the *VB* test-set a number of dual-extensioned samples as well as the now notorious .SHS extension. Since the extensions included for default scanning have often been a bugbear for *NetWare* products, will these new additions cause upsets in the VB 100% awards for this month?

As it happens, there are a few problems along these lines as I write this introduction, with barely half the results in. The culprits are likely to be kicking themselves, or at least their extension-handling departments, but as to who these bumblers might be, the accusing finger is pointed below.

### Test Procedures

The usual speed tests were performed – on-demand scanning speeds returned against executable and OLE2 file scanning plus the on-demand scanning speeds against archived executables and OLE2 files. The scanning speed tests double up as false positive tests and the VB 100% award can only be gained by those products having no false positives in addition to full detection of ItW viruses. This includes only 'full' false positives, and not files flagged as 'suspicious', very relevant to one product this month. These tests were performed either directly from the console or, where at all possible, from the console application designed for control of the product. The latter method of testing is assumed to add a little overhead in the use of a console and associated network transfers, though this reviewer suspects that given the added ease of use the console may be considered as a usual operating method. Some may disagree, in which case appropriate weighting should be applied to considerations of scanning rates.

### CA InoculateIT v4.5

| | | | |
|---|---|---|---|
| ItW File | 100.0% | Macro | 100.0% |
| ItW File (o/a) | 100.0% | Macro (o/a) | 100.0% |
| Standard | 99.6% | Polymorphic | 98.8% |

This product, as befits *Computer Associates* whose stock in trade is central administration, had one of the more complete and smooth installation procedures encountered. It, among other procedures, offered to back up important disk information in case of emergency.

Scanning, however, was less of a pleasure if only due to the slowness of the procedure. It seems likely that this is related to logging, since the problem was at first minor, increasing as the scan progressed. As such it should not really be problem in real-world situations unless mass infestations are being scanned. No false positives were encountered and thus *InoculateIT* earned the first VB 100% award of the review.

### CA Vet NetWare Anti-Virus v10.1.9.a

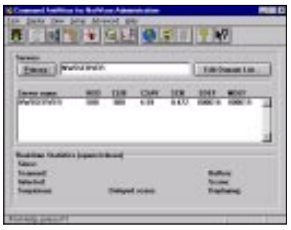
| | | | |
|---|---|---|---|
| ItW File | 100.0% | Macro | 99.5% |
| ItW File (o/a) | 100.0% | Macro (o/a) | 99.5% |
| Standard | 99.8% | Polymorphic | 94.3% |

Despite requiring a degree of manual installation twiddling, since appropriate users are not set by the installation routine, once in place *Vet* performed with no problems or difficulties. As far as detection was concerned *Vet* achieved good results in most areas, with the polymorphics, as for other products in this test, proving to be the sticking point. A full complement of ItW viruses were, however, detected both on-demand and on access which together with a zero false positive rating merited *Vet* with the second VB 100% award in as many products reviewed.

| On-demand tests | File | | Macro | | Polymorphic | | Standard | |
|---|---|---|---|---|---|---|---|---|
| | Missed | % | Missed | % | Missed | % | Missed | % |
| **CA InoculateIT** | 0 | 100.00% | 0 | 100.00% | 9 | 98.87% | 2 | 99.61% |
| **CA Vet AntiVirus** | 0 | 100.00% | 19 | 99.59% | 266 | 94.36% | 2 | 99.87% |
| **Command AntiVirus** | 1 | 99.96% | 3 | 99.70% | 1 | 99.98% | 9 | 99.20% |
| **DialogScience DrWeb** | 0 | 100.00% | 0 | 100.00% | 2 | 99.95% | 0 | 100.00% |
| **Eset NOD32** | 0 | 100.00% | 0 | 100.00% | 0 | 100.00% | 1 | 99.90% |
| **Kaspersky Lab AVP** | 6 | 99.19% | 3 | 100.00% | 0 | 100.00% | 23 | 99.41% |
| **NAI NetShield** | 1 | 99.96% | 3 | 99.97% | 6 | 99.25% | 5 | 99.83% |
| **Norman Virus Control** | 4 | 99.74% | 4 | 99.89% | 286 | 91.23% | 5 | 99.83% |
| **Sophos Anti-Virus** | 0 | 100.00% | 13 | 99.66% | 190 | 95.36% | 15 | 99.54% |
| **Symantec Norton AntiVirus** | 0 | 100.00% | 17 | 99.53% | 259 | 94.81% | 16 | 99.46% |
| **VBuster VBShield** | 79 | 92.61% | 236 | 94.17% | 2595 | 77.46% | 72 | 95.54% |

## Command AntiVirus v4.59

| | | | |
|---|---|---|---|
| ItW File | 99.9% | Macro | 99.7% |
| ItW File (o/a) | 99.9% | Macro (o/a) | 99.9% |
| Standard | 99.2% | Polymorphic | 99.9% |



Another application where control is exerted at a client machine, *Command's* product gained the 'security by obscurity' award for this Comparative. With very little tweaking it was possible to activate the NLM in such a way that only CPU usage was available as a check for whether a scan was progressing. The client also lacked communication ability, the actions on scan seeming to bear little if any relation to those selected at the client.

Having said that, solidly respectable detection rates were not good enough to gain *Command* a VB 100% award this month. Extensionless O97M/Tristate samples were not scanned and one such sample exists in the WildList.

Also lacking was any facility for the scanning of statically compressed archive files, which is reflected in the archive scan rates table. The slow rates of scan encountered for normal files, however, possibly explain this dearth of a feature which would potentially exacerbate the velocity problem yet further.

## DialogueScience DrWeb v4.20

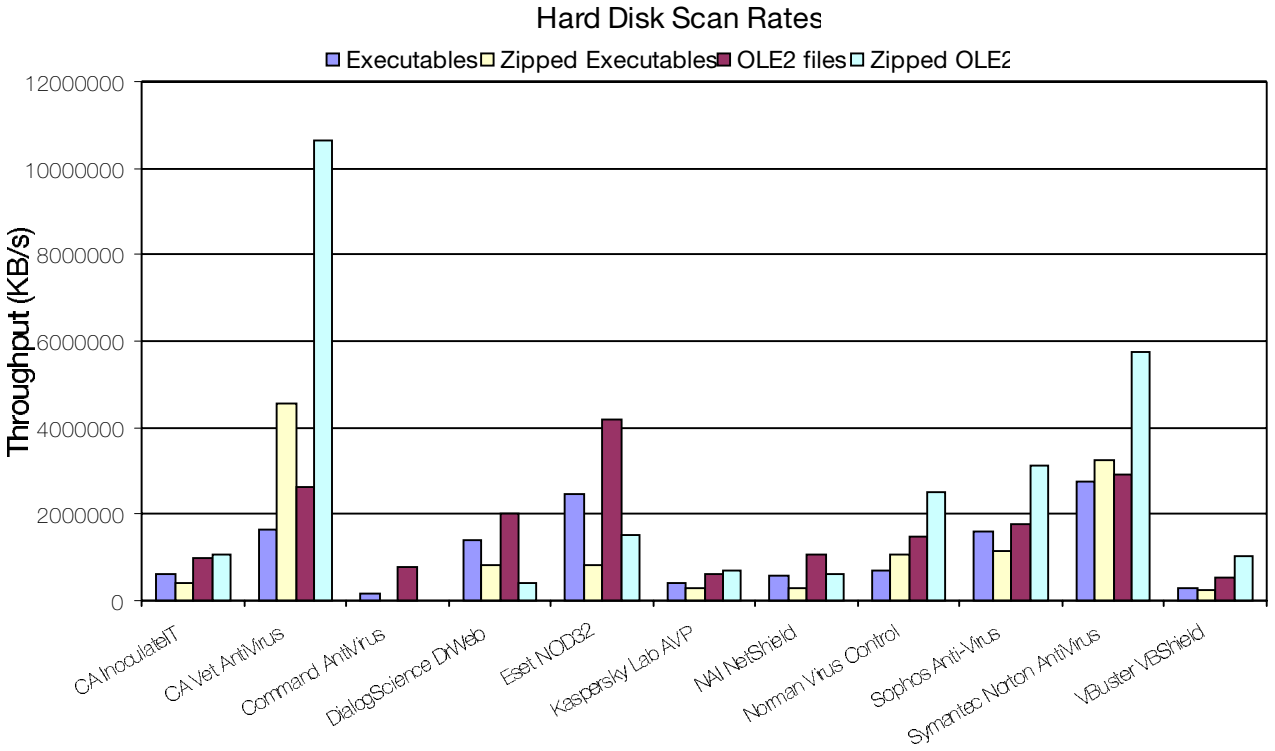
| | | | |
|---|---|---|---|
| ItW File | 100.0% | Macro | 100.0% |
| ItW File (o/a) | 100.0% | Macro (o/a) | 100.0% |
| Standard | 100.0% | Polymorphic | 99.9% |

For reasons unknown, almost all native *NetWare* GUIs in this test were of a standardised blue and white nature, a trend bucked by *DrWeb* which opts for a more ancient monitor-style green screen effect.

The scanning of files on-access does not occur on file opens, a trait which *Sophos SWEEP* for one shares, leading to the on-access testing being performed by moving the virus collection. Somewhat oddly, the copy was allowed to proceed despite the log file showing ample evidence of viral files.

There was also some confusion as to how on-demand scans are performed – the tests were all completed via scheduled jobs. The results for scanning proved to be speedy enough with all ItW viruses detected at a good rate of knots.

Despite numerous 'suspicious' flags, *DialogueScience's DrWeb* can be justifiably proud of its VB 100% award. These suspicious files have, on the other hand, remained constant, not only in the previous *NetWare* reviews but also in *DrWeb's* outings on other platforms and thus remain a tenacious thorn in the flank of *DialogueScience*.




## Eset NOD32 v1.42

| | | | |
|---|---|---|---|
| ItW File | 100.0% | Macro | 100.0% |
| ItW File (o/a) | 100.0% | Macro (o/a) | 100.0% |
| Standard | 99.9% | Polymorphic | 100.0% |

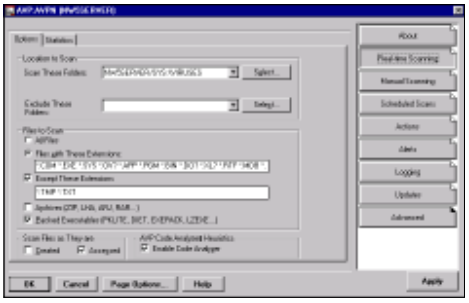
The pair of *NOD32* NLMs provided one of the more minimalist installs in this Comparative, the on-demand NLM being singularly limited to a command-line interface. This interface, however, did not prevent NOD32 from performing at its usual impressive level of detective skill – a level which gains it yet another VB 100% award. The rudimentary nature of control available in this product is a recurring feature in this review and is addressed in the conclusions.

Of all the test-sets scanned, *NOD32* missed only one sample in the Standard set, a feat difficult to improve upon and unique to this review. Coupled with good scanning speeds and no false positives, this is a gratifying result this time around for the Slovakian anti-virus company.

## Kaspersky Lab AVP for NetWare v3.5

| | | | |
|---|---|---|---|
| ItW File | 99.1% | Macro | 100.0% |
| ItW File (o/a) | 99.1% | Macro (o/a) | 100.0% |
| Standard | 99.4% | Polymorphic | 100.0% |

*AVP for NetWare* was the first product reviewed where administration was fully integrated within the NWADMN32 program within *Windows NT*. The ease and clarity of operation was thus much improved over the pure console-driven applications expected of *NetWare* and was the only console which could in fact load the NLM.



On the downside the application suffered from intermittent instability, requiring Dr Watson to be summoned on a couple of occasions. There were also oddities in the method used by *AVP* for counting files, as more were reported scanned than actually existed. Log files caused some confusion, primarily by the marking of files as 'OK' when in fact this referred to file structure rather than a lack of viral content.

It was with *AVP* that the perils of extensions reared their heads once more with the problem areas being the major surprise since the new double extensions were all detected happily. Not alone in missing the extensionless sample of O97M/Tristate in the WildList, the chaps at *Kasperky Lab* will be joined by others in their reversion to problems with this file – problems long since banished on other platforms.

The nature of the console is also of note as a potential slowing factor, the scan rates here being very low indeed. There was a, certainly related, torrent of network activity present while scans were being performed. The console it seems is updated very regularly, rather too regularly perhaps since the scan rates 'felt' much slower than *AVP* operating on other systems. This would appear to be the one flaw in the console, which otherwise performed admirably and, not surprisingly, was always well informed of its NLM partner's status.


## NAI NetShield for NetWare v4.5.0

| | | | |
|---|---|---|---|
| ItW File | 99.9% | Macro | 99.9% |
| ItW File (o/a) | 99.9% | Macro (o/a) | 99.9% |
| Standard | 99.8% | Polymorphic | 99.2% |

*NetShield* is supplied, as would be expected from a network-based company, with a client control program which is a welcome sight in such a review. Perhaps more welcome is that the client neither constantly polls for information, as *AVP* does, nor has information, as in the case of *Command's* submission. This allows information on scan status to be present at the client without overwhelming network activity.

All of this goodwill is, however, frittered away by the speed of scanning through the Polymorphic sets, which moved with a speed akin to the rate of evaporation of granite. Of particularly agonising note was the scanning of Splash, which took several minutes for many of the *VB* samples. This slowness was also reflected in the Clean set testing, where speed was not an *NAI* strong point.

It is lucky, therefore, that detection rates have something to show for all this effort, with good detection across the board except in one simple area. *NetShield* falls among those products which do not scan extensionless files by default and thus denies itself a VB 100% award by the slimmest of margins.

## Norman Virus Control for NetWare v3.98b

| | | | |
|---|---|---|---|
| ItW File | 99.7% | Macro | 99.8% |
| ItW File (o/a) | 99.7% | Macro (o/a) | 99.8% |
| Standard | 99.8% | Polymorphic | 91.2% |

*Norman's* offering for this review achieved notability in the main by its having two names, *'FireBreak'* being the alternative, which were used interchangeably throughout the operation of the program.

It also succeeded in niggling as it was unable to fine-tune scanning within areas smaller than an entire volume. Short of creating a volume specifically for the investigation of viral suspects this makes checking individual files something of an onerous pursuit and leaves all scans on the SYS volume doomed to be extremely lengthy indeed.

The Polymorphic set was the great divider in this month's testing, with all but one product faring well in all other areas. The bane that is ACG.A flummoxed the *Norman* product completely, as it did more than one other scanner, leaving it with the highest aggregate total of missed files in the test. A lack of scanning for .HLP files prevented the detection of W95/Babylonia in the ItW set, which in turn denied the product a VB 100% award.

## Sophos SWEEP for NetWare v3.36

| | | | |
|---|---|---|---|
| ItW File | 100.0% | Macro | 99.6% |
| ItW File (o/a) | 100.0% | Macro (o/a) | 99.6% |
| Standard | 99.5% | Polymorphic | 95.3% |

| | Hard Disk Scanning Speed | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Executables | | | OLE2 files | | | Zipped Executables | | Zipped OLE2 | |
| | Time (sec) | Throughput (kB/s) | FPs [susp] | Time (sec) | Throughput (kB/s) | FPs [susp] | Time (sec) | Throughput (kB/s) | Time (sec) | Throughput (kB/s) |
| **CA InoculateIT** | 895 | 611097 | 0 | 79 | 1004224 | 0 | 375.0 | 425111 | 70.0 | 1065821 |
| **CA Vet AntiVirus** | 329 | 1662407 | 0 | 30 | 2644458 | 0 | 35.0 | 4554760 | 7.0 | 10658214 |
| **Command AntiVirus** | 3611 | 151462 | 0 | 101 | 785482 | 0 | N/T | N/A | N/T | N/A |
| **DialogScience DrWeb** | 395 | 1384638 | [27] | 39 | 2034199 | [1] | 196.0 | 813350 | 185.0 | 403284 |
| **Eset NOD32** | 223 | 2452610 | 0 | 19 | 4175461 | 0 | 196.0 | 813350 | 49.0 | 1522602 |
| **Kaspersky Lab AVP** | 1392 | 392911 | 0 | 130 | 610260 | 0 | 531.0 | 300220 | 110.0 | 678250 |
| **NAI NetShield** | 964 | 567357 | 0 | 75 | 1057784 | 0 | 540.0 | 295216 | 125.0 | 596860 |
| **Norman Virus Control** | 802 | 681960 | 0 | 53 | 1496864 | 0 | 150.0 | 1062777 | 30.0 | 2486917 |
| **Sophos Anti-Virus** | 200 | 1589919 | 0 | 27 | 1762973 | 0 | 49.0 | 1138690 | 13.0 | 3108646 |
| **Symantec Norton AntiVirus** | 344 | 2734660 | 0 | 45 | 2938288 | 0 | 140.0 | 3253400 | 24.0 | 5739038 |
| **VBuster VBShield** | 1787 | 306061 | 9 | 148 | 536039 | 2 | 687.0 | 232047 | 72.0 | 1036215 |

The *SWEEP* NLM falls firmly in the middle ground of control sophistication in this review, the greatest idiosyncrasy being in the area of scanned file selection where recursion is selected by the addition of a '>' to the path.

Irritating from *VB's* point of view is the inability to have logs greater than 999 KB in size, though, as with many *VB* niggles, this is less of a problem in the real world than in *VB* tests. Detection-wise affairs seem to be tightening up after the extension problems of the last Comparative, with a clean sweep in the wild. The NLM does not, admittedly, scan within some file types which might otherwise have upped percentages in the Standard and Macro sets, .MDB being an example.

The objective here is presumably to raise scan speeds at the expense of non-detection of perceived low-risk viral threats, since *Access* infectors are not famed for their rampant spread. This is also the assumed reason for the continued non-detection of W95/Navrhar and Positron, both mid-infectors requiring slower scanning methods for positive detection.

This 'need for speed' ethos proved possibly to be a success as *SWEEP* not only performed quickly on the Clean sets, but with a combination of full detection in the wild and no false positives, *Sophos* regains its position as a VB 100% award holder.

*SWEEP* was once rare in that it routinely detected the same viruses on-demand as on-access. This review was no different as far as *Sophos's* detection rates go, though the uniqueness is certainly a thing of the past – all products managed such detection, a fact reflected in the combined detection rate graphs and tables for on-demand and on-access scanning.

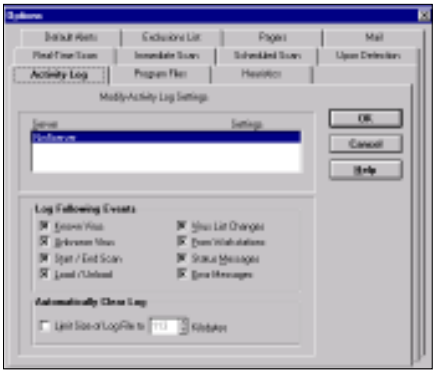
## Symantec Norton AntiVirus for NetWare v4.04

| | | | |
|---|---|---|---|
| ItW File | 100.0% | Macro | 99.5% |
| ItW File (o/a) | 100.0% | Macro (o/a) | 99.5% |
| Standard | 99.4% | Polymorphic | 94.8% |

*Norton AntiVirus* was one of the very few recipients of a VB 100% award in the last Comparative, at which juncture it was pointed out that such a distinction was perhaps not all it could be.

This month, however, surrounded by companies wilfully shooting themselves in the foot, the VB 100% award gained by *Symantec* will be more pleasing to them and is much more representative of a superior all-round performance.

The *NAV* program is one of those sporting a *Windows* front-end to the NLM and this operated with a near complete lack of problems. Admittedly, the viral scanning speed was a little sluggish in comparison with some of the faster scanners on show, though as with the other slightly slow entrant previously mentioned, this was coupled with a good detection rate. Much more impressive was the extremely fast Clean set scanning speed, which by and large put rivals to shame.

Despite these sterling features *NAV* was in fact one of the products which missed the most samples overall, ACG.A being a culprit here as elsewhere. It seems odd that scanning of infected samples should be so slow with so many misses, while Clean set scanning was so fast. The likely explanation seems to be that false positives have been singled out for eradication, thus more checking is done than usual as to the validity of viral identities.

One extra niggle arose during false positive testing when the *Windows* front-end caused a general protection fault, though this was not reproduced on further testing.

## VBuster VBShield for NetWare v1.02

| | | | |
|---|---|---|---|
| ItW File | 92.6% | Macro | 94.1% |
| ItW File (o/a) | 92.6% | Macro (o/a) | 94.1% |
| Standard | 95.5% | Polymorphic | 77.4% |

The only newcomer to the Comparative, as noted in the introduction, *VBuster's VBShield* proved unable to load successfully at the first attempt. It was also the last product scheduled to be reviewed and thus a great deal of frantic information exchange occurred between *Virus Bulletin* and the Hungarian developers at *VirusBuster*. The problem turned out to be a simple oversight on *Virus Bulletin's* part, though the expected activities of the installer were not noted in the documentation supplied, a fault which was not unique to this product. As a new entrant, therefore, somewhat more detail is supplied as regards *VBShield's* capabilities and limitations.

The *VBShield* NLM has a constant on-access thread running, which gives the options to disinfect, deny or quarantine, but gives no deletion option. The on-demand section provides the same options and in both cases the activity may be sent to a log. By keeping the log on screen a measure of information can be observed concerning ongoing scans. These logs are written to a single file which holds details of both on-access and on-demand detections. On-demand scans can be created in a scheduled mode and applied to chosen directories or volumes.

Scanning proved to be no problem, files denied on-access being determined easily by *VB's* in-house tools. With the treatment of the log files, however, problems arose in the parsing of the files. The logs are written to in such a way that detection notifications are logged across several lines and thus impenetrable to the usual *VB* methods. For the sake of reaching deadlines, the on-access results were used for both on-access and on-demand results, this being in accordance with trends across the other products.

*VBShield* is clearly somewhat outclassed in detection by the other products reviewed this month, though in fairness the best comparison is with *VBShield's* performance in previous reviews. From that point of view there is good news, since detection rates are up, most specifically in the Standard and ItW sets.

Speed remains something of a problem, but it is good to see that the important issue of detection is indeed being addressed by the developers.

### Notes on Testing

As noted in the comments for *Sophos SWEEP* the results this month were unique in one aspect, this being the first time that this reviewer can recall all products rating the same for detection for the on-demand and on-access tests. This is attributable at least in part to the lack of boot sector virus testing in the *NetWare* test regime.

It also seems likely that the *NetWare* environment itself is one which makes this more likely, since the situation was very nearly the same on the last occasion that a *NetWare* Comparative took place. Although this reviewer would be much more happy performing only one set of tests for both on-access and on-demand testing, this does not seem likely in the future.

Furthermore, some products showed scan speed results far below those that would be acceptable in a corporate environment. The *VB* test machine is clearly at the extreme low end of performance for a *NetWare 5* server – a situation which might cause concern among readers.

The reason behind this apparent lack of server technology is not all due to *Virus Bulletin's* penury. By using a machine which is at the limits of performance a better understanding of scan rates under stress can be gained than on a machine running at one or two percent CPU usage. The scan throughput rates, like the other figures supplied, are not valuable in isolation but as a means of comparison between the scanners.

### Conclusion

After the last Comparative's warping by the addition of VBS/Unicle.A to the WildList, this test did not hold as many potential pitfalls in the way of new viruses added to the Standard set, though many products managed to fall over on the detection of old favourites instead.

The age-old problem of extensionless samples fooled a disturbing number of the products in the line-up, a quirk more disturbing since the same samples caused problems in the *NetWare* Comparative of July 1999. The same story was repeated in the Polymorphic test-sets, where both the last Comparative and this one saw mass problems with ACG.A and to a lesser extent ACG.B for some products.

There have, however, been minor improvements on that last *NetWare* test in other areas – the overall detection rates are up and false positives on those products tested on both occasions are very marginally down. Major differences in manageability are, though, hard to come by in all but the case of *Kaspersky Lab's AVP*, the product (in this reviewer's opinion) which is most fully integrated with the *NetWare* operating system.

The manageability of some products in fact borders upon the arcane, with features absent which would be taken for granted on any other platform. Being unable to tell whether a scan is operating, to monitor a scan, to delete offending files or to be unable to run an on-demand scan would instantly consign a *Windows* or Mac product to exile, to the tune of alternating hails of derision or shrieks of laughter.

What lies behind the state of the *NetWare* market? *NetWare* as an operating system has certainly suffered from the rise of *Windows NT* as an, arguably, secure platform for large networks. In addition to this, the Console One interface of *NetWare 5.x* is sufficiently doddering that few could consider *NetWare* to be on a par with *Windows* systems as far as new-user friendliness goes.

For these reasons the anti-virus developer community may have decided that *NetWare* has had its day and that a *NetWare* scanner, although useful for the sake of completeness, should not be allocated a great deal of development time. It also seems, at a guess, possible that the aspects of security involved in remotely administering a *NetWare* product might well make this a task few would relish.

The reviewer's personal opinion, however (also a wild hypothesis), is that there has been no real call for improvements. *NetWare* users are used to obscure and Byzantine procedures for the simplest of tasks. Thus, the odd and obscure ways of some of the scanners tested are perfectly at one with the *NetWare* environment and likely to exist long after we are all safely tucked up in retirement homes.

**Technical Details**

**Server:** 500 MHz Athlon with 6 GB HD, 64 MB RAM, CD-ROM and 3.5-inch floppy running NetWare 5.1 with Service Pack 1.

**Workstation:** 166 MHz Pentium with 4 GB HD, CD-ROM and 3.5-inch floppy, running Windows NT 4 with Novell's Client for Windows.

**Virus test-sets:** Complete listings of the test-sets used are at http://www.virusbtn.com/Comparatives/NetWare/200009/test_sets.html.

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

**SUBSCRIPTION RATES**

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

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

**Editorial enquiries, subscription enquiries, orders and payments:**

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

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

**US subscriptions only:**

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

Tel (781) 2139066, Fax (781) 2139067



# END NOTES AND NEWS

**It is not too late to reserve your place at VB2000, *Virus Bulletin's* 10th international conference, which takes place on Thursday 28 and Friday 29 September 2000 at the Hyatt Regency Grand Cypress Hotel in Orlando, Florida.** Discount prices are offered to current subscribers, bona fide charitable/educational organizations and for multiple delegate bookings from one company. Download a registration form or a conference brochure containing programme details of the line-up of technical and corporate sessions, evening social events, product exhibition and hotel accommodation information from http://www.virusbtn.com/. For more information contact the conference organiser Karen Richardson; Tel +44 1235 544141 or email VB2000@virusbtn.com.

There are currently exhibition opportunities at the **Windows 2000 eNTerprise Exhibition and Conference** to be held in the Grand Hall at Olympia, in London's Earls Court from 21–23 November 2000. For further information about the event contact Deborah Holland; Tel +44 1256 384000.

***McAfee*, an *NAI* business, has launched the *McAfee AVERT Wireless***, a wireless Web service that enables mobile phone users to access the latest information about Internet viruses and malicious code. Available now, the service offers users with Wireless Application Protocol (WAP)-enabled mobile phones round-the-clock access to the latest virus news. For more information on *McAfee AVERT Wireless* visit the site; http://www.mcafee2b.com/.

The 17th world conference on Computer Security, Audit and Control focuses on all aspects of e-commerce. **CompSec 2000 takes place from 1–3 November 2000 at the Queen Elizabeth II Conference Centre in Westminster, London, UK.** For details, visit the Web site http://www.elsevier.nl/locate/compsec2000 or contact Gill Heaton; Tel +44 1865 373625.

***Sophos* is to host a day-long course entitled 'Managing Internet Security' on 14 November 2000** at the organization's training suite in Abingdon, Oxfordshire, UK. From 15–16 November a two-day course on Implementing Windows NT Security will take place at the same location. For more details about the different courses and training days available, or to reserve your place, contact Daniel Trotman; Tel +44 1235 559933, or email courses@sophos.com.

**The 16th Annual Computer Security Applications Conference (ACSAC)** will take place from 7–11 December 2000 in New Orleans, Louisiana, USA. Email publicity_chair@acsac.org or visit the Web site http://www.acsac.org/ for more information.

***F-Secure Corporation* has announced a partnership with *Fujitsu Siemens Computers* in Munich, Germany** whereby *Fujitsu Siemens Computers* will ship *F-Secure* anti-virus software on the Driver & Utility CD included with every professional PC and *LIFEBOOK* Notebook. The arrangement is effective immediately. For more information see the *F-Secure* Web site: http//www.F-Secure.com/.

***Central Command Inc* has launched its Live Virus Training and Certification Program.** For more information on the four levels of certification available, contact; Tel +1 330 723 6062 or email certification@centralcommand.com.

**The UK Security Show 2001, incorporating The IT Security Showcase**, is to take place at Wembley in London, UK from 14–15 February 2001. For further details about the prospective programme and current exhibition opportunities, visit the event's Web site http://www.securityshow.com/.

***MessageLabs* has announced a joint venture with *KPN*, a telco operator in the Netherlands**, to host a Virus Control Centre in *KPN's* CyberCentre in Amsterdam. The *MessageLabs* Virus Control Centre scans over two million emails a day for 250,000 individual users. For more details, see the Web site http://www.messagelabs.com/.

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

**The Internet Business Exhibition & Executive Conference is to be held at the Brighton Metropole, UK on 5–6 December 2000.** There are currently sponsorship and exhibition opportunities available. For more details contact Richard Cole; Tel +44 1273 773224 or visit the Web site http://www.ibshow.com/.