ISSN 0956-9979 JUNE 2001

VIRUS BULLETIN

THE INTERNATIONAL PUBLICATION ON COMPUTER VIRUS PREVENTION, RECOGNITION AND REMOVAL

IN THIS ISSUE:

• **Recognise these?** No fewer than four culprits are analysed – two worms and two viruses. They are good indications of the developments and changes virus writers are incorporating into their creations, starting on p.6.

• **Will this ethic work?** Jaak Akker puts the case for a code of good conduct in the anti-malware arena according to his experience as Chairman of *SIG Sec*. See p.13.

• **A morality tale:** the *Open University's* David Phillips reminisces about the lessons he learned following a new AV roll-out for his off-campus students. Home users take note, on p.15.

CONTENTS

COMMENT

Fear of Flying

“ I couldn’t have wished for better wing men ”

The reactions to my imminent departure as Editor of *Virus Bulletin* have ranged from the touching to the downright depressing, but none amused me quite so much as this one – cryptic or cruel, you decide –‘why soar like an eagle when you can fly like an albatross?’. Everyone fears change, but in an industry like ours change is not just inevitable, it’s what separates the high fliers from the stragglers. When I started here I despaired of ever getting off the ground. Having spent no less than four years in the giddy slipstream of some of the biggest names in AV, I am proud to have flown the magazine to its current position ahead of the rest of the field.

It helps to be a ‘sticky beak’ in this job and I must admit I have enjoyed ruffling some feathers in the News pages from time to time. If it’s not a rooster it’s a peacock whose wings need clipping. It hasn’t all been plain sailing. Sometimes the issue comes in on a wing and a prayer. Sometimes I’ve been cleared for landing ahead of schedule and there’s an eerily clear flight path and nothing but blue sky behind me. No matter how crazy things get, it’s all water off a duck’s back to the heavy mob who backs me up. Sincere thanks and ‘sayonara chaps’ to Jake (Jakub Kaminski), Beard Man (Nick FitzGerald), Whallers (Ian Whalley), Richard and Ed. Likewise, I can always rely on the unflappable knowledge of the wise old owls (I can hear Sarah now, ‘less of the old!’) on the Advisory Board. It’s been a real honour.

So what’s over the horizon? While the view from my perch is certainly changing, it’s good to have familiar landmarks. I am sure that the same old feathers will keep flying in all directions within *CARO*, and that there’ll be buzzards circling over the odd mess in the middle of the road where a corporate Juggernaut fails to put the brakes on. Conspiracy theorists predicted a time when the ‘big four’ anti-virus companies would rule the roost but, ironically, I think that the world getting smaller will be one of the reasons this monopoly of sorts won’t fly. While the big brand names still dominate the market-share, the more exotic AV breeds are becoming the ones to watch.

Which came first – the chicken or the egg? The virus or the anti-virus? Users are getting braver and beginning to ask questions of an ethical nature. Amid the squawking , the very term ‘anti-virus’ is getting lost as the species starts diversifying. The industry will only evolve if trust between vendors and users becomes mutual. Birds of a feather flock together but I’ve witnessed the AV industry become more welcoming. Integration is already under way. Look at the speaker line-up for VB2001 in Prague in September. Alongside the famous anti-virus names you’ve come to expect there are industry watchdogs, corporates, academics and product testers ready to share information.

Vendors shouldn’t be afraid of customers flying south – they are in for the long haul. I know this because of the rising numbers of *Virus Bulletin* conference delegates in San Francisco, Munich, Vancouver and Florida. They are not burying their heads in the sand – they are renewing their subscriptions to *VB* for three years at a time. Users remember all too well how Melissa and the LoveBug put some pretty scary cats among the pigeons. I probably shouldn’t admit it but I am still using *Windows 95* – there are dodos like me all over the place, and we still rely on AV to keep our environments safe so we don’t die out.

Finally, I want to thank *Virus Bulletin’s* readers. You make us what we are. We will continue to ask questions on your behalf and sort the factory-farmed from the free range. I shall encourage the new Editor to continue to peck a few ankles. Like any synchronised flying team, we’re a tight unit here and I couldn’t have wished for better wing men. Matt’s got the quick-eyed tenacity (not to mention the plumage!) of a raven, while Bernadette is every bit the lovely swan. There’s not a ripple on the surface but she’s paddling like mad underneath to keep *VB* on course. My favourite compliment for the job I’ve done? You must have guessed by now – ‘not a bad effort, for a bird’. This is not goodbye, but ‘au revoir’. I shall be swooping into Prague to introduce my successor in person – so, best behaviour everyone, and the last one to the bar is a lame duck!

Francesca (Ceskie) Thorneloe, Editor

NEWS

Eleven Years of VB

We are pleased to enclose with this issue of *Virus Bulletin* a complimentary copy of the 11-year back issue CD which contains PDF versions of the *Bulletin* from 1989–2000. Subscribers using *Adobe Acrobat Reader v3.0+* can search the comprehensive index for a particular virus analysis or test result. There are plans to release a back issues CD every year ▮

All Change?

Rod Fewster, *KAV* (née *AVP*) distributor in Australia, has dropped the *Kaspersky Lab* product in favour of Slovakian company *Eset's NOD32*. This parting of ways appears to be mutually amicable – that is, compared to *Kaspersky's* scrap with *Central Command* in the States. Given the above reshuffle and the recent announcement from *F-Secure* of a 20% cut in staffing levels, this looks to be a major realignment in the sales policy of *KAV* in particular and also the smaller AV players (in terms of market share) in general.

F-Secure remains optimistic about the future of its flagship anti-virus products and business partnerships, on which it intends to concentrate, and blames a fall in demand in the corporate encryption area. As customers become more confident of their requirements of the anti-virus market, does this mark the beginning of a global modification of the industry's topography? ▮

AVAR 2001

The fourth Anti-Virus Asia Researchers Conference will take place from 4–5 December 2001 at the New World Renaissance Hotel in Hong Kong. The organisers of the event are calling for paper submissions that address any area related to computer viruses. Prospective authors should submit an abstract of not more than 200 words in plain text, PDF or *MS Word* format to avar-papers@yuikee.com.hk no later than 15 June 2001. The official language of the conference is English but Chinese translation will be supported at the event.

For more information about the conference or the call for papers, see the Web site http://aavar.org/ ▮

Diabolical!

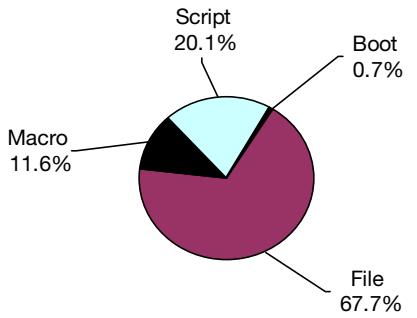
UK ISP provider *Demon* has egg on its face this season. The spring issue of its freebie magazine @DEMON carried a story on viruses which was riddled with inaccuracies and howlers of all descriptions. Our personal favourite reports Melissa as the 'first Windows macro virus'. One wonders where *Demon* gets its research from and who cleared the story for inclusion on its Web site ▮

Prevalence Table – April 2001

Virus	Type	Incidents	Reports
Win32/Hybris	File	1044	35.3%
Win32/MTX	File	350	11.8%
Kak	Script	276	9.3%
Win32/Magistr	File	209	7.1%
Win32/Navidad	File	163	5.5%
VBSWG	Script	107	3.6%
LoveLetter	Script	105	3.5%
Laroux	Macro	92	3.1%
Divi	Macro	72	2.4%
Tam	Script	50	1.7%
Win32/Funlove	File	45	1.5%
Win32/QAZ	File	41	1.4%
Win32/Msinit	File	40	1.4%
Marker	Macro	28	0.9%
VCX	Macro	26	0.9%
Win32/Ska	File	25	0.8%
Ethan	Macro	21	0.7%
Win32/BadTrans	File	21	0.7%
Stages	Script	19	0.6%
Pica	Script	18	0.6%
Win32/Pretty	File	16	0.5%
Win95/CIH	File	13	0.4%
Barisadas	Macro	12	0.4%
Netlog	Script	12	0.4%
Tristate	Macro	12	0.4%
Melissa	Macro	11	0.4%
Others[1]		131	4.4%
Total		2959	100%

[1] The Prevalence Table includes a total of 131 reports across 47 further viruses. Readers are reminded that a complete listing is posted at http://www.virusbtn.com/Prevalence/.

Distribution of virus types in reports



LETTERS

Dear Virus Bulletin

The Disinfection Debate

As frequently happens, both opinions in the disinfection argument featured in last month's issue are quite right in what they say – the rationale was perfect! The reason for disagreement seems to lie in the fact that Dr Ford is drawing his conclusion from quite a realistic and pragmatic position, while Mr FitzGerald is defending a purely academic point of view.

Scanners are part of everyday computer hygiene now – they are a familiar element of life with all the associated problems: no backups, tangled tapes, slow IT departments and a necessity to reduce downtime. Therefore, as much as I would like to live in the better world envisioned by Mr FitzGerald, I rather have to agree with the conclusions of Dr Ford.

One argument, however, was missed. There are users who prefer to live with a virus on their computer until they run into trouble. Such people frequently do not use a scanner at all, and spread infections to their mates. I suppose Mr FitzGerald would suggest these individuals are isolated from other people! But, realistically, disinfection is the only way these poor souls may start fighting their personal hygiene problems! The academic approach may be good for a properly maintained environment like a research lab or modern hi-tech company where you have several layers of backup protection. But even in these rare environments, a scanner that offers cleaning and does a better job at disinfecting more viruses would have an advantage. It will simply be a safety net below the backup solution.

The bottom line is that users should be given a choice whether they use a scanner's cleaning capability or not. If restoring from a backup is going to be easy, it is always better to go that route! That is why having automatic cleaning in default mode is generally not a good idea. There is always a small risk that cleaning may go wrong and you will end up with another problem. But we all know there is a risk when we simply walk in the street. Does it really mean we should not allow ourselves to have some fresh air from time to time?

Igor Muttik
Network Associates Inc
UK

Furthermore…

Dr Richard Ford and Nick FitzGerald both brought forth some very valid points, but in the end they managed to side-step the real reason for AV companies having to implement repairs for viruses. The key points they made are as follows:

Repair is good because

- some repairs are easy to do and can restore back to the original 100%
- other ways of restoring the system to a known, clean state are more expensive (both in time and resources)
- repairs do not change user settings
- backups are not always available
- repairs can't make things worse than they are
- the risk of a bad repair is minimal

Repair is bad because

- most repairs cannot restore the original 100%
- in the long run, it is more expensive to repair (both in time and resources)
- most repairs cannot change user settings (that were modified by the virus)
- if the work is important then there must be a backup
- repairs do make things worse than they are, some repairs need to modify the host so much that the host becomes unrecognizable (possibly preventing detection of underlying Trojans/worms/viruses)
- the risk of bad repair lies in undetectable variants of older worms/viruses and thus, it is not minimal

Both views are correct, but the focus shouldn't be on whether repairs are good or bad but rather on *when* to apply them. The real reason for having a repair available to the customer *is* the customer. Each one is unique and has unique needs. The customer cannot and will not be subjected to some number crunching that calculates what supposedly is the norm for him.

Ultimately, it is the customer who decides how he wants to deal with specific situations, based, of course, on information that has been presented to him. We are in the business of providing solutions and fixing problems, not creating them. That's why we include repair as one of the possible solutions to deal with an infected system.

Finally, I would like to make a note regarding the risk of undetectable variants. These are created by a repair of another virus, and are usually the result of bad detection in the first place, such as using a CRC for detection. The problem lies with the original detection, not the repair.

Atli Gudmundsson
SARC
Netherlands

And Another Thing

Richard Ford and Nick FitzGerald discussed the issue of virus disinfection. I have no doubt that the topic could easily fill a few thick volumes printed in subscript. This is one of those interesting issues which look plainly clear in theory and are extremely complex, difficult and unsolvable in practice.

In an ideal world, infected files and affected systems should be restored from backups. The idea of businesses relying on virus disinfection to restore their affected systems is as ridiculous as the idea of using those systems in the first place. However, in real life, everyone uses disinfection and whoever would try to rally against it is quickly considered insane, or at least completely ignored.

To keep it short: disinfection is evil. And these are some of the reasons why:

- Disinfection is tempting
- Disinfection is convenient
- Disinfection is what the public wants
- Disinfection is a feature that sells
- Disinfection gives a false sense of security
- Disinfection is an excuse for laziness (in maintaining backups)
- Disinfection is an excuse for sloppiness (in executing infected code)
- Disinfection is an excuse for incompetence (in system securing)
- Disinfection replaces precaution
- Disinfection is used as an everyday cure
- Disinfection should be used only if there are no other means left
- Disinfection is complex and sophisticated
- Disinfection is unpredictable
- Disinfection pretends to be your best friend

Sooner or later, you'll be sorry you relied on it.

Jakub Kaminski
VB Technical Editor
Australia

Then Again

I don't see the reason for discussing whether disinfection is good or not. Leaving an AV customer without disinfection is like a doctor telling a patient 'Yes, you're ill. I am very sorry about this. You should have been taking vitamins to avoid catching this disease'. In my opinion, disinfection is a major part of every modern anti-virus program. Without it, AV is just a pricey diagnostic tool that does not *solve* the problem, but confirms the presumption of virus infection, which in many cases is useless. On the other hand, disinfection should be added only for malware that really deserves it. Primarily, it should concern all viruses currently found in the wild. It is a must for an AV product to provide its customers with an effective, comprehensive (including restoration of infected objects' original content and rebuilding the *Windows* system Registry) and quick disinfection *option*, not a kind of compulsory feature that cannot be switched off. At the same time, I don't see the point of developing disinfection routines for malware that is not ItW and is unlikely ever to be. Imagine anti-virus experts wasting their time developing disinfection for ancient DOS viruses that will never ever strike customers. In this case they will certainly diffuse their efforts and will not be able to concentrate on really hot issues such as 'wild' viruses.

Denis Zenkin
Kaspersky Lab
Russia

The Tester's View

In my opinion, disinfection is very useful for users, but only under some restrictions, and if it is implemented correctly. Having read last month's debate, I looked at some of our test results (http://www.AV-Test.org) and especially checked the disinfection part. Then I made a short 'overall' statistical analysis and I was really impressed: of DOS and Win32 file viruses, only 75.1% of all files were disinfected correctly; about 1.9% were badly cleaned but still executable; 7.3% were destroyed; and 15.8% were not disinfected at all. (I should mention here that not all anti-virus programs are able to remove file viruses – this was counted here as 'not cleaned'.) Of all sorts of macro viruses only 76.5% were cleaned correctly; 14.3% of the files had some 'bugs' (like macro virus warnings or warnings about a corrupted macro storage); 2.9% of the files were completely destroyed (crashes during opens in *Word* or when the VBA Editor was started); and 6.3% were not cleaned at all (most of them are PP97M and XF macro viruses).

Therefore, a file should only be disinfected if the original state can (as nearly as possible) be reached again, and not just because 'if we include a new signature we add a disinfection routine, too'. Secondly, in modern viruses, worms and backdoors, disinfection does not only remove the virus or delete one single file – it is much more complex. In some malware, a lot of files and Registry values have to be changed – but only a small subset of programs are doing this right now.

Thirdly, a message to vendors – please test all disinfection routines on more than just a couple of files, and on one machine only. In my eyes, only ItW viruses and viruses a company receives from a customer should be 'disinfectable' by the program. This saves time and money with which to concentrate on the 'correct disinfection' of the really important viruses, which can take some time.

Andreas Marx
University of Magdeburg
Germany

VIRUS ANALYSIS 1

Mind the Gaps

Costin Raiu
Kaspersky Lab, Romania

At approximately the same time the .X variant of the VBS worm VBSWG struck worldwide, I received a call from a friend who administers a small cluster of servers for a Web development firm in Romania. He was bothered by the fact that his Web site appeared to have been hacked into, with the entry page replaced by a black one, containing an offensive message regarding the US Government and something (someone?) named 'PoizonBOx'.

It only took a couple of hours to receive a forwarded *CERT* alert which cast some light over the incident. 'CERT Advisory 2001-11' covers a computer worm known as 'Sadmind' which hacks *Solaris* systems and subsequently attacks random *Microsoft* IIS *v4* and *v5* servers from the Internet by replacing their root page with a custom one, stating the message I mentioned before.

The Worm

The Sadmind worm is an interesting combination of six Sparc ELF and ten script/text files, which travels between machines running the Sparc version of *Solaris*. Sometimes the worm travels as 17 files instead of 16, because of a traditional Unix crash 'core' image file that is usually present in the worm directory – this is copied along with the worm code during replication.

To replicate, the worm makes use of a very popular buffer overflow in '/usr/sbin/sadmind', a component of the system administration suite software *Solstice*. The code used in the worm to exploit this bug was originally written by someone going by the nickname 'elux' (elux@synnergy.net). However, it is my belief that that person has nothing to do with the worm's author.

Two ELF executables are used to break into remote systems – one named 'brute', authored by the person named 'elux', and one named 'sadmindex-sparc', originally written by someone who calls himself Cheez Whiz. 'brute' is basically a brute-force wrapper for the real exploit; since the buffer overflow in the sadmind program requires a very precise stack offset to work, 'brute' attempts various offsets in order to find the right one. During such attempts it is quite likely for the sadmind program to crash, so attacked systems usually have entries reporting 'Segmentation Fault – core dumped' or 'Bus Error' in their syslogs.

The worm also uses another ELF executable to search for suitable hosts – *Solaris* systems for infection, and *MS IIS* servers to deliver the payload. This executable, named 'grabbb' is again written by a third party, someone going by the name 'scud', from a large organization known as 'teso'. This small program attempts to connect to a range of IP addresses, either to a specific TCP/IP port, or a range of ports, and grabs the output (usually known as 'banner') sent by the respective host(s) after the connection is initiated. It does this using multiple sockets at the same time, and is generally quite flexible, allowing a multitude of options to control timeouts and so on.

The remaining three executable files, 'nc', 'wget' and 'gzip' are standard and widely used utilities in the '*nix' world. 'nc', short for 'netcat' is exactly as its name suggests, a network-aware 'cat' program, which can basically create a link between stdout/stdin and a remote system via the TCP and UDP protocols. 'gzip' is nothing more than the popular 'GNU ZIP' implementation, used to compress and decompress data, while 'wget' is a simple tool which can download files from remote ftp and http servers.

The remaining ten text/script files are used to propagate the worm and attack IIS servers. The scripts are written both in Perl and the standard Bourne Shell. To ensure that Sadmind will run on systems without Perl installed, the worm takes care to download and install Perl 5.005, from an ftp host in China – 'bak-px.online.sh.cn'. That, along with a couple of other things, suggests that 'Sadmind' was written by a Chinese hacker, like the recent Ramen worm.

Execution and Replication

Whenever an infected system is restarted (or infected, see below), the main worm entrypoint, named /dev/cuc/start.sh is launched from the /etc/rc2.d/S71rpc system script. 'start.sh' will first check for the presence of a directory named '/dev/cub, and create it if necessary. This directory will be used by the worm later for the logs and inter-components communication tasks.

Next, 'start.sh' will launch three other scripts, respectively 'time.sh', 'sadmin.sh' and 'uniattack.sh'. The worm will create five different instances in memory of 'sadmin.sh' and 'uniattack.sh', obviously with the purpose of increasing the overall attack power. 'sadmin.sh' is the part of the worm responsible for the propagation of the worm code to other *Solaris* systems.

'time.sh' will only be started once, and will be responsible for two main tasks – first, to terminate idle sessions spawned by the worm to attack *MS* IIS servers, and second, to deliver the worm's local payload. This is executed after the worm managed to crack 2000 IIS servers, when all the local files named INDEX.HTML will be replaced with a custom one.

In order to find other potential targets, the worm uses a small Perl script to generate 16 random bits which fill the

first two bytes of a TCP/IP address, such as 'a' and 'b' in the generic 'a.b.c.d' TCP/IP address. The worm will then systematically iterate the remaining two bytes, checking if any of the respective addresses run the 'portmap' service, that is, if port 111 of the remote host is open.

To implement the check reasonably quickly, the 'grabbb' program I mentioned before is used, with settings to wait three seconds for each connection attempt, and running 25 parallel checks (threads) at the same time. Next, the worm will sequentially check the list of hosts that seemed to run the portmap service, and use the common tool 'rpcinfo' to check for a registered service with the index '100232', which is the 'sadmind' system administration daemon. If such a service is found, the worm attempts to hack it, using the 'brute' pre-compiled executable. The 'brute' copy present in the worm was probably modified by the author to deliver its payload on port 600 – that is, to add a root shell in 'inetd.conf' with the name 'pcserver', listening on port 600 for remote connections.

Obviously, if the hacking attempt succeeds, the worm will continue by adding a standard '+ +' line to the root's '.rhosts' file, allowing anyone to authenticate with the respective machine as 'root' via rlogin, rcp and so on. Next, the worm will archive a copy of itself from '/dev/cuc/' using the 'tar' utility, and use 'rcp' to copy it to the target system. After that, the worm connects to the remote system on port 600, adds a line to '/etc/rc2.d/S71rpc' to execute 'start.sh' upon each system reboot, gets a copy of Perl 5.005 from the Chinese site I mentioned earlier, installs it using 'pkgadd' and exits, but not before taking care to launch itself on the remote system as well.

The other important script, 'uniattack.sh' will likewise generate random IP addresses, and after checking for an http daemon listening on port 80, it will attempt to hack them using a recent bug in *MS* IIS *v4* and *5* servers, described in the *Microsoft* Security Bulletin MS00-078.

The Perl script that implements the attack is quite large, over 700 lines of code, mainly because the worm will attempt to use 14 different methods to exploit the vulnerability. The payload replaces the entry page of the IIS server with one containing the same message as the one that will replace the local 'index.html' files on the *Solaris* system after 2000 successful cracks. In *Netscape Navigator 4.x*, the page will not always look as expected (big red letters on a black background) – sometimes it will be black letters on a black background, sometimes black on white. This is probably due to a bug in *NN 4.x*, since both *IE* and *Netscape Navigator 6* seem to display it correctly.

Logs

The worm maintains logs both of hacked *Solaris* systems and compromised IIS servers. The logs are stored in the '/dev/cub' directory, and named RESULT.TXT (the file that stores the compromised IIS servers), and SADMINHACK.TXT (the file that contains the IP addresses of the *Solaris* systems to which the worm managed to replicate. Besides those, the worm will also create a large number of files containing results from the 'grabbb' utility, also in the '/dev/cub' directory, with names like A.B.TXT, where A and B are two random bytes representing the upper half of the IP address classes tested by the 'sadmin.sh' and 'uniattack.sh' scripts. Using the respective files, one can trace the infected servers.

Origins

There are indications that the worm is of Chinese origin. Judging from the message the worm puts into cracked Web pages, the author didn't seem to be a fan of 'PoizonBOx'. 'PoizonBOx' is the name of a group of pro-US hackers who hack Chinese Web sites and replace their start pages with various anti-Chinese messages.

The worm does contain a possible contact address for the author as sysadmcn@yahoo.com.cn – the local Chinese version of Yahoo.com. That, along with the use of the Chinese ftp site 'bak-px.online.sh.cn' leads to the conclusion that the author had at least some strong Chinese connections. He definitely wasn't an expert. For example, in many places throughout the source, he uses constructions such as 'j=`/bin/echo "$j+1"|/bin/bc`' to increment a variable, instead of the usual 'j=`expr $j + 1`'.

Conclusion

The fact that a worm exploiting a 2-year old vulnerability can actually spread is proof that too many users, and even worse, system administrators, don't really pay enough attention to security updates or keep their systems patched. Unfortunately, the worm-infiltrated entry page is still there – no-one can have checked the relevant machines for long periods of time. Finally, the AV world will soon have to pay more attention to *Linux*, *Solaris* and the others, since I'm sure we haven't seen the worst yet.

Solaris/Sadmind.A

Aliases:	SunOS/BoxPoison.worm.
Type:	Network-propagated *Sun Sparc/Solaris* worm.
Payload:	Attempts to exploit a bug in IIS *4* and *5* servers, and replaces their index page with a custom one. After hacking 2000 IIS servers, replaces all local index.html files with one similar to those used in hacked servers.
Detection and disinfection:	*Solaris*: delete '/dev/cub' and '/dev/cuc', remove the worm-added line from '/etc/rc2.d/S71rpc'. IIS servers: reinstall the affected pages from backup.

VIRUS ANALYSIS 2

Play Listi for Me

Markus Schmall
OAR Development, Germany

W97M/Listi.A, the first virus claiming to be native to *Office XP*, appeared in mid-March 2001. Tests related to this analysis have been performed on an English version of *Word XP*. From the OLE perspective, files generated by *Word XP* should be directly scannable using current AV scanners. The differences are minimal, and even the main VBA revision number remains the same.

This virus resides within the ThisDocument stream and consists of a Document_Open macro and a polymorphic function designed to change typical names and to make detection based on first generation checksumming routines harder. The polymorphic function is comparable to engines first found within the W97M/Pri family.

A predefined set of strings will be modified inside a string, which is passed to the function as a parameter (typically the complete virus body as used in this virus). There are no additional tricks (e.g. parasitic behaviour or (in)line order changing) within the polymorphic code, which would make it harder to detect the virus using traditional checksum or scanstring technologies.

The macro is called Document_Open and contains the main part of Listi. First, the virus sets the built-in macro virus protection to low (= value 1) using simple Registry functions. By doing this all macros can be executed and no messagebox will appear to warn the user. The default setting for the macro virus protection is high (= value 3), in order that no untrusted macros may be executed.

Next, access to the VBOM (Visual Basic Project Object Model) is enabled using a Registry write operation (key: 'HKEY_CURRENT_USER\Software\Microsoft\Office\10.0\Word\Security', 'AccessVBOM'). The current trust status can be controlled by looking at the macros/security menu item. The 'Trusted sources' dialog contains all the necessary information. Actually, this dialog is comparable to that found within *Office 2K* with the additional information, which is related to the VBOM.

First introduced in *Office 2K*, *Office XP* also allows third party companies to install virus killers/scanners. It is quite remarkable that viruses do not try directly to attack programs installed using this interface. The same thing can be achieved using Registry manipulation. When *Microsoft* published the first beta versions of *Office XP*, it was announced that the activation code for the VBOM access would be heavily protected and that the activation key would be generated uniquely for every machine, so that virus writers would have no easy way to enter the system.

Looking at this virus shows the opposite to be true. The protection seems to be as 'secure' as the protection found in *Office 2K* and easily overrideable using a single write operation to the Registry.

Nevertheless, Listi tests if it can access the VBOM. If not, it activates the VBOM access using the above technique and quits. Otherwise, it continues the program flow and starts the replication routine, which is partly designed to cheat first generation heuristic engines. The routine for testing/activating access to the 'VB Project' could have been better, as the security mechanisms are deactivated by the first Registry write operation and the virus has the chance to continue directly with its work.

Next, the virus checks if the status of the current active document is read only. If so, all the file attributes will be removed and it will be reloaded. Additionally, all file attributes for the global document template will be removed. Again, this part is written utilizing simple anti-heuristic techniques. The program flow continues with the modification of the SaveNormalPrompt option and a check as to whether the current document or global document template contains the macro code. This check is performed using a unique trick which utilizes the 'IIF' VBA method. This part of the virus can be rated as 'anti-heuristic' despite the rare usage of the 'IIF' method.

The replication routine itself is a typical *Office 97* SR1-compatible, line operation-based routine (using 'lines', 'countlines', 'deletelines' and 'insertlines' functions). After the replication routine, the virus parses through all accessible tasks and tries to close all tasks which contain the string 'vir'. If the current time contains the value 5, then the payload is activated – this tries to work with the Agent Control object. The Merlin wizard appears at this point – no really damaging payload is ever executed. As a result, W97M/Listi.A is a typical *Word 97* SR1-compatible macro virus which manipulates *Office XP*-specific Registry entries and contains a special payload, which only works in newer environments.

W97M/Listi.A

Type:	Polymorphic, anti-heuristic.
Infects:	Documents and templates.
Self-recognition in files:	Check for the special string with the ThisDocument stream.
Payload:	Shows the *Office* agent.
Removal:	Overwrite the infected stream.

VIRUS ANALYSIS 3

Zevota Confidence

Gabor Szappanos
VirusBuster Ltd, Hungary

Word97/Zevota is a slow polymorphic macro virus with mass-mailing capabilities. The virus does not mutate while attacking documents on an already infected *Word* installation, it only changes shape when it mails itself elsewhere. Zevota uses random variable names and most of the constants (even the numeric ones) are encrypted with a key, which is not fixed either and changes with each mutation and code line. The procedures of the virus are also shuffled upon each mutation.

When an infected document is opened, the Document_Open procedure is activated – this is the only non-varying function name Zevota uses. It creates a mutated image of itself, saves it in the default document store directory (often the same as place as the infected file). The name of this document is the user name with a .DOC extension. Then the virus infects NORMAL.DOT, without changing its shape. After that, Zevota mails itself out to, at most, 50 recipients picked from the first *Outlook* address list. The outgoing mail messages have the subject line:

```
{Username} "- Curriculum Vitae"
```

{Username} is the user name as registered in *MS Office*. Furthermore, the virus sends a reply to the last nine messages from the incoming mailbox, with an empty mail and the mutated virus copy attached. Finally, it sets the registry keys Winlogon to http://www.2600.com and Legalnoticetext to 'You Were Infected By FreeCham Virus' in the Windows\CurrentVersion\Winlogon section. As a result, a dialog box with the above caption and text will pop up at subsequent *Windows* logons. Given the location it is stored in, this payload will work only on *Windows 95/98* systems, since on *Windows NT* or *2000* installations this key is stored under the *Windows NT* path (not *Windows*).

After the virus code is inserted into NORMAL.DOT, the virus will infect every *Word* document when it is opened. The Document_Open procedure of the virus will fire up, then Zevota copies its code from NORMAL.DOT to the document without mutation.

Mutation Engine

Zevota utilizes the old variable-name mutating trick, combined with a little twist. Not only do the internally used variable and procedure names (except for Document_Open) change, the numeric and string constants used by the virus are also encoded. This encryption key is not the same for the entire code module, rather it varies with each line, and changes with each mutation.

The virus stores the name of the variables to mutate as a comment line. It is recognized by starting with the ' comment sign and ending with *. The variable names are separated by characters. The line is inserted in a random location in the code module of the infected document.

Once the old variable name pool is collected, the new names are created in seven to eight characters of capital letters from the English alphabet. As the new variable names are generated, the entire code module is processed and the new variable names replace the old ones. One of them is the name of the variable encryption function.

Most of the numeric and character constants are stored in encrypted form, e.g. OISOJQVM("Y{r•j}n", -9) stands for the string "Private" and Val(OISOJQVM("HL", -22)) for the numeric value 65. The first mutation step processes these encrypted constants.

All the above function references to the variable decryptor are parsed from the code module text, the encrypted string and the encryption key are extracted, then a new key is generated. After that, the variable is decrypted with the old key, then encrypted with the new key and the new expression is collected and inserted in place of the old one in the code module.

The virus code is processed line by line, with the encryption key generated only once for a line. Different lines use different encryption keys. While generating the new encryption key the virus checks if the encrypted text would contain the characters " or (. These separator characters would cause serious problems when inserted into the middle of a variable name.

To make the situation more delicate, the virus shuffles its procedures upon each mutation. It collects the procedures into a string array (the procedures are recognized by starting with "Private"). The public declarations on top of the code module (that, according to the VBA syntax, cannot be placed in a different location) are inserted first, then one of the procedures is picked by generating a random number.

This procedure is inserted first into the new code module, then all of the remaining procedures are inserted in order. Therefore, after the mutation, only one procedure changes its place, moving to the beginning of the code module.

Word97/Zevota follows the long line of the variable-name mutating macro viruses, showing somewhat more creativity than the average. The fact that almost all of the variable names are decoded or mutating makes virus analysis and identification rather difficult. It has not been seen in the wild and most probably never will be. However, if this type of polymorphism is utilized or even improved on in other macro viruses, that would cause problems for us.

VIRUS ANALYSIS 4

Warped Logic?

Péter Ször
Symantec Corporation

It is becoming an all too familiar story in this industry of ours. In early April 2001 the news coming from the marketing departments of certain anti-virus vendors was yet again spreading faster than the actual worm to which it referred. A new *Logo* worm had been written and mass-mailed to some of the anti-virus companies by its creator. It never became wild though, and there is definitely more than one reason for that.

Its author happens to be female and she calls herself Gigabyte. Yes, that is right. It is actually written by a female virus writer – this is pretty rare. At least this is the claim made in stories of virus writer meetings published in various places on the Web.

Gigabyte has a background of creating other malware and in particular she authored *MIRC* worms. As we will see, she tried to use her existing *MIRC* knowledge to create the Logo/Logic worm.

The actual worm is created in *Super Logo*, a reincarnation of the old *Logo* language for *Windows* platforms. It is claimed to be 'the Windows platform for kids'! Well, when I was 14, I came across several *Logo* implementations for various 8-bit computers.

I must admit that back then I only dreamed about the graphical capabilities that *Super Logo* provides on modern *Windows* computers. Our 8-bit school computer had a top screen resolution of about 118x72 dots in black and white. Since that no longer constitutes a challenge any more, people try to write a worm in *Super Logo*. Logical, isn't it? Let's see how it was done.

Turtle Torture

The *Logo* language's primary purpose is to provide drawing with a 'Turtle'. The Turtle is the pen and its 'head' can be turned around and instructed to draw. For instance, *Super Logo* uses the following commands: HIDETURTLE, FORWARD, PENUP, PENDOWN, WAIT, etc. The set of commands can be formed as subroutines and saved in a *Logo* project file with an .LGP extension.

The actual project file is a pre-tokenized binary format but the set of commands, as well as variable names, remain easily 'readable' and stored as *Pascal*-style strings. The project file can be loaded and executed with the *Super Logo* interpreter. Furthermore, even the demo version shows the easy-to-understand source of any project files.

Up until now, many European schools teachers have been using *Logo* to teach the programming basics to young students. The original *Logo* language has been very well extended in *Super Logo* to compete with other existing implementations. It can deal with multiple graphical objects at the same time and move them around on screen with complete mouse support.

However, it is easy to find out that the *Super Logo* language supports neither mailing nor embedded executables. Furthermore, it does not support the 'Spawning' of other executables or scripts. Fortunately.

Unfortunately, *Super Logo* does support a PRINTTO 'XYZ' command. XYZ can be a complete path to a file. With that statement a *Logo* program might modify, for example, WINSTART.BAT, overwriting its content with:

```
"@cls
@echo You think Logo worms don't exist?
Think again!".
```

Get the point? When the LOGIC.LGP project is loaded and executed, the worm will draw 'LOGIC' on the screen and then it prints 'Logic, the Logo worm © Gigabyte' to the *Logo* prompt. The project file will be executed by clicking on it once *Super Logo* is active.

The worm will make sure that a STARTUP.VBS file is created in one of the *Windows* startup folders and as such executed automatically the next time when *Windows* is booted. Furthermore, the worm also tries to modify the shortcuts (if any) of some of the common *Windows* applications such as NOTEPAD.EXE to start the VBS file without the need of a reboot.

This VBS file would propagate the 4,175 byte-long LOGIC.LGP worm project file to the first 80 entries of the *Outlook* address book – a pretty standard VBS email propagation. The subject of the email is 'Hey friends!' and the body of the message reads 'Hello! Look at my new SuperLogo program! Isn't it cool?'. The worm, however, has a set of bugs: the actual project file will always be in E:\MIRC\DOWNLOAD\LOGIC.LGP.

On most machines *MIRC* would be more likely to be installed on the C: drive. On the top of that, the VBS- based propagation will fail if the LOGIC.LGP file has not arrived via *MIRC* first. Oh, well.

I can hear you say that this would never work, but the worm actually supports *MIRC* propagation by using the /DCC send command in the SCRIPT.INI of the active *MIRC* directory. The Logic worm only checks for SCRIPT.INI in a few specific locations but it might get it right. The SCRIPT.INI file will have an accurate drive letter for the path of the LOGIC.LGP file.

Assuming that the LOGIC.LGP attachment arrives beforehand via an *MIRC* infection, and that the file is then placed in the MIRC\DOWNLOAD directory, the SCRIPT.INI modification will propagate the LOGIC.LGP file to

anybody in the active IRC channels. Thus, a machine needs to have *MIRC* installed. It needs to be compromised via *MIRC* first.

If all that happens, it is likely to email its project file as long as the *MIRC* directory is on the E: drive, as it was on the worm's creator's machine. So what we can say about this new creation is that it is certainly not an intended worm, but it is a buggy one.

Conclusion

The Logo/Logic worm was an interesting one for me to investigate in many ways. There are several possible endings to this on-going story. It's clear that innocent project files might well become the platforms of tomorrow's worms from one day to the next.

Having said that, corporations might not get hit by a *Super Logo* worm easily (especially if they do not run *Super Logo* in the first place). However, they might well get into the situation whereby they run a set of interpreter-based logic in many hitherto uninfected applications.

As ever, people need to think before they click. That's all very well, but they will not necessarily know all the bad things they should not click on at any given point in time. Just to be sure, let's add yet another extension to our extension list!

Logo/Logic	
Aliases:	Logic worm.
Type:	*MIRC*, VBS worm replication initiated from *Super Logo* project file.
Activation:	Worm displays LOGIC in big letters on each execution.
Removal:	Delete infected files and restore from backups.

OPINION 1

Tests, Tests,Tests – Reviews, Reviews, Reviews

Peter Morley
Network Associates Inc, UK

Ray Glath complained in a recent *Virus Bulletin* article (see November 2000, p12) that anti-virus vendors were giving little attention to virus prevention. He went on to say how they were concentrating their efforts on feeding their scanners with data, and were marketing 'scan, scan, scan'.

Ray is, of course, right. As a guilty party, I have been unable to reach the Holy Grail of UVP (Universal Virus Prevention) because I cannot see how to do it. Nor, I believe, can anyone else.

Viruses and Trojans are still flooding in. Each month, I receive up to 12 collections ('a collection' is those items the sender has dealt with in the last month) from other vendors, and I process them all. If we could reach the UVP nirvana, I would not have to do much work, the authors would get fed up with writing them and go back to sex, users could get on with their normal work, and anti-virus vendors could start finding something useful to do.

Meanwhile, the collections have to be processed. Vendors will tell you that each item they fail to handle is a potential field call. When the number of field calls rise significantly, life becomes sheer misery.

I think we have now reached the point where *all* anti-virus vendors are receiving most of these monthly collections, and there are some common elements in the way they deal with them. Most anti-virus reviewers receive them too, and do little with them other than take a few samples to add to their test suites. If they have to change their test procedures, they do that too, but reluctantly.

This article suggests they make a major change. You can expect a lot of resistance, and maybe even some debate!

The rest of the article covers 3 topics:-

i) What the vendors do.

ii) What the customers want, and what they want to know.

iii) What the reviewers could consider, to satisfy the customers.

What the Vendors Do (or may do)

To process a collection, all vendors first scan it with their latest, up-to-the-minute scanner. They are keen to see how much they already handle correctly, because they can

ignore most of it. They then classify the rest into groups, which have to be processed, and they usually do the easy ones first, followed by the ones they don't detect at all. Finally, the oddments…

That first category (which is already handled correctly) is the key. It will split immediately into 2 sub-categories:-

i) Items which have been processed previously (not of interest).

ii) Those which have not been seen before. These tell you how you are doing with any generic techniques being used.

Most vendors now take the approach that when they get a second variant of a virus they already have, they do a little extra work, to avoid having to do any work at all on the third and subsequent variants. That's what I mean by generic techniques, whether they are called that or not.

When they get a new group of viruses, these should be handled so that further variants are already non-events. In my case, I usually find we already catch over 60%. That 60% was 50% some two and a half years ago, and raising it has been a long hard slog. The 60% will rise slowly, but it may never get to 75%. That's the closest I think we will get to UVP, and it does not include prevention, which is a separate, extremely difficult exercise. As for the viruses not already detected, they have to be processed.

What the Customers Want to Know

At *Virus Bulletin* conferences I get a chance to talk with lots of customers, and (what a surprise!), some of them ask awkward questions, and even make awkward comments.

How about this one, from a fairly large customer: 'I read reviews in *VB*, and find that all vendors detect nearly all the viruses. There are occasional problems, most of which have already been fixed by the time I read about them. But I know that some vendors occasionally fall well behind. Remember Solomon's in 1993. How do I know which ones are falling behind *now*?'.

Or this: 'You continually make a point about generic handling of viruses (at least he had read some of them!), but I never seem to get comments from reviewers. How do I know you *really* detect and repair viruses you've never seen? And even if you do, how do I know it matters?'.

Possible Actions by Reviewers

I am writing this section as a result of a discussion between Igor Muttik of *Network Associates Inc* and Andreas Marx, of the University of Magdeburg. Andreas had already considered these problems and was beginning to take steps to handle them.

I should like these proposals to be considered by *all* reviewers, even if the result is outright rejection, and outraged written comment!

i) Set up a test suite, which will change every time, and which consists of the latest monthly collections from *Sophos*, *Kaspersky Lab*, *Symantec* and *NAI*. Other collections could also be included. But make no attempt to edit these collections, or to remove the rubbish, (even leave in the few OFFVs!) and make it clear that detection failures are not up for discussion. Each time, use *only* the latest collections.

ii) Review the test suite using *two* different editions of the same product from each vendor being reviewed.

   a) Two months old

   b) Four months old

In the case of the two-month old product, I mean not the latest but the one before that. This will give a view of how good the vendors are at processing what they currently get, since items they processed two months ago will appear in other vendors current collections.

In the case of the four-month old product, detection rates will be much lower, and will reflect the ability of the vendors to handle what they have never seen.

Possible Effects of Accepting My Proposals

There will be considerable pressure from vendors who underperform to use the latest product too. If reviewers have to give in to this, it should be an additional test, not a substitute test.

I can see that there might be a lot of discussion about the effect of heuristic options, and whether they should be used at all. Most importantly, these proposals mean a substantial amount of work! However, they will produce useful debate, perhaps not least about the usefulness of the WildList.

The range of test results will be large compared with the range of traditional test results, and may vary, depending on how vendors respond to the new results. Last but not least, the question of whether the tests should replace traditional tests, or be done in addition. In my view, replacement would be preferable, because of the workload involved. I would like to hear your feedback, either direct to me or via the Editor of *Virus Bulletin*.

OPINION 2

A Question of Ethics

Jaak Akker
Sweden
(Jaak_sigsec@hotmail.com)

The *Swedish Information Processing Society's Special Interest Group Security* (*SIG Sec*) is Sweden's biggest association of IT security professionals, numbering approximately 2,500 members.

In 2000, *SIG Sec* appointed a Malicious Code Committee, chaired by the author of this article. The committee consists of eight members – both from public authorities and from various large corporations in Sweden – whose chief vocational responsibility is malware protection.

The purpose of the committee is to exchange information and provide *SIG Sec* with expertise in the malware area. Opening this up to international scrutiny will benefit us, and hopefully you in the long run.

Introducing Ourselves

A recent initiative saw the committee defining an official 'Code of Conduct' that should be viewed as 'good' behaviour in both technical and non-technical efforts to combat malware. Internationally, the malware scene has changed rapidly, but the classical aims of computer security on how to protect information are timeless: availability, integrity, confidentiality. The new challenge to information security in general is non-repudiation in e-business. What is the value of non-repudiation mechanisms if Trojans can compromise the origins of electronic signatures and encryption keys?

Today's counteraction to malware is mostly reactive, and consists of implementing and updating protective software. Considering the explosive spread of some malware, this is unsatisfactory. The knowledge and information about threats and counteractions is limited both within companies and between common citizens.

There appears to be no common code of conduct, neither are there any commonly accepted and continuously communicated definitions, even though the debate among malware specialists reveals great expertise in several of these areas. In public standards malware is mentioned in the BS7799 standard (recently approved as ISO 17799) but a true definition is lacking.

Only a minor part of today's malware is made up of what is traditionally defined as the 'computer virus'. The convergence of malware and hacking is increasing, both in frequency of the quantity of malware identified and the degree of its technical sophistication. Besides worms there are also many espionage programs and malware used for attacking third parties in Denial of Service (DoS) attacks.

Malware spreading is also facilitated by the rising number of common citizens permanently connected to the Internet by xDSL and cable network connections, thereby making them more susceptible to intruders.

About the Code

Our code of conduct is not technical. There is a resemblance to the standard 'safe hex advice'. One has, however, to remember that the malware problem is threefold. The problem is not only about protecting yourself; as a responsible citizen you have to protect your society. This is true with regard to all malware, and especially with regard to malware clients used for DoS attacks, where the malicious software is most cumbersome for the victim.

Last, but not least, the evolution of cyberspace has been so fast that ethical standards are lagging behind. Culprits have always existed and always will. It is, however, not clear what behaviour is ethical in cyberspace. A good example of this is some companies' economical encouragement of malware manufacturing.

This proposed code of conduct not only could but *should* be freely copied, and most importantly the source should be acknowledged. If the text is changed, items may continued to be used but *not* quoted as the original text compiled by the author. This is the best case scenario, and the wish of the author is that the code of conduct should be thoroughly discussed, both in order to clarify the issues therein, and to improve its comprehensibility to readers and users.

Some items in the code may seem to be far removed from today's reality. This might, however, be more of an implication of the complexity of the problem, rather than the level of ambition of the code. The aim is that the code should be short and easy to understand. Thick volumes about malware protection can and should be written. The aim of this code of conduct is, however, to create a baseline for 'common sense' in its literal meaning.

The Code of Conduct

1. **It is everyone's responsibility to counteract the spread of malicious software.**

   Why? Every computer user may be a source of malware spreading. It is not enough to refer to

'somebody' or 'anybody' to take measures. Today's computing designs require a mix of good conduct and active surveillance of malware protection by all computer users in order to minimize the malware problem.

2. **Malware is software that:**

   • harms or degrades computers/data or unwantedly spreads data or

   • lacks the user's consent to spread and start

   Why? A definition is necessary. This is maybe the most critical issue to reach a common understanding on, so improvements on the definition are highly appreciated. This definition is also applicable to unknown backdoors in legal programs.

   A debate is ongoing and law suits have been processed on, for instance, user behaviour patterns automatically reported to service providers. The definition seems applicable to these issues on an ethical level, though most national legislations have not yet addressed this issue.

3. **Everyone must know enough about malware to be able to protect themselves in the situations their computer usage needs.**

   Why? Few people can be malware experts. Everybody running a computer should, however, be knowledgeable enough to specify their functional demands for malware protection. More is expected from network 'techies' than from somebody just using a Word Processor.

4. **Everyone must seek help against malware if you are in a situation you cannot manage.**

   Why? This is to support your conscience when getting in trouble. Seeking help must be viewed as a sign of responsibility, not a sign of stupidity.

5. **Be suspicious of unexpectedly received programs.**

   Why? Elementary safe hex. It is obvious to AV professionals that users' gullibility is one of the explanations of the malware outbreaks that make the big headlines in the media.

6. **Everyone who suspects he has spread malware must notify those he thinks are affected.**

   Why? You should not cry wolf, but telling your suspicions to others is the only responsible way to act when encountering malware problems.

7. **Everyone must use a protective program and keep it updated.**

   Why? Haven't we all heard this one –
   Help desk: 'Uhuhh. You say you suspect you have a computer virus, though you had virus protection installed when you acquired your PC a year ago. But when did you last update it?'
   User: 'Update???'

8. **The transfer of programs may not be performed without the consent of all the parties involved.**

   Why? An increasing problem is the automatic, unnotified installation of spyware when accessing Internet pages. It must be considered unethical to install software without the user's consent. (If you haven't encountered this yet, run a spyware search program on your PC. You might be surprised what you find.)

9. **In- and out-going messages must be screened for malware.**

   Why? Many reports indicate that email (and maybe, in the future, computer-to-computer messages) is the largest channel for spreading malware.

10. **The computer must be restored to its original state after a malware attack.**

    Why? Many users are unaware that several items of malware manipulate the security settings in the software.

11. **The exchange of information must be performed in a way that prevents spread of malware.**

    Why? This may be the toughest one to implement, considering today's software design. How many users are aware that word-processing documents and spreadsheets are actually programs? As an example, when data and programs are separated in different files in a way which is comprehensible to the user, malware can be combated efficiently.

12. **Thou shalt not make malware.**

    Why? It is the only possible way to go if you consider yourself to be responsible. If you are a conscientious researcher, you can test parts of virus behaviour. Creating complete malware with payload and spreading mechanisms can never, ever be acceptable!

Conclusion

It makes common sense for every single computer user to take a stand on the issue of ethical behaviour. It benefits everyone. Having shown you what we have come up with here in Sweden at the *Swedish Information Processing Society's Special Interest Group Security*, I would appreciate any and all feedback that the readers of *Virus Bulletin* can provide. Contact me at the email address on p.13 or get in touch with the Editor with any comments or suggestions you may have.

In the absence of an internationally respected code of conduct as regards the issue of malware and computer security, we consider this to be a sensible starting block. It would carry even more weight should respected members of the anti-virus industry get involved with its creation and implementation. A summary of the above points would be a sensible way of beginning to educate your office or your users.

FEATURE

Playing Home and Away with AV Software

David Phillips
Open University, UK

No, this is not about an Aussie soap opera, although the way they all arrived at once, all over the place, they could be classified as viruses! It's not even about football (soccer for those outside the UK). No, this is about how setting up an anti-virus system in the work place can give you a false sense of success when what you're actually doing is rolling out a package to 60,000 home users.

Setting the Scene

Over 19 months ago we changed our anti-virus package at the *Open University* (see my VB'99 paper for details). Having made our selection, our preferred choice was rolled out to the site with great success. It enabled us to update across campus and the regional offices with a couple of clicks of the mouse – most of the time without hassle. Furthermore, our users are protected, no matter where they are – on campus or in one of our many regional offices.

We had another product on our conference system for our students to use for over five years, but the user licence came to an end in September 2000. We tried for a deal with the developers (one of the biggest in the business) after promises the year before that they would support us, but after three months I had to go elsewhere, with their sales team saying 'Oh I didn't realise there was a time limit'!

So, I investigated other AV software and over the three months waiting for our old suppliers, I chose another program from an overseas developer. Over December I did tests on different platforms, *Windows 95*/*98*/*ME*/*NT.4* and *2000* with good results. We also had the added bonus in the UK of a good report from *Which?* magazine for this particular home user product.

Moreover, I was able to download the software via a modem to install on the platforms, and use the update facility with success. This was a major requirement as we deal with students who never actually come on campus. To me, it looked like the package for the students from 2000 to 2003 (our three-year contract) would be a simple system to install and administer.

Just to be sure I had it tested by a couple of work colleagues and some Associate Lecturers. Any problems? Not on testing. I had to create my own installation documents, not mentioning specifics but the foreign translation was far from helpful! Mind you, I would have done so anyway. Some of our students are beginners in computing and the documents we create for our first level courses are based around 'hand-holding'. Yes, there are still beginners in our vast IT industry – and it's for them that we write a lot of our own installation documents.

So, January 2001 duly arrived and after the comprehensive tests carried out over the previous month, I felt that we could roll out the new software to the students starting to study this year. Was my confidence misplaced!

Whether you roll out across a campus or across a company, there is a good chance that a lot of the machines are running desktops which the company has decided on, and the software which they are supposed to have installed plus the odd extra. Nothing there is out of the ordinary – except the odd machine. In other words, the desktop you test on is pretty well guaranteed to be the same you roll out to.

Our students' machines are different in every way possible. One thing I keep noticing is that a lot of new users who have just bought their PCs from the superstore down the road don't know what is installed on it. The store doesn't explain the machine properly, and I find a number of users do not know their machines come with an anti-virus package at all.

And what is more, these kinds of users definitely do not know how to update the anti-virus software. Is this the problem of the store or the AV industry, which allows their software to be packaged on new PCs without proper documentation and support?

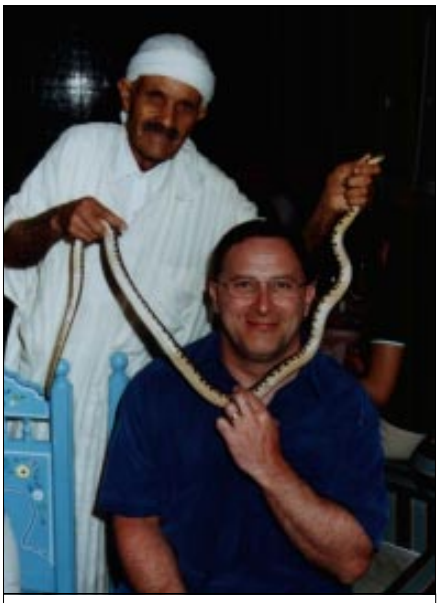
I should explain here that we were offering to roll out the particular vendor's version of anti-virus software which has the option of heuristic scanning. Need I continue? That's right, the first problem that showed up as the students downloaded the software was a suspicious file – belonging to *NTL*, an ISP company in the UK. Of course, that meant they were not able to dial out and had to disable the software to get advice.

It took two weeks to get the fix to stop the false alarm but I have found that telling the students to disable the heuristic scanner removes a few problems like this.

I feel heuristic scanning is still not a tried and tested option for virus detection, the false alarms that have shown up during the past few months have proved that turning the option off is the best way for home users to go.

Another Problem, Other AV Software

Since some of the students did not know they were running anti-virus software already, they ended up putting new AV software on top of the installed package. This should not be a major problem as you can have more than one AV package



AV roll-outs? It's a jungle out there!

on a machine – most of us in our line of work do – but they had their copy and the new software memory resident.

Now, that does cause problems, especially as some of the AV software that is pre-installed on a new machine does not have a simple 'remove' button for the user if they decide that they want a different package. That reminds me of *Internet Explorer* – why do some pre-installed packages not have the correct uninstall options available?

Don't Believe the Ads!

If *Microsoft Windows* is unstable, install 'Jim'll Fix It' to save your data or software when *Windows* crashes – so the ads tend to say. So, gullible users buy the latest software for protecting the system and data, running in memory. I know this could surprise some sceptics out there but if you don't play with the settings on your machine too often, it is remarkable how stable a *WinTel* box can be.

I know that might worry some of the companies out there trying to sell the 'complete' package but I have a number of users who follow my advice and they have not lost data. Nor have I had to rebuild any machines and I have not needed any 'extra' software either. One student had a package installed that had eight separate modules in memory; it goes without saying that when he removed the package, his PC became stable. Wasn't that the point of the package?

I think that the point one-trap vendors are missing is that, in the UK, a lot of the 'off the shelf' packages only come with 64MB RAM with optional upgrade. If you don't take the option you have a lot of packages all trying to use this memory on start-up and leaving very little for the user to work with.

What's the Point?

My mistake was that I got complacent about rolling out anti-virus software to home students after the easy operation we conducted with on-site updates on campus. I also forgot about the political and technical hassles I had when I first issued our AV software back in 1996. I came down to earth with a bump with lots of emails and calls about the new package. I can say that after three months, the problems and questions have reduced to a few a day – quite a few with the old problem; install first, then read the manual.

The main problem we had was not with the software as such, but with all the different flavours the home user has of the same thing. You can never tell where they bought their machine from, was it from a superstore or the guy down the road that puts bits together? You can't tell what packages they have, and with the growth of hard disks how much 'junk', sorry, 'useful' packages have been installed, which leads to problems and help-desk heart failures.

Conclusion

Here's the moral of my story: if you ever find yourself shipping anti-virus products to vast numbers of home users, be prepared that they don't have what you think they have, and in some cases that they don't have what *they* think they have.

As for the *Open University* – has our new anti-virus decision been the wrong one? Judging by the number of students who have downloaded the software and reported viruses detected – no! The problems I keep seeing concern only a small percentage of the happy students, but I found that I had fallen into the complacency trap and had a rude awakening.

Is the problem that we are trying to pre-package too much on users' machines without fully understanding the consequences for new, inexperienced users? Are anti-virus vendors dictating what they have to sell to their users rather than what their users want to buy?

So beware; installing in corporate/campus environments – having some control of the software on the desktop is a breeze compared to the cornucopia of machines the home users can come up with.

Finally, I have posed some questions in this article that highlight possible causes of problems and it would be interesting to hear other users' thoughts on them and the anti-virus vendors' responses.

I should note that although I keep referring to the vast amount of software pre-installed on new computers, AV software is only a small part, and probably one of the really necessary packages. The Internet may be one of the main selling sources of new machines, but with vendors of other packages bundling onto the new PC, shouldn't the anti-virus vendors take the lead on explaining the possible pitfalls to the user and offer better documentation?

[*Dave wanted me to mention names – both that of the company which he feels let him down with their AV software and the company whose software the* OU *decided to opt for. We've always felt that this kind of feature wouldn't really benefit from naming names. There's a lesson to be learnt here, no matter whose software you're using! Ed.*]

PRODUCT REVIEW 1

CA InoculateIT v6.0

Matt Ham

Computer Associates (*CA*) is a vast company, so large that it markets a pair of anti-virus products, having acquired another Australian-based anti-virus company in its policy of growth by acquisition. Other activities of the company are diverse, covering security, development tools, administration applications and more. With *InoculateIT* regarded as part of *CA's* product range in a more integrated fashion than is often the case, the management aspect of the company's anti-virus arm has long been one of the more comprehensive on the market.

There is also a surprising blurring of distinction between this product and the other flagship *CA* anti-virus solution, *CA Vet Antivirus*. The engine used in *InoculateIT* in its default mode claims to be exclusive to *InoculateIT*, but is also included and available for use in the *Vet* engine – making this a product with the possibility of performing a Comparative Review against itself.

The Package

The box for *InoculateIT* has seen little change in this latest version – it is still a muddy brown affair but with the minor alteration of the new corporate logo. Despite quite a bulky box the contents are slimline – a manual, a letter-sized sheet of paper advertising *eTrust Security* solutions and the CD in its cardboard wallet. *eTrust* is a blanket name for the slightly less than full range of *CA's* various products, specifically directed towards the great buzzword that is e-business. The CD wallet also contains two CD-sized chunks of paper with further information dedicated to the registration and licensing of the software.

Documentation and Resources

The manual contains all the practical printed reference material and was subjected to appropriate scrutiny. It has clearly been revamped for the new version, a relief when so many manuals are supplied out of date. Oddly, it also stars a cartoon-style chap called 'Bud' – an addition out of place in what is essentially a technical document for administrator use. Singularly lacking was any reference to Web site information, or indeed an immediately obvious URL for database updates, alerts or patch information.

The URL is, in fact, present, though hidden in one of the Bud hints boxes in the centre of the manual. Other information in the manual is, overall, good and up to date, and covers installation across network and sites in a comprehensive manner. There are, however, slight problems with the final Glossary which is still living in a world where DOS viruses are the only peril of note.

The *CA* Web site, once reached, contains a great deal of information, though it is not well advertised. In the first place, there seems to be no intuitive link from the main page to anything but product descriptions. However, using the manual-provided URL, a more virus specific area can be reached via the http://www.esupport.ca.com site. Despite simply being labelled as Virus Signature Updates there are also a large number of virus-related resources available when the link is followed.

Unfortunately, as is the case with the aforementioned manual's Glossary, not all of this information is currently relevant, and much exudes an aura of cobwebs accumulated over years of stagnation. Historians might be interested in, for example, a very early copy of the Alt.comp.virus FAQ, but the information within this mid-90s version is woefully inadequate now. There is also a pleasant reference to *Virus Bulletin* tests being 'potentially incomplete and biased' and thus there being 'no need to consider' them, enabling me to be as harsh as possible without repercussions!

Installation and Upgrade

The CD autoruns on insertion to reveal a brief splashscreen which quickly vanishes to be replaced by the main installation screen. This is, in itself, rather more involved than the complete installation and operation interface for many products. Options here are the installation of Advanced, Workgroup or Client versions of the software and the viewing of documentation, both general and product specific. Information includes the system requirements for the various products and remote installation is supported in addition to local.

Installation of the Client version was selected for installation onto a test laptop – while the more sizeable versions were reserved for higher specification machines. Most descriptions in this review refer to the Client version, unless

otherwise noted. After the general licence and user information have been passed, the choice of custom or standard setup is offered – custom setup allowing the addition of a Netware Domain Manager module. After this, installation proceeds to the selected directory – with the majority of installed files seeming to be help files – and the option to create a rescue disk is offered.

Registration is performed in a slightly non-standard form since the *CA* products share a common registration application – RegisterIT. This is described on one of the scraps of paper in the CD case, remarkable only in the admonition to presumably very forgetful users to 'have the following information available before you proceed: Your name…'.

Updating is performed within the *InoculateIT* program and can be performed for the Client version in default mode by the push of a button if an Internet connection is available. Further tweaking is available which enables updating from UNC, ftp, local or dedicated server sources and the times and time intervals at which these should be attempted. For more paranoid administrators more than one option can be set, which might ease the updating of unpredictably LAN-docked or Internet connected laptops.

Features

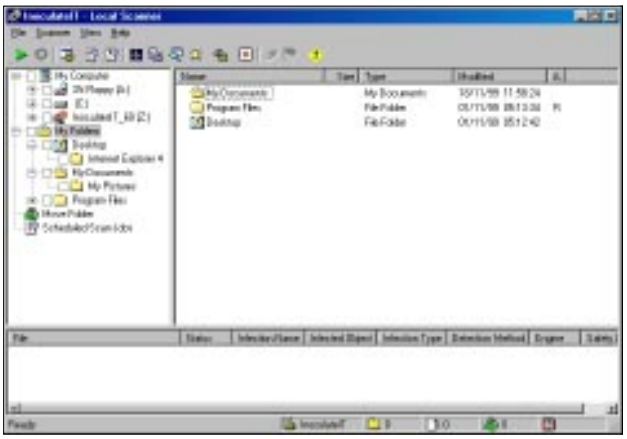
In what is quickly approaching an industry standard interface, *InoculateIT's* scanning is initiated within an *Explorer*-style GUI, though with an added panel for the display of scanning information. This toggles with a somewhat less than standard Log Viewer interface.

The Scanning Interface

Control of the scanning process is initiated from two places, the drop-down menu bar and icon bars above the central display area. The drop-down menus cover File, Scanner, View and Help, while the menu bar has more initial options but in considerably less depth.

The File menu consists of the limited options of setting up printing, or exiting the GUI and can thus be skipped over quickly. View is another area where brevity can be justified – simply toggling between the Scanner view as described here and the Log view as described later.

Somewhat more complex is the Help menu, which allows links into the Contents and Procedures help areas as well as the 'About InoculateIT' area. The two help areas are loaded not as the standard *Windows* Help files but as HTML – possibly intended for the use of the help files in a cross-platform corporate environment. Information is liberally hyperlinked, though some links end in pop-up boxes and others in new pages, which can lead to slight irritation if many links are selected. The information provided within the 'About' area is of twice as much interest as that usually seen, containing as it does information on both the *InoculateIT* and *Vet* engine version numbers, and an indication that *InoculateIT* is the selected engine.



This leaves the Scanner menu – the one with by far the most options available, and where the bulk of configuration choices will usually be made. It is divided into six subsections, chosen by the area of control involved. The first and one of the most simple sections contains the start and stop scan controls, while the second allows the launching of the rescue disk creation wizard.

The third subdivision is used for setting options, covering Local Scanner, Realtime Monitor, Signature Update, Contact and Alerts. Local scanner options are, as might be expected, many and varied. Selecting this option brings up a tabbed GUI which contains these further choices. Even here there are still sub-menus available from buttons on the tabs, though in the description these will be considered as part of the home tabbed page. The Scan tab allows the setting of a safety level – Secure or Reviewer, the latter being rather more paranoid. It is also here that the scanning engine may be set – *InoculateIT* or *Vet* – heuristics enabled, incremental scanning selected and scanning of alternate data streams set as an action. The last is off by default and is present for the purposes of any future malware making use of this data storage method.

Finally in this tab comes the selection of action to take on detecting a virus in files. This is relatively standard when most actions are considered, with the exception of quarantine being termed 'Move' and the options available when curing of files is attempted. Here Registry modification, dropped file deletion and the like are available as options. Trojans may be flagged for automatic deletion, backup files created before cures, all macros deleted from infected OLE files and actions selected if a cure fails. This is indeed an impressive selection of options, and should satisfy even the most hardened software configuration tweakers, though more are available on the remaining tabs.

The selection tab offers a choice of objects to scan – boot sector, memory or files. For files, all are scanned by default, though an extension list to be scanned or not scanned can also be implemented. Compressed files are also scanned by default and options here determine how archive file format is detected, which formats are scanned, which extensions should be associated with compressed files and the treatment of infections and extension list scanning filters within

compressed files. Display tab options are somewhat less complex, covering which type of media are scanned and how the results are displayed on screen.

The Directory tab controls locations of home, engine, log, and move directories for the program together with the extension to be used if infected files are renamed. The final Scanner Options tab is dedicated to which events are added to the scan log files, and contains options for clean files, infected files and skipped files. Since these are individually selectable, a log providing missed and skipped files can be produced, which makes the review process an easier one.

The Local scanner options having been inspected, next on the list are the Realtime Monitor Options, again controlled from a tabbed set of menus. The subject of some of this control is identical to that for the Local scanner, safety level, action on infection, compressed file handling, extensions to scan and selection of scanning engine falling under this category. This last option is particularly interesting, as it gives the choice of running on-demand scans with one engine and on-access scans with another all from one integrated interface.

Other Realtime scanner options are somewhat different, and include selecting which activities are scanned and setting excluded directories for this scanning. There is also the ability to 'blanket deny' access to certain files, generally defined by extension but certain files can be exempted from this. This is helpful in those situations where gateway content management is not appropriate but individual machines require some form of control.

It is, in addition, possible to set whether floppy, network or CD-ROM drives are scanned on-access, with CD-Rom being defaulted to off. Floppy drive scanning on shutdown is supported and as the producers of *ARCServe* it is also not surprising that *CA* provides support for exempting tape backup procedures from scanning. A final tab here gives a statistical breakdown of the scanning performed.

Signature Update Options are next, which have been described already, leaving Contact Information Options as the next stop. These allow virus samples to be sent to a designated mail address, by default at *Computer Associates* but this can be set to an address within the company. Mails sent contain company details and contact information within the organization. In order that these sendings be triggered, however, it seems that the administrator components need to be installed – these are not available in the *Windows 9x* and *ME* versions.

Linked to this location in general subject matter is the Alert options area. This, again, requires additional components for all the functionality to be complete, though even as it stands there is an array of conditions available for notification related to either severity or custom event-based filters, and reports may be set to queue, time out or skipped if sufficiently old. As it stands, reports may also be forwarded – the further options of reporting to an event log and/or local alert manager were not available on the machines primarily tested.

Having covered the Options areas of the Scanner menu, the remaining controls available become less vast in their scope. Next on the menu are areas to clear the Output of the last scan, and also to show the summary of it. Following this is another section devoted to the manipulation of objects in the Move Directory, as mentioned before. Objects within this directory can be either replaced in their original position, replaced with a rename, or replaced after having been disinfected.

Arriving at the last item in this menu we reach the Scheduled Scan Job control area – with sub-sections devoted to Create job, Options, Statistics and Stop job. Creating a scheduled job allows much the same level of control to be exerted as an on-demand job, including the ability to specify a different scanning engine. A notable if expected addition is the scheduling tool, which allows slightly more control over intervals than most such applications, while lacking the ability automatically to select specific days of the week as targets.

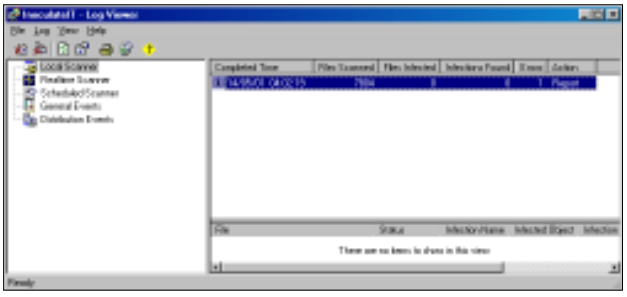
Scans may also be set for startup of the machine, with targets and exclusions also customisable. One feature notable and mourned by its absence is the ability to browse for directories to be scanned or excluded, in both these cases the path must be typed in rather than selected. The drop-down menu items form a superset of the items on the icon bar which holds no great surprises.

It was mentioned previously that the left hand part of the screen contains an *Explorer*-style directory tree, though there are some extra entries here – namely Move Folder and Scheduled Scan Jobs. These two additions are used as shortcuts for the administration of quarantined files and editing of scan jobs respectively.

The Log Interface

The log interface shares the split screen design of the Scanning interface and provides log information divided amongst Local, Realtime and Scheduled scanners and General and Distribution events. The information here is stored as numerous discrete logs rather than one, which makes browsing much more convenient.

In the *NT* and *Windows 2000* versions of the product, additional options become available for administrators,



who might be expected to make use of such higher security platform for their duties. These features are primarily related to the handling of large numbers of installations, by permitting organization into smaller units which may be managed more easily.

The Administration view is integrated into the main scanning application as a third view in addition to those already mentioned. Again it exists as a tree structure, which in this case has its root in the logged-in administrator. Branches allow the setting of configuration policies, managing of domains containing outdated *InoculateIT* software and the organization tree for adminstration.

Policies can be designed and imposed on machines or organizational units and have five categories – the Alert, Realtime, Signature Distribution and Scheduled policies exactly match the options of the same name configurable on a Client-only installation. The new addition is the Send for analysis policy, briefly mentioned earlier as a feature not available on the Client-only *Windows 98* and *ME* versions used for the majority of testing.

The level of control able to be exercised here approaches the completeness of the scanning controls but a full treatment is somewhat outside the scope of this review. Remote installation is also supported, though in this case from a standalone application triggered from the start menu. This lack of integration comes as a slight surprise though possibly allows for the alternative use of *Computer Associates* Unicenter TNG software management system. As is to be expected, the standalone remote installer can be set for pre-configured and automated roll-out and installation settings saved to file for future use on alternative targets.

Scanning

Since it was available, the production of log files listing skipped and clean files was used as the method of determining missed files in these tests. A standard default installation check was the first step – and was quickly followed by the same scan performed using the minor variation of a different scanning engine, *Vet* being the *de rigeur* choice.

Once the tests were started, the first scan of the test-sets showed an impressive detection rate – of all the samples in the test-set only 21 samples of W95/SK.8044 were missed. With the *Vet* engine this scan was marginally less impressive, misses being recorded on two W95/SK.8044, two W95/SK.7972, all ACG.A and ACG.B samples and 16 macro samples covering 4 viruses. Changing both these scans to Reviewer rather than the default of Secure Scan had no effect on detection rates.

Enabling heuristics also had no effect upon the results of the *InoculateIT* engine scan, but added detection for all the macro viruses previously missed by the *Vet* engine. A problem did come to light at this point with slight stability issues – after changing to the *Vet* engine and performing a scan of the *VB* test-sets, the GUI thereafter failed to respond to inputs. Smaller test sets did not result in this problem, and shutting down the application by means of Alt-Ctrl-Del appeared to solve the lock-up with no side effects.

As for the relative speeds of these scans, this was tested on the *VB* Clean and Macro sets, as used in Comparative testing. Results here were slightly confusing in parts. On the Clean set, containing executable files, *InoculateIT* was fastest on its default setting (340s), with heuristics (350s) being a very slightly slowing influence and the Reviewer scan option (410s) adding some 20% to run times. The Reviewer scan also added two false positives. So far, so predictable, but changing to the *Vet* engine caused a little confusion. In this case the Secure (390s) and Secure with heuristics (390s) scans were about 15% slower than the equivalent *InoculateIT* scans but the Reviewer scan (350s) was considerably faster.

Scans across the Macro test-set were impressive, though showed little variation other than *InoculateIT* being the faster of the two engines. The odd timings, combined with the instability mentioned before and the slower *Vet* times, seem to indicate that the integration of the *Vet* engine, while good in most cases is slightly less than perfect.

Conclusion

The *InoculateIT* scanner is interesting and notable for two main features. Firstly, the level of control offered is impressive, with the presumed target audience being the zealous administrator, since it is not just attractive but does, in fact, result in the addition of functionality. The second area of note is the integration of both *InoculateIT* and *Vet* engines in a product simply calling itself *InoculateIT*. This latter is definitely useful for those situations where a second opinion is required concerning a possible infection.

It also begs the question of whether *Vet Antivirus* is approaching a change in its state. Several scenarios present themselves – the *Vet* name remains unchanged, *Vet* is swallowed by the *InoculateIT* product line, or *Vet* integrates the *InoculateIT* engine within itself to form another dual engine-capable product. Whatever the changes caused as a result of the design decision, however, *InoculateIT* remains well implemented from the dual points of view of detection ability and breadth of control over the scanning process.

Technical Details

Product: *Computer Associates InoculateIT v6.0.*
Developer: Computer Associates, 1 Computer Associates Plaza, Islandia, New York 11749, USA; Tel +1 800 2255224; fax +1 631 3426863; WWW http://www.ca.com/.
Price: Contact *CA* for pricing details.
Test Environment: Three 750 MHz AMD Duron workstations with 128 MB RAM, 8 and 4 GB dual hard disks, CD-ROM, LS120 and 3.5-inch floppy, running *Windows ME and 2000*. Pentium laptop with 48 MB RAM, 1.4 GG hard disk, CD-ROM and 3.5-inch floppy running *Windows 98*.
Virus Test-sets: Complete listings of the test-sets used are at http://www.virusbtn.com/Comparatives/NT/2001/02test_sets.html.

PRODUCT REVIEW 2

McAfee VirusScan v4.5.1

Matt Ham

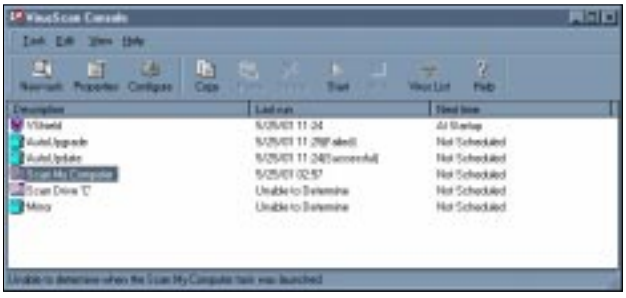
A large company with a high profile in the anti-virus field, *Network Associates Inc* (*NAI*) needs no great introduction, and the name is a good clue as to where the product strengths of the company lie. Such products as *Sniffer* and the *McAfee* and *Dr Solomon's* lines have gained the company a great deal of public recognition. The most common question about *NAI* is, in my experience, simply how it relates to the *McAfee* and *VirusScan* product lines. The product reviewed here is produced by *NAI* but badged as a *McAfee* product, but the name is less relevant to our purposes than what it is and does.

The Package

VirusScan was supplied as part of the *Total Virus Defense* suite, though the box also referred to *McAfee Active Virus Defense* and the product family of that name. The box arrived slightly crushed and was, as ever, of a new design – the *NAI* design team seem to work on overdrive as far as box art is concerned – which in this case was emblazoned with a rusty bio-hazard warning symbol. Inside, however, the contents remained much as ever in bulk and nature. The CD of software is the most important part of the package, to which are added five softcover manuals and a similar number of much smaller documents.

The manuals are all 'Getting Started' Guides, and are devoted to *GroupShield Exchange, GroupShield Domino, NetShield for NetWare, NetShield for Windows NT and Windows 2000* and the reviewee – *McAfee VirusScan*. When referring to the manuals as softcover the emphasis is here definitely on the soft, the manuals being made of relatively flimsy paper, which has in the past led to them quickly becoming tattered. The manuals and CD supplied were already out of date since version 4.5 has recently been updated, and for the purposes of testing the newer version was used.

Other than the outdatedness, the manuals are the usual useful fare from *NAI*, though across the whole range there are few redundant chapters which might have been better condensed into a general overview. This is especially true since the final chapter in each manual is devoted to support options and the first to contact details, which are more or less covered in the accompanying smaller documents. These consist of a multi-language support options pamphlet, a support certificate with authentication code for electronic support, a further document on support options, and a registration card containing yet more information on the same support.



The thus vaunted support provided with the product itself consists of access to the support site, query submission by email with a two business day response time and electronic versions of software updates. Optional extras as far as support are concerned can be expanded with telephone access, assigned engineers and contacts for proactive planning, priority handling of calls and theft protection plans. All of these bonus options do, however, come with an expected charge for the service.

As it is an included option, the Web support was inspected for overall usefulness. There was a degree of confusion at this point, as the URL supplied for support registration did not exist, and only that for product registration could be discovered after an extensive browsing session. Electronic registration was a long-winded process, with a requirement to input two strings of characters from the box label, a grant number and a large amount of personal data – most of which was required information.

Installation, Update and Upgrade

Among the new features in the latest version of *VirusScan* is the ability to remove previous installations of software from competitors, using a variety of described methods including triggering the standard competitor uninstallation programs and the more dubious use of custom installation options. The custom uninstall option is a possible cause of worry, given that the competing developer's uninstalls do not always work despite being written by folk with, one hopes, greater internal knowledge of the products involved. Since this is an option applied only by a patch for the scanner it was not put to the test on this occasion.

Installation of all products mentioned as having manuals is supported from a single opening menu auto-started on the CD. Also available for installation at this point are versions of *WebShield, Management Edition, SuperDAT Utility* and an *Installation Designer*. Of these extra and mysterious applications *WebShield* is a proxy or SMTP-based virus detection application, while the other three are used in the management and installation of various anti-virus products, across a network. Although hard copy documentation is not available for these, there are PDF and text documentation files available and accessible from the CD front-end.

There are also links provided to the *McAfee* homepage and beta program areas in addition to the *AVERT* (Anti-Virus Emergency Response Team) Web pages. This last is a useful data source for most virus-related queries. The *VirusScan v4.5.0* area contains the product installation link, PDF versions of documentation and the v4.5.0 Service Pack. It appears that the SP must be installed as well as the product, making this a two-stage process at the very least.

Triggering the base installation passes through a product introduction and licence agreement, before heading to the choice of a typical or custom installation. Custom installation options are extensive and displayed as a tree structure, with available options for each component to run from hard drive, CD, or not at all – alternatively they can be installed as and when required.

Options selectable here (default settings) are the main on-demand scanner, right-click scans, scheduling, on-access scan, alert manager, send virus utility, emergency disk creator and command-line scanner plus the amusingly named *McUpdate* utility. Options not installed by default are email scanning, Internet scanning and scanning when screensaver is active. For the purposes of the review all options were turned on.

The installation process then completes quickly and configuration is offered along with an option to scan the memory before it is attempted. Configuration options here are whether to boot scan at startup, create emergency disks and/or run a default scan directly following installation. Also offered are several autoupdate features which were on this occasion skipped since the Service Pack and upgrade to v4.5.1 were to be applied first.

The former was a simple click and execute action, while the latter required unzipping first, but was thereafter simple to perform. More optional features are added to the Custom installation tree, namely a console branch into which the *McUpdate* option is inserted, neither of these upgrade options requires a reboot, which came as a pleasant surprise when performing multiple installations.

The next stage in the procedure was the testing of updating virus definitions, which were performed in two different fashions. First was the download method, the .dat file being downloaded as an executable and run locally. This proved to be a simple operation, though finding the correct files to download among the many available was more of a trial both for updates and the upgrades previously discussed.

The second method of update (and upgrades) testing may be performed is within the program console. The console is described later in the review – at this point only the AutoUpdate and AutoUpgrade features are considered. To the accompaniment of a throbbing petrol pump, the update progressed smoothly when triggered, though the upgrade is not supplied with a default location, presumably for licensing reasons, and thus required more tweaking to make it operational.

Features

The *VirusScan* Installation adds four applications to the start menu under *'Network Associates'*. These are Create Emergency Disk, VirusScan Alerting Configuration, VirusScan console and VirusScan itself. The emergency disk is self-explanatory, though the offer to use *NAI-OS* for formatting shows that some considerable effort has been put into the procedure.

The console, on the other hand, is more complex and the seat of most configuration options for the scanner. This is an *Explorer* style of interface at first glance but only has one main window – clicking on any object in this window will either trigger an activity or bring up a tabbed dialog for control of another activity. The options available after a full installation on a machine with one partition are VShield, AutoUpgrade, AutoUpdate, Scan my Computer, Scan Drive 'C' and Mirror.

'Vshield' controls the on-access scanner of that name by means of a four-tabbed Properties dialog. Tabs are labelled System Scan, E-mail Scan, Download Scan and Internet filter, and the Scan tabs as their main contents display statistics of files Scanned, Infected, Cleaned, Deleted and Moved. For the Internet files tab this is slightly changed with Java applets, ActiveX controls and Internet sites being noted as being scanned and/or banned. Each tab also has an option to enable or disable that particular function and a button which triggers that task's configuration.

Each of the Property buttons leads to the same dialog – a mini console with tabbed boxes individually available for each of the scans and filters and an additional area where security is configured. On-access System Scanning is probably the most important of these areas in most organizations, and allows scanning of files to be triggered when inbound, outbound, neither or both and floppies to be scanned on access, shutdown, neither or both.

Scan targets are selectable between Default files, All files and User specified file extensions. As a new feature in the v4.5.1 upgrade the default extension list is configurable from within the virus database update files and thus the previous 'all files scanned' default is no longer used.

Compressed files and network drives can also be flagged for scanning on this tab. Since this program is designed for corporate use there is the further option of disallowing disabling of the on-access scanner and the icon for that component in the task bar may be disabled if undesired.

A further level of defining the on-access tasks is available with the Advanced button on this tab, which is used to enable heuristic scanning. If enabled, this can be targeted to macro files, program files or both. Whether or not program files include script worms and viruses is unclear from the help description provided. This is one of the few places in this set of dialogs where a help button is present – though help is not lacking, since right-clicking on any point of a dialog brings up a help box for that function.

The system scan actions may also be further defined, across the usual range of clean, delete, move, and block access, with the less common addition of allowing a user to exclude a file from future scans. This last is presumably not designed for the average user, but it is defined as one of the standard default choices to be given to users upon detection of a suspect file.

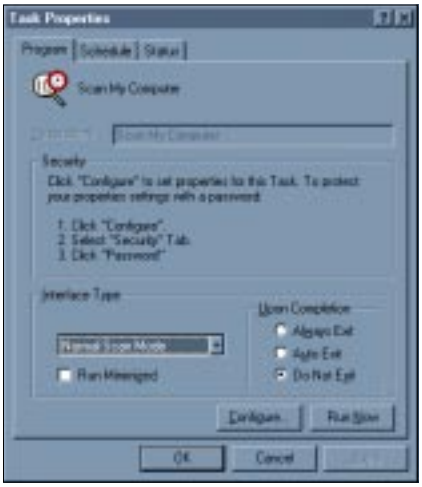
Individual drives, folders, subdirectories and files can in any case be preconfigured as excluded from scanning. Alert boxes can be selected as being foreground only or modal full screen affairs and the alerts generated can be sent to an Alert manager if desired, with messages and simple audible alerts also configurable.

The last option for the System Scan is the report produced, which can be forced to log all manner of activities both user and infection related. Custom report names are configurable and a size limit may be imposed upon the report file if required. Report files for each of the components mentioned here are configured and controlled in the same way, but independent of one another, as are Alerts.

The other on-access scan properties tabs are less vital in most corporate environments but still worthy of an overview. Mail scanning offers control over which attachments should be scanned, allowing extension lists to be prepared much as in the standard on-access scanning already described. Scanning is offered for traditional email using the MAPI protocol and the so-called internet email services provided by the likes of hotmail.com.

Download scan also acts as a normal on-access scan, though integrated with POP3 or SMTP email clients, while Internet filtering is concentrated upon ActiveX, Java and blocking URLs and IP addresses. The security area allows the locking of any or all of these individual control areas from those not having the correct passwords.

Next in the console are the AutoUpdate and AutoUpgrade functions mentioned earlier. These can both be set to attempt to perform their activities from either UNC, ftp or local paths and schedules can be set for these actions. Ftp logins with password and user data or as an anonymous connection are catered for and the software is proxy-aware.



These sections are aided by and linked to the Mirror option, used to automate the production of a mirror of the *NAI* ftp site on a more convenient machine and to do so automatically as a scheduled mirroring task.

The final options here are 'scan tasks', which can be added to by using the console's drop-down menu or icon bar. Upon creation of a new scan task it can either be added to the list of tasks on the console, scheduled or not, or triggered immediately, though in either case the *VirusScan* program is spawned as a result of activating the scan.

Control over the scan tasks is very similar to those options available for the on-access scan configuration with the addition of a more comprehensive target selection for determining which files are to be scanned. The *VirusScan* program itself is much less involved, allowing no directly obvious control of settings bar target, actions to be taken and reports to be made. Advanced options are available through the tools menu within this program and here follow the configuration options available elsewhere.

The final program in the installation is the Alert Manager component, which enables alerts to be transferred to centralised or DMI-based alert management systems if required. Configurability is minimal here as *VirusScan* itself and the further applications are those which do the more complex filtering.

Conclusion

The discussion of documentation and Web options for the product may have seemed to paint *NAI* in a bad light, though the problems there were amost entirely caused by the lag between producing a product and the printed and CD medium catching up with the most recent versions. The new features in v4.5.1 are extensive for what is only a minor version number change and the mirroring option in particular is a useful addition to the arsenal of administration features.

There is a move towards making the actual scan program more simple in its configuration by moving the configuration options away from the application which performs the scanning. Whether this will be taken to its ultimate conclusion of using scan jobs rather than triggering the engine and then scanning remains to be seen, as this might be seen as too radical for the retail sales which are still very important to the *McAfee* product line, if not *NAI* in general.

Technical Details

Product: *McAfee VirusScan v.4.5.1*

Developer: *Network Associates Inc*, 3965 Freedom Circle, Santa Clara, CA 95054, USA ; Tel +1 888 8478766; WWW http://www.mcafeeb2b.com/.

Price: 5 nodes – $60 per node, 10 nodes – $55.20 per node.

Test Environment: Three 750 MHz AMD Duron workstations with 128 MB RAM, 8 and 4 GB dual hard disks, CD-ROM, LS120 and 3.5-inch floppy, all running *Windows 98*. Pentium laptop with 48 MB RAM, 1.4 GG hard disk, CD-ROM and 3.5-inch floppy running *Windows 98*.

Virus Test-sets: Complete listings of the test-sets used are at http://www.virusbtn.com/Comparatives/NT/2001/02test_sets.html.

SUBSCRIPTION RATES

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

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

Editorial enquiries, subscription enquiries, orders and payments:

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

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

US subscriptions only:

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

Tel (781) 2139066, Fax (781) 2139067

END NOTES AND NEWS

Linux Expo 2001 Exhibition & Conference is to take place at Olympia, London in the UK from 4–7 July 2001. To find out about exhibition opportunities or to register for the show, email the organisers jonathan.neastie@itevents.co.uk or visit the conference Web site http://www.itevents.co.uk/.

***Sophos* is to host a two-day Anti-Virus Workshop on 24 and 25 July 2001** at its training suite in Abingdon, Oxfordshire, UK. For 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 *Internet World Event Network* has published autumn 2001 dates for exhibitions in Glasgow (Scotland), Dublin (Ireland) and Manchester (UK). For more information about exhibition opportunities and the full event line-up, see http://www.internetworld.co.uk/.

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

***F-Secure* has joined *Symbian's* Embedded Technology Partner program** to develop security technologies for mobile phones and wireless devices based on the *Symbian* platform. For more information, see http://www.F-Secure.com/.

Computer Weekly published the first issue in its Essential Series at the end of April. The Security issue contains comprehensive information about virus statistics, descriptions and advice concerning all aspect of content threats. For more details of upcoming issues on e-business and encryption, email jill.harrington@rbi.co.uk or contact her; Tel +44 208 652 9571.

***McAfee's ASaP* is an Internet-based service which is hosted, monitored and centrally managed by *McAfee* on behalf of service providers, and is transparent to the user.** It delivers automatic anti-virus and security vulnerability updates without requiring resources from the service provider or customer. *McAfee's WebShield e500 ASaP* anti-virus appliance simplifies service provider (xSP) deployment of AV solutions. For more details, see http://www.nai.com/.

The *AntiVirus Information Exchange Network* (*AVIEN*) has announced plans to create a new mailing list for smaller organizations. Currently, *AVIEN* membership is limited to those who work with more than 1,500 PCs and discussions revolve around enterprise-level issues with malware, and anti-malware solutions. For more information, see http://www.avien.org/.

***Symantec Corporation* are offering a free check-up service – Symantec Security Check – to identify and address vulnerbailities in PCs and Macs.** After running a scan, results are made available immediately. Users can then print out a detailed report of the results. The check runs a variety of tests, including a network vulnerability scan and a NetBIOS availability scan to assess hacker availability; an anti-virus software check, anti-virus definition check and active Trojan application to assess virus susceptibility; and a browser information check. See http://www.symantec.com/securitycheck for more details.

Swedish organization *Telia* has introduced an anti-virus service to its 700,000 Internet and broadband customers. A dedicated customer support unit will be established to complement the new venture. The service is added to the subscriber's regular Internet invoice or telephone bill. New virus definitions are updated when the computer is connected to the Internet. For more information, see the Web site http://www.telia.se/antivirus/.