ISSN 0956-9979 13666614 JANUARY 2003

VIRUS BULLETIN

THE INTERNATIONAL PUBLICATION ON COMPUTER VIRUS PREVENTION, RECOGNITION AND REMOVAL

IN THIS ISSUE:

• **Conventional wisdom:** Nick FitzGerald reports on recent efforts to extend and formalise the CARO Virus Naming Convention. See p.7.

• **It's the quiet ones you have to watch.** Although considered a 'minor' curiosity when it made its initial appearance, W32/Opaserv is fast becoming a major headache. Martin Overton looks at the spread of what some consider to be the 'quiet twin' of Klez. See p.10.

• **Upwardly mobile?** Analyst *IDC* forecasts the number of wireless SMS messages soaring from 1.4 billion messages in 2002 to a whopping 42 billion in 2006. However, users of SMS devices are increasingly being victimized by both spam and email worms. Mary Landesman investigates the dark side of SMS text messaging that could derail the gravy train. See p.14.

CONTENTS

COMMENT

“ Let’s face the truth: today’s software technologies do not stand a chance against the new and emerging metamorphic viruses. ”

If Not Now, Then When?

I believe that overcoming complacency should be one of our top priorities this year. As long as our methods and our thinking are reactive, so will be our solutions.

Will we continue to allow our adversaries to dictate how we use our applications, how our IT environments are run and structured, and the quality of our lives and economies? We fool ourselves in believing that we are doing ourselves a service when we disable a feature in a program so that a malware coder cannot use it as a security exploit against us. This is not freedom. It’s time for us to stand up and say ‘no more!’ to these pariahs. We are not strangers in this fight, so we should pool our resources, and up the ante for these perpetrators. It is time for the full legal resources of our corporations and organizations to unite and turn the tables.

It is no longer acceptable to play defence. We must go on the offensive. Let’s face the truth: today’s software technologies do not stand a chance against the new and emerging metamorphic viruses. We need to look outside the current toolsets to find innovations that will reduce our risk and mitigate the effects of a malware attack.

It is time for us to break with our current conditioning, to visualize the emerging horizons. It is time for upper management to buy into the seriousness of the situation at hand, and fully comprehend the risks and financial ramifications if we fail to act. It is time for our corporate officials to take a visible role in the battle against malware terrorists.

Until then, there are a few things each of us can start doing to improve the situation:

1. Demand extremely tough sentencing for convicted hackers and malware terrorists. Cooperate with and aid law enforcement in the capture of these criminals.

2. Look for new technological breakthroughs, new concepts in network and computer workstation security. For example, several new workstation hardware solutions have emerged.

3. Think like a hacker or a virus writer. What weakness(es) in your IT enterprise would you attack, given the opportunity? Start looking for ‘proactive’ solutions to fix those problems.

4. Join credible anti-hacker and anti-virus organizations, such as AVIEN (Anti-Virus Information Exchange Network). Get involved as a member and participate in the anti-virus forums. Power comes through joint cooperation, learning, and dissemination of information!

5. Walk the talk don’t just talk the walk! The best leadership is by example.

6. Educate senior management and give them honest appraisals and risk assessments. Encourage them to buy in by securing their commitment to your success. If you fail they fail!

7. Present senior management with an in-place plan for disaster management and recovery. The plan should include funded training for a disaster recovery team.

8. In an emergency have an offsite location to relocate computer operations.

9. Back up critical data daily and store the backup at a separate location.

10. Never, never give up the fight!

We have been conditioned to avoid confrontation and seek the path of least resistance, but confrontation is what is needed. If we do not unite and take the battle to the malware terrorist, we will continue to lose on all fronts. The control of the future can be ours, but we must be willing to earn it.

If not now, then when?

Joseph A. Broyles, P.C. Safe Devices, Inc., USA

NEWS

Lessons to be Learned

It seems that W32/Winevar.A was not the only virus 'story' to have arisen from the AVAR 2002 conference in Korea. While it seems likely that the release of the W32/Funlove.4099-dropping Winevar virus was timed to coincide with the event (infected messages may contain the subject line: 'Re: AVAR(Association of Anti-Virus Asia Researchers)'), another 200-odd copies of Funlove were merrily stowed away at the anti-virus conference itself.

Proving that mistakes can happen to us all, a red-faced and highly apologetic AVAR (Association of Anti-Virus Asia Researchers) Administrative Office reported in December that the AVAR 2002 conference CD-ROM contained an inactive version of W32/Funlove (virus code embedded).

The organising committee's investigations into the matter revealed that the CD-ROM's autorunner.exe file had become infected at the CD printing facility, where anti-virus shareware *Turbo-vaccine* was used to disinfect the file. Once back in the hands of the AVAR organising committee, anti-virus products from a number of vendors – *Network Associates*, *Symantec*, *Trend Micro*, and *Ahnlab* – were used to check the final product. On attaining no detections or alerts, the safety of the CD was verified and it was distributed to the conference delegates. However, following the discovery of viral code on the CD-ROM, a number of different AV packages – including those produced by *Kaspersky Labs*, *Computer Associates*, *DialogueScience* and *H+BEDV* – were found to pick up on the file, generating an infection alert, while *Sophos*'s product alerted on a viral fragment.

The embarrassed committee have offered to replace delegates' CDs with copies that do not contain virus code and have requested that the original versions be returned or disposed of ∎

A Happy New Year

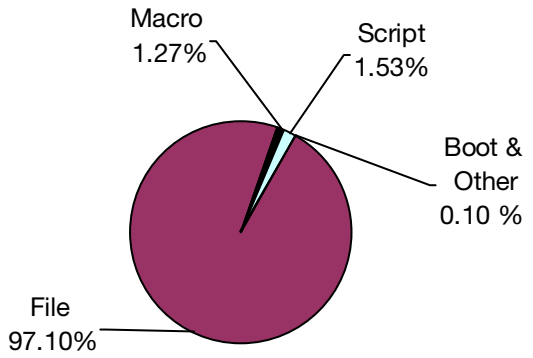
In a cheery end-of-year message, *mi2g* has made ten security predictions for 2003. Amongst predictions that global political tensions and conflict will be mirrored in the digital world are forecasts that the malware threat will continue to escalate in 2003 and that, while the number of new viruses and worms released may fall, the damage caused by a small number of '*killer*' viruses or worms will run to billions of dollars. The company anticipates that the proliferation of broadband will lead to more frequent attacks on small-to-medium sized entities and home users, and that Eastern Europe will continue to be 'the centre for virus and malicious code development'. But will 2003 be a year free from the over-used publicity technique of scaremongering? It appears not ∎

Prevalence Table – November 2002

Virus	Type	Incidents	Reports
Win32/Klez	File	3804	56.62%
Win32/Bugbear	File	1226	18.25%
Win32/Yaha	File	368	5.48%
Win32/Opaserv	File	325	4.84%
Win32/Magistr	File	198	2.95%
Win32/Braid	File	114	1.70%
Win32/SirCam	File	86	1.28%
Redlof	Script	74	1.10%
Win32/Nimda	File	64	0.95%
Win32/Funlove	File	53	0.79%
Win32/Hybris	File	51	0.76%
Win32/BadTrans	File	48	0.71%
Win95/CIH	File	45	0.67%
Win95/Spaces	File	41	0.61%
Win32/Elkern	File	36	0.54%
Laroux	Macro	33	0.49%
VCX	Macro	19	0.28%
Win32/MTX	File	13	0.19%
Divi	Macro	11	0.16%
Fortnight	Script	9	0.13%
Haptime	Script	9	0.13%
Others [1]		91	1.19%
Total		6718	100%

[1] The Prevalence Table includes a total of 91 reports across 50 further viruses. Readers are reminded that a complete listing is posted at http://www.virusbtn.com/Prevalence/.

Distribution of virus types in reports



LETTERS

Geographical Challenges

In the December 2002 issue of *Virus Bulletin*, Juha Saarinen shows some confusion over the sub-divisions of the English countryside (see *VB*, December 2002, p.8). Scunthorpe is in the county of Lincolnshire; the unitary authority (UA) of 'North Lincolnshire' is purely an administrative area. As a one-time resident of 'Bomber County' the notion of 'Humberberside' [*sic*] or the replacement UAs is abhorrent. The shire of Lindum Colonia suffers enough from tidal erosion without the further losses to the boundary commission.

Name and address withheld
UK

VB Responds

To set the record straight, VB *takes full responsibility for the insertion of the words 'North Lincolnshire town of' into Juha Saarinen's article. While the information presented was accurate,* VB *does apologise for any offence caused and extends its sympathies to residents and ex-residents of the shrinking 'Bomber County'.*

Definitions

Peter Sergeant's very useful review of Schweitzer's *Securing the Network from Malicious Code* (see *VB*, December 2002, p.17) is a little harsh in one respect. The term 'social engineering' does have the same ambivalence he observes in Schweitzer's book. In the social sciences, 'social engineering' tends to have the ethically neutral meaning of changing the opinions and behaviour of a population by legislative and other means. Use of the term may well predate its adoption by the black hat community with reference to gaining access to systems by duping legitimate users – which is why, in my formal writings on the subject in and out of the virus arena, I prefer to use the broader definition 'psychological manipulation of an individual or set of individuals to produce a desired effect on their behaviour'.

David Harley
Independent Researcher & Author, UK

Missing the Bigger Picture?

I disagree with some of the statements made by Péter Ször in his Comment 'The Bigger Picture' (see *VB*, December 2002, p.2).

It's a fact that integrated AV and security products with a one-point management console are a must-have feature for all larger companies. However, the top four products on the market offer almost identical features and purchasing decisions are made according to a few minor differences (including the prices). The products are so similar that you could easily substitute one of the products for another!

From a vendor's point of view, the best strategy to avoid this situation is to make your product unique in some way, so that no other company offers the same, or nearly the same, features.

Take speed, for example. An assembly-written program is able to run only on specific platforms without a complete rewrite. That's correct.

No program is as lightning fast as *Eset*'s *Nod32*. Even when run on very old PCs there is no visible decrease in performance. Of course, the program has some limitations – limited support of archive formats, for example.

Then there are programs whose features are the complete opposite: *Kaspersky AntiVirus* is able to scan inside nearly every compression, email and every file format I know of. But this unique feature comes with its own limitation – it slows down the PC a lot, so it needs to be run on a fast computer.

Between these two extremes you will find plenty of other programs that do not include such unique features – easily 'replaceable' programs.

From the 'integration' point of view, it is helpful to be able to manage not only your own product, but the most important functions of competitors' products, too. In many cases, big companies use more than one AV product in their environment – with different products on the servers, clients, mail servers, gateways and so on.

We should also keep in mind that there have been a lot of company mergers recently. In such cases, there may be different AV solutions in place within the newly formed company at the same time – which then have to be managed by the system administrators. They are likely to be unable to switch products due to pre-existing long-term contracts with the providers of the products and the high costs involved in switching.

And we should not forget that it is not only the large companies who need protection. Of course, these customers are very important for a regular income and the costs of serving them are relatively low compared to the same number of clients in a lot of small companies. But small companies have other interests, and their requirements are very different. They do not need a huge management console for the protection of maybe five or ten PCs. They need other, smaller and easier tools that everyone is able to use – to update all products at the same time, for example. A lot of AV companies do not offer such a 'small' solution!

Currently, the most interesting growing platform is *Linux*. I know of two AV companies already that make more money with products for this platform (mainly for file servers and mail gateways) than for *Windows*. Why? Because no other AV company offers products like theirs! Most companies already have a command-line scanner for Linux – but who cares about that? The demand for an integrated solution not only for *Linux*, but for all Unix platforms is high …with unique features, of course.

Andreas Marx
AV-Test.org, Germany

AV Tops the Spam Charts

According to *Brightmail*, anti-virus software offers were the most frequent type of spam messages in 2002, which is interesting. Could it be that the excessive hyping by certain rapscallions in the AV industry is the cause of my unsolicited commercial email katzenjammer?

The deluge of anti-virus offers could be seen in either a positive or a negative light – on the one hand, they may serve as reminders to home users that they should be using anti-virus software. On the other hand, however, those users who do 'Protect [their] Computer Against Viruses for $9.95' may be lulled into a false sense of security, installing the software and assuming blindly that they'll never get bitten. Personally, I am inclined to state that any spam I receive is a barrier in front of my achieving a eudaemonic lifestyle.

So, who's to blame? Should we be looking to anti-virus vendors to go after spamming resellers more vigorously? Is it *Symantec*'s job to track down those pimping cheap copies of *Norton Anti-Virus*? Is there an address to which I can forward emails trying to flog me *McAfee*'s software at rock-bottom prices?

I find it beautifully ironic that vendors themselves receive this type of spam – but it turns out that the AV industry is not the only one to experience recursive marketing. From the *Google* website: 'Dear google.com, I visited your website and noticed that you are not listed in most of the major search engines and directories …'

Lavash en-Rangé
UK

VIRUS ANALYSIS

IM a Hot Rod(ok)

Robert Pareja
TrendLabs, Trend Micro Inc., Philippines

Email has revolutionized the way people communicate with each other. Since the advent of email it has become possible to conduct business correspondence with less expense and a faster delivery rate. On a personal level, correspondence no longer needs to involve a visit to the local post office.

On a less positive note, however, people who have little to occupy their spare time have also taken advantage of this revolution in communication, and the result is the existence of email worms. Employing social engineering techniques and taking advantage of vulnerabilities in email client programs and users' ignorance of safe-computing guidelines, gave rise to some of the computing industry's worst nightmares: Nimda, Klez, Frethem and BugBear to name just a few.

Whatever is Popular Deserves Attention

The increasing popularity of instant messengers (IMs) has introduced a new way in which worms can spread. Although messenger services have been around for a very long time, it is only now that they are becoming more widely used. Their rising popularity may be attributed to the more personal touch they bear – buddy icons, emoticons and graphical smileys – as well as the new features that most messenger clients offer, such as file exchange, voice communication over IP, group chat and integration with email services.

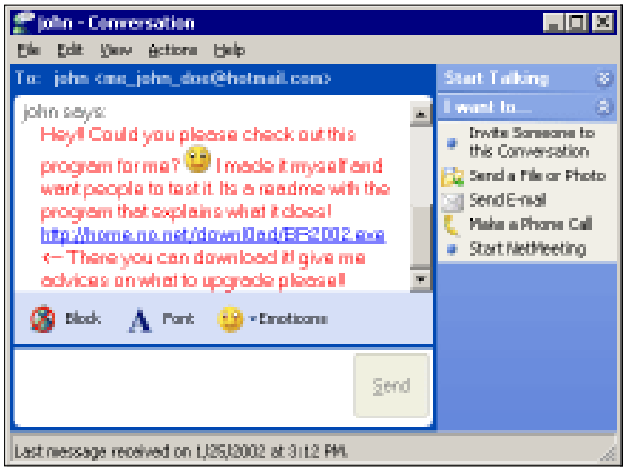
This increase in the use of instant messengers gives them the same attractiveness worm writers saw originally in email. Using old methods on this system, worm writers now have a new instrument by which they can steal vital information, cause nuisance and create havoc for the unsuspecting user.

Key to Success

On 2 October 2002, one such worm caught the attention of the computing industry. W32/Rodok.A seems to have originated from Norway, but its infection was most noticeable in Asia.

W32/Rodok was created in Visual Basic, and it comes disguised as a CD key generator downloaded from a website.

The worm's life cycle begins with the delivery of a message (shown below) to the unsuspecting IM user. The message contains a link to a website and asks the recipient to 'check out' a program and 'give advices [*sic*] on what to upgrade'.



The website hosts the actual copy of the worm, and at this point the user is not yet infected, since the worm is not physically present in the recipient's system.

Since the message appears to have come from a valid contact, many users will not be suspicious of the request. Enticed, the recipient clicks the URL and the default web browser downloads a copy of the worm onto the user's machine.

Depending on the security settings of the web browser, the user may be prompted to download a copy of the file into the local file system and another prompt may ask the user whether they wish to execute the downloaded file.

Once the downloaded file is run, the worm manifests itself to the user as a CD key generator. The supposed generated key is actually just a random number.

The worm then sends out a copy of the same IM message received by the user to all the active or online contacts in the infected user's MSN contact list. (Luckily, MSN doesn't yet have a feature that enables the sending of messages to offline users; otherwise the worm's distribution would have been greater.)

At this point, the worm has succeeded in the first part of its infection.

The Evil Twin

The worm then connects back to the website to download an updated copy of itself (http://home.no.net/downl0ad/Update.exe) to C:\update35784.exe. At the time of infection, this downloaded file and the initial copy of the downloaded worm were one and the same.

Many Internet worms register themselves into the host's system using the Registry, initialization scripts and special dropped files. Worms do this to increase the chances of

infecting more users by making sure the worm runs every time the system is restarted. However, W32/Rodok.A does not do this.

After sending its messages to the MSN users, it downloads C:\update35784.exe from http://home.no.net/downl0ad/Update.exe. It is believed that this feature was created to enable the worm to download updates of itself in the future. However, at the time of the infection, the downloaded file is actually just an exact copy of the backdoor Trojan discussed in the next paragraph.

Next, the worm downloads the file http://home.no.net/downl0ad/CS-Keygen.exe, saves it to C:\hehe2397824.exe and executes it as another process. A modified version of backdoor Trojan Troj/Evilbot.A (alias Win32/Brat.02.A), this UPX compressed program drops a copy of itself in the Windows folder as WinUpdat.exeupdate.ur.address (6,688 bytes) and installs itself in the Windows Registry in order to run at the next re-boot. As with its other variants, the primary function of this backdoor Trojan is to receive instructions via IRC and launch denial of service attacks from the compromised machine.

The Key-napper

The worm sends out another message – still in the form of an instant message – which goes to an MSN user, styggefolk@hotmail.com, who is believed to be the author of the worm. The message appears as follows:

```
<infected person> says:
I have loaded the ur CDKEY Generator 1.3! CS:
<CS Key> HL: <HL Key>
```

where <infected person> refers to the email address of the person whose machine is currently hosting the worm.

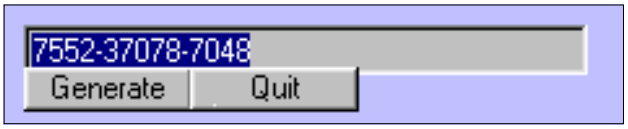
```
<CS Key> is the data of the following registry
value:
HKEY_CURRENT_USER\Software\Valve\CounterStrike\Settings,
Key
<HL Key> is the data of the following registry
value:
HKEY_CURRENT_USER\Software\Valve\Half-
Life\Settings, Key
```

Old Lessons Relearned

The demise of this worm's spread was seen just a few hours after its initial infection. The posting of pattern updates and, most importantly, the removal of the worm from the website were the major contributing factors to its demise.

Despite the low threat of W32/Rodok, the short amount of time this worm spent in the limelight was enough to encourage system administrators and IT personnel to review their security policies. While some blocked *MSN* IM ports, those who took more drastic action disabled all outgoing instant messaging ports.

This worm has demonstrated that the resolve of worm writers remains firm. As anti-virus companies, IT security



firms, and software vendors have increased the reliability of email scanning software, worm writers have continued in their search to find new avenues by which to infiltrate the privacy of individual users and organizations alike.

Detecting a worm like this is not always an easy task. In fact, we should consider ourselves lucky that W32/Rodok.A was not as harmful as some of the other worms that preceded it. Some companies may turn to packet filtering software to detect this type of malware. This may work in some, but not all cases – messaging protocols are becoming more flexible, to such an extent that they can override other protocols, such as security and data compression, and thus pass through the network in an encrypted form.

The users themselves also have a role to play in computer security if worms like this are to be thwarted. Knowledge and awareness of safe computing guidelines (and putting them into practice) are essential. You can never be sure who your friends are, and it is better to be wary of everyone, than to be sorry in the end.

Anti-virus companies still bear a special task in the defence against such threats as consumers continue to enjoy the benefits of instant messenger services. As messaging crosses the boundaries of personal computers to PDAs and other wireless devices, the value of such software will establish itself.

Conclusion

W32/Rodok.A is a worm that crawls via the *MSN Instant Messaging* protocol, stealing registration keys as it moves along. It uses social engineering techniques to encourage unwary users to download and execute the worm which itself downloads a backdoor Trojan application. Despite the simplicity of this worm, W32/Rodok.A has shown us one thing: the next time we receive a message, whether via email, IRC or IM, we should look again. It might be another ROD on the loose.

W32/Rodok.A	
Aliases:	WORM_RODOK.A, W32/Fleming.worm, W32.HLLW.Henpeck.
Type:	Worm, downloader program.
Size:	53,248 bytes.
Removal:	Delete the files detected. Registry entries created by Troj/Evilbot.A may also be present and should be removed.

FEATURE 1

A Virus by Any Other Name – Virus Naming Updated

Nick FitzGerald
Computer Virus Consulting Ltd, New Zealand

[*Recent efforts have been made to extend and formalise the CARO Virus Naming Convention in order to accommodate the new forms of malware that have appeared since the Convention's initial release in 1991, and in order to address the broad range of naming concerns and confusions facing today's malware researchers. Here, CARO member Nick FitzGerald summarises some of the points that have been agreed upon in advance of the production of a formal description of the specification.*]

Early CARO members Vesselin Bontchev, Fridrik Skulason (of *FSI*, manufacturer of *F-PROT*) and Alan Solomon (then of *S&S International*, manufacturer of *Dr Solomon's AntiVirus ToolKit*) met at the 1991 *Virus Bulletin* conference in Brighton. Together they compiled a list of dos and don'ts and formalized the structure of virus names. Some time later their notes were 'tidied up' and distributed in a file known as 'naming.txt'. Their intention was to use this proposed naming scheme in their own products; their hope was that other anti-virus developers would follow their lead and use the suggested standard in their products too.

But, as the old saying goes, 'you can lead a horse to water, but you cannot make it drink' – the scheme suggested in naming.txt was largely ignored. Apparently, in fact, some vendors even went so far as to adopt the stance that the scheme should not only be ignored, but actively resisted because of one of the proponents. Thus, the first and only attempt to standardize computer virus names largely floundered.

Undeterred by this reaction, these and some other CARO members adopted the naming convention for their anti-virus products; but some did not. Some developers who were not in CARO also committed to using the scheme in their products and copied the names used in CARO members' products. Unfortunately, the rate of growth in new viruses started to take off. The rate at which they spread around the world increased as PCs became a more common 'business tool'; and data (and program) sharing between sites increased.

CARO members tried to handle the situation by holding naming meetings to agree the names for each new virus. Such meetings were usually held at conferences, where it was convenient for all the members to come together. Some AV developers without CARO members on their staff copied the new names, but many did not – and even those who did copy the virus names were seldom aware of the renamings that were agreed at these meetings. Thus the naming mess deepened.

Enter VGrep

As, hopefully, all regular *VB* readers know, VGrep (which is available for online searching or download as a standalone *Windows* executable version from http://www.virusbtn.com/resources/vgrep/) provides a comprehensive cross-referencing of virus names reported by about 20 virus scanners run against a sample set of approximately 250,000 malware samples. Please don't get me wrong on this, given the situation we are in, VGrep is a very useful thing and I am very happy that former *VB* editor Ian Whalley developed it, and that Dmitry Gryaznov of *Network Associates* has taken over the task of updating the cross-reference database.

VGrep's very existence is evidence of an odd contradiction in this industry. The fact that it is needed is proof of how little the industry as a whole cares about naming consistency – if the industry really cared about naming, VGrep would not be needed (neither would this article).

Yet, VGrep is very useful because, for many, many people, it answers naming questions that really *do* matter. Whether you are an anti-virus developer, a system or network administrator, or technical support staff, if you must deal with malware names from more than one vendor, VGrep is very useful.

So useful, in fact, that many AV developers felt that VGrep would satisfy the needs of users who complained about their virus naming inconsistencies. VGrep may have resulted in these developers paying even less attention to naming issues than before – or may at least have left them feeling justified in failing to address their own contribution to the naming mess.

Unfortunately, the release of VGrep coincided roughly with the greatest increase in malware numbers we have seen, and the consequent growth in the staffing of AV research labs. Therefore, many of those new staff learned that 'naming does not matter as VGrep will take care of the mess'.

The upshot? We now live in a world where your virus scanner can be updated within minutes of the initial detection of a new, fast-spreading virus. That is a significant achievement. However, in many, many cases, your scanner will not be updated in hours, days, weeks or even months to reflect that the broader AV research community has agreed on a 'better' name for that virus.

What's in a Name?

In a word, identifiers. An identifier is simply a string of characters chosen from a specified character set, possibly

limited further by convention and dependent on the type of malware.

The revised CARO naming scheme describes what identifiers are used where and how, and is based on the original scheme described in naming.txt. It incorporates extensions necessitated by the increasingly complex array of malware developed since 1991, most of which have been agreed by various different groups of malware specialists since then.

Names are comprised of sets of identifiers. Each identifier has its own set of rules and some less specific guidelines for choosing 'good' names are also included. In addition, these rules specify which identifiers are required for what types of malware.

An important issue to bear in mind when discussing malware names is the distinction between the formal, technically correct names used when anti-virus researchers are talking among themselves, and the simpler naming forms that are required when reporting a malware detection to an end user.

For example, while technically correct the name 'virus://{W97M,X97M,PP97M}/Tristate.A' should (probably) never be presented to an end user. The scheme includes some guidelines for simplifying complex technical names in order to produce names that are suitable for reporting to users. The recommended reporting form in this case is O97M/Tristate.A.

Anyway, we're getting ahead of ourselves here – what does an identifier consist of?

Naming.txt allowed a fairly complete set of English alphanumeric and punctuation characters. The new standard limits this set, removing all but two non-alphanumerics: [-_A-Za-z0-9].

Making existing names comply with this set will require quite a lot of renaming in many products, but this will be a one-time change (we will not – in fact, just about *cannot* – reduce the character set any further) and it provides some desirable simplification. What's more, many developers indicated that, for new family names, they already try to limit themselves pretty much to this set anyway.

Naming.txt also allowed [$%&!'‘#] plus the space character. If existing viruses are to be renamed to comply, the following guides should be followed:

- Spaces should be replaced with '_' or removed.
- '&' should be replaced with '_And_'.
- Replace '%' with '_Pct_' (or '_Pct' if it was the last character in an identifier).

All the rest of these characters should just be removed from names that include them currently. Note that, for the purposes of comparing names, '_' is treated as a null character so, for example, 'Foo_Bar' and 'FooBar' are equivalent names.

Identifiers are limited to 20 characters, but some must be shorter. For example, normal locale specifiers must be two characters. However, even for those that are not constrained by size, shorter is preferred in general (although shortening identifiers arbitrarily is not an end unto itself).

Alphabetic characters are case-insensitive, but mixed-case use is encouraged. Use of numeric characters and 'number names' is very strongly discouraged apart from specific identifiers that are required to be numeric. This is described in detail in 'A Virus by Any Other Name', *AVAR 2002 Conference Proceedings*, p.141.

Aside from characters in identifiers, naming.txt allowed two separators to delimit identifiers – the dot character ('.') and colon (':'). The new delimiter character set is extended to [!#./:@].

The use of delimiters is very specific and spelled out in the new standard. Do not use arbitrary combinations of your own – use only specified delimiters in specified places.

Full Names

So, how do we combine identifiers to make full names?

Identifiers are combined to form full names according to several rules. These affect the order of combination, the types of identifier required in the names of various types of malware, and which optional modifiers can be used where. The general form of malware names under the new guidelines is:

```
<malware_type>://<platform>/
<family_name>.<group_name>.<infective_length>.<sub-
variant><devolution><modifiers>
```

For a detailed explanation of each name component, the AVAR paper should be consulted. The most important aspects of each component will be described briefly here.

The *malware type* identifier specifies whether the malware in question is a virus, a Trojan, a dropper, an intended virus, a malware generator kit, or 'garbage'. The correct use of these is explained in detail in the AVAR paper, but note that this list is very constrained and does not contain the type 'worm'.

The *platform specifier* is an area of some confusion. It denotes the operating system, interpreter or other kind of runtime environment necessary for the malware to function.

Several AV developers make up platforms for their own diverse reasons, often associated with gaining publicity. This is simply wrong – if we are to maintain naming consistency, acceptable platform names have to be agreed across the industry and several recent 'new' platform names that vendors have used are clear examples of ignorance or laziness in considering the issues involved. Appendix A of

the AVAR paper lists the currently accepted platform names and Appendix B lists some commonly, but wrongly, used platform names. A few previously popular platform names have been put in Appendix B – for example, the platform names of the various *Windows OS*-specific viruses have been rationalized.

The *family name* is the most important component, given that it is the only one that must be reported to users. Thus, the most confusion is caused if there is inter-vendor disagreement over this component of a malware's name. Malware is grouped into families based on code similarity. There are some broad guidelines as to how to choose good names and avoid bad ones in the AVAR paper, but in general this is the hardest part of naming new malware. The common suggestion of the scheme allowing the use of popular names (or 'aliases') was rejected. First, this naming scheme provides a formal taxonomic structure, so such names are irrelevant to the scheme itself. Second, AV developers make such a mess of things as it is, that 'encouraging' non-consistency will surely not help make things better!

Group names have been retained mainly for backwards compatibility – many old DOS virus families are further divided into groups. The use of group names is strongly discouraged when naming new malware families.

The *infective length* identifier has been renamed and its use is more tightly specified in the revised naming scheme. It must only be used (and *must* be used) when fully specifying a parasitic executable infector. Non-viral malware and non-parasitic viruses no longer have infective lengths in their names. Obviously, this identifier is one of the exceptions to the 'avoid numbers' rule. For malware that requires an infective length to be specified it will usually be the main variant identifier.

Every malware variant is given a *sub-variant* identifier in its full, formal name. These start with '.A', '.B' and so on to '.Z' then '.AA', '.AB' and so on. For malware that does not require an infective length identifier, this is the main variant identifier.

The devolution identifier is the other purely numeric one, and is used to denote very closely related variants that are often created through 'imperfect' replication. Devolved variants are rare and tend only to feature in the naming of a few macro viruses. More details about the correct use of this identifier can be found in the AVAR paper. Note that it is incorrect to use this identifier to indicate a minor variant caused by imperfect disinfection and the like – such variants should be assigned a new sub-variant identifier rather than a devolution ascription as is becoming increasingly common, but wrong, in some products.

Modifiers

That is the end of the identifiers that are, or may be, required to name a malware properly. Several *modifiers* which add further, possibly useful, information are also allowed. These are all optional.

```
[[:<locale_specifier>][#<packer>][@'m'|'mm'][!<vendor-
specific_comment>]]
```

A *locale specifier* is used to indicate that the object is only malware on a specific localized version of its platform. The identifier is formed with a colon followed by the standard two-letter abbreviation of the locale or language localization. The only exception is the newly introduced special locale specifier ':Uni', which is used to indicate objects that are malware only on Unicode versions of the platform and/or on unknown/unspecified DBCS language localizations.

The *packer* modifier is used to indicate that something is a known malware that has been compressed or encrypted with a runtime decompressor or decrypter. In practice is seems to be seldom used.

The *@m* and *@mm* modifiers have become well-established and well-understood by corporate system administrators where their use effectively indicates 'make sure we get the emergency updates as soon as possible'. As many people set their pagers to go off on receipt of vendor alert emails containing a name with these modifiers, they should be used cautiously. Use them only for genuinely functional self-mailers – if you are not sure, do not use them.

The *vendor-specific* comment is a new addition to the standard. It allows vendors who feel they really must tell users something more about a piece of malware to do so in a standard way, thus removing excuses for not using the rest of the standard naming scheme. If used, this comment absolutely *must* be the last (right-most) modifier. What this modifier really means is 'everything from the first exclamation mark to the end of the virus "name" is not actually part of the name'. It can contain pretty much anything – although it is limited to the characters valid in modifiers, delimiters and the simple set notation used to specify multiple values in one identifier's place (this last rather technical feature is not described in this article due to space limitations).

Another new feature is the introduction of a pseudo-platform name. This is purely for reporting purposes – that is, it is never part of a formal malware name – and is intended for easing reporting of complex multi-component and/or multi-platform malware. This is described in more detail in the full paper, but its use is still under discussion and not fully decided. The name is 'Multi' or 'Mul' as a short-form.

What's Next?

There are still a couple of issues to resolve. These discussions will be finalized soon, followed by the production of a formal description of the specification. We would be interested in hearing readers' comments and any further suggestions, which may be sent to comments@virusbtn.com.

FEATURE 2

Are You Being [Opa]Serve[d]?

Martin Overton
Independent Researcher, UK

W32/Opaserv, which first appeared as a 'minor' curiosity at the end of September 2002 (see *VB*, December 2002, p.6), is fast becoming a major headache not just for the 'great unwashed' (the public) with their xDSL/Cable/Modem Internet-connected systems, but for a large number of organisations too.

Underestimating

It appears that many people did not consider Opaserv to be a threat because they felt that it couldn't become widespread using Windows Shares as its only infection vector.

This 'incorrect' thinking was partly due to the fact that many companies (quite rightly) do not allow Windows Share (SMB) traffic to traverse their corporate firewalls (Port 137 UDP 139 TCP).

Furthermore, a subset of these companies prohibits the use of 'open' shares on their internal network(s). An even smaller subset prohibits P2P use, for the same reasons – fear of confidential data leakage, copyright infringement, malware, and other security risks.

However, the 'great unwashed' have shown that they have little understanding of security. They do not seem to understand the need to run anti-virus products or personal firewalls, and do not comprehend how 'open' their systems have been left as a result of not following basic safe computing guidelines.

Not only have they left their systems exposed to the likes of Opaserv, but also to the mounds of other malware, including RATs and DDoS bots/zombies and to the ubiquitous 'hackers', 'script kiddies' and other mischief makers. Furthermore, they may unwittingly be exposing their personal data, including personal documents and correspondence, credit card details, passwords, ISP details and who knows what else.

The Quiet One

In many ways, Opaserv is the quiet twin of Klez; like Klez it is very widespread but it uses a different infection vector (Windows Shares [SMB] rather than email [SMTP]) to achieve its ends. (Allegedly Klez can spread via Windows Shares, however this is a secondary 'vector' for it, and I have yet to see a sample that has been dropped to my worm lure via this route.)

Although Opaserv is relatively harmless, it has caused similar traffic patterns to those caused by Nimda and CodeRed. Thankfully, Opaserv is not as aggressive in its scanning of a network for new victims.

Let's have a look at some of the reasons why Opaserv and its increasing number of variants are spreading so far and so quickly.

Evolution

Over the last few months a number of new variants of Opaserv have been created. In the majority of cases the changes have been subtle – for example, new website addresses from which the worm grabs updated versions of itself, and the use of varying file compression and/or encryption tools in an attempt to conceal the modified malware.

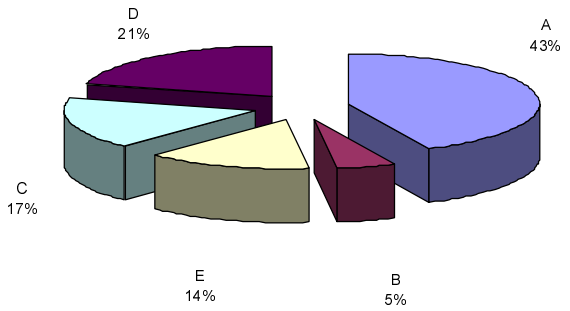
The table below shows the dates when these new (i.e. unknown) variants were first detected by my Internet-facing worm lure. The lure is used to catch new malware In the Wild and to monitor how widespread a particular piece of malware using this infection vector becomes (along with other tools for malware which use other infection vectors). In most cases the 'new variants' of Opaserv were caught by my worm lure within hours of their initial deployment by the author(s).

The samples were catalogued, and the information that was gleaned from the initial (brief) investigation and testing was sent to a number of anti-virus vendors, along with the trapped samples. In addition, the data was transmitted to the AVIEN mailing lists (minus the samples), to give members a 'heads-up' on the new threat.

Shown below is the timeline of new Opaserv variants trapped (the variant naming used for this timeline is from *F-Prot* [F] and *NAI*/*McAfee* [N]):

Filename	***Date Trapped***	***F/N***
Scrsvr.exe	N/A	A/A-D
Brasil.pif	19 October 2002	A/E
Brasil.exe	20 October 2002	A/F
Alevir.exe	22 October 2002	C/G
Marco!.scr	28 October 2002	D/I
Puta!!.scr	7 November 2002	?/?
Instit.bat	10 November 2002	E/K

(Brasil.exe was first spotted by AVIEN member Mark Ackermans.)



Change for Change's Sake

The pie-chart above shows the distribution spread of almost 10,000 trapped samples of the current (as of 26 November 2002) known variants of Opaserv. The variant naming used for this chart is that used by *F-Prot*.

As can be seen, some variants have spread more quickly and widely than others. In many cases it may be the changing filenames and the use of packing and encryption tools that have allowed new variants to gain a 'beachhead'.

It seems clear that many anti-virus companies need to improve their handling of packing and compression tools. It appears that plenty of malware authors are well aware that a number of anti-virus products are 'limited' in this area.

Risky Business

Known MS Security Holes

Opaserv takes advantage of the password exploit on *Windows 9x/Me* which is fixed by installing MS00-72. Luckily *Windows NT/2000/XP* are not vulnerable to this specific attack.

Holey Network Batman!

Opaserv takes advantage of Windows Shares, including password protected shares, as long as:

1. The whole of drive C: is shared as 'C'.
2. It is either open (with no password) or it is password protected but not patched with the MS00-072 (*Win 9x/Me* only).
3. It is writeable.
4. The share is visible to an infected system on the network (internal/home) that it lives in, or the system has Netbios over TCP/IP enabled (which makes Windows Shares available to other Internet systems and/or users).

What are the risks in a corporate environment, where the corporate firewall does not allow SMB traffic to/from the Internet?

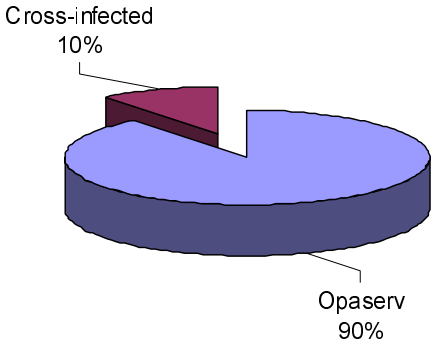
Opaserv can still get in. Let's investigate possible back- and side-doors that it could use to bypass the front-door bouncers (firewall and perimeter AV).

These include:

- Modems on desktops, laptops and servers.
- Laptops that are taken home or to customer sites.
- VPN installations that allow dual-access (Internet and corporate network at the same time) for home-based workers or those on customer networks.
- Other remote access software for home-based workers via their xDSL/Cable/Modem connection.
- Web-based email (yes, I have seen Opaserv sent as a file attachment via email).
- P2P, such as *KaZaA*, *WinMX*, *Gnutella* (maybe disguised as another piece of software).
- Hacked or back-doored systems.
- Disgruntled employees.
- Re-packaged with a compression/encryption tool that the perimeter AV can't handle or doesn't recognise, but doesn't quarantine.

Cross infection

The following chart clearly shows that Opaserv is acting as a 'carrier' and is transporting other malware along with it (unbeknown to Opaserv) – acting as a sort of binary 'Typhoid Mary'.



This could also explain the resurgence of W32/Funlove (especially with W32/Braid dropping W32/Funlove too), as this virus was the most common 'passenger' on files dropped by Opaserv.

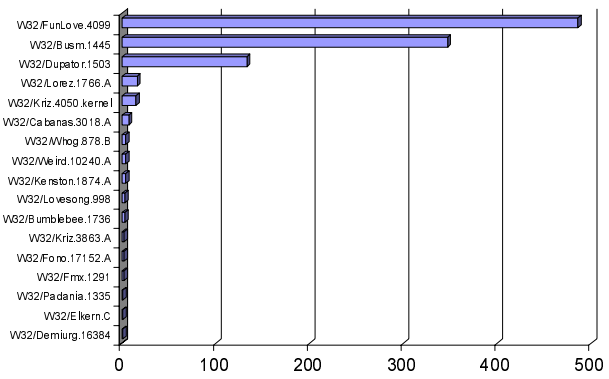
On the following page is a bar chart showing the full list of 'passenger' malware that has been found hitching a ride on the Opaserv malware transit system so far.

I Can See You

Not only does Opaserv drop files onto a new victim system, it also makes changes to ensure that it is executed the next time the system is restarted.

Dropped Files

An integrity system will alert on new files and system modifications which may indicate that Opaserv or another



SMB-aware (or other infection vector-based) piece of malware is active on a system.

To date, Opaserv has dropped files masquerading as .BAT, .EXE, .PIF and .SCR file extensions with fixed names (Brasil.exe, Brasil.pif, marco!.scr, scrsvr.exe, scrupd.exe, puta!!.exe, alevir.exe and instit.bat). Luckily, it hasn't (yet) used random or similar (list) filename generation.

System File Modifications

Opaserv modifies WIN.INI by changing the 'run=' line to read as follows:

```
run=c:\windows\<variant_name.variant_extension>
```

In some cases this line can contain multiple variants on the same line, for example:

```
run=c:\windows\scrsvr.exe,c:\windows\marco!.scr,
c:\windows\Brasil.exe,c:\windows\alevir.exe,
c:\windows\instit.bat
```

On systems that have been infected it is quite likely, even though the worm has been blocked or removed by an on-access or on-demand anti-virus solution, that the WIN.INI will still be modified and contain numerous references to the worm files, which will cause errors to be displayed when the system is next restarted. This seems quite ludicrous!

I have even seen several cases in which the WIN.INI file has become truncated or corrupted due to modification by Opaserv.

Once Opaserv has been run (usually via the modification it made to the WIN.INI 'run=' line) and the system has been restarted, it adds an entry and a call to itself in the following Registry key, which ensures that the worm is loaded when the system starts (even if the WIN.INI call has been removed):

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\
CurrentVersion\Run
```

There has been a lot of confusion as to whether Opaserv can (and does) drop copies of itself on *Windows NT/2K* or *XP*-based systems.

The worm lure I use is based on *Samba* running on *Linux*. However, to other *Windows* systems this appears to be a *Windows NT 4.0* server. Opaserv will happily drop copies of itself onto this system and modify the WIN.INI file as if it is a 'real' *Windows* system. However, most anti-virus companies' descriptions claim that Opaserv can work and drop copies of itself only on *Windows 9x/Me*-based systems. A little clarification is needed! While Opaserv will successfully copy itself to *Windows NT/2K/XP* machines as long as it finds an open (not password-protected), writable share 'C' with the directory 'Windows', the chances of meeting those requirements are much slimmer that finding the suitable conditions on *Windows 9x/ME*.

Personal Firewalls

Firewalls can be used as an effective way to slow or even block Opaserv, by enabling the 'Windows File and Print Sharing on the Internet protection' (the exact wording depends on your personal firewall product).

Most personal firewall products have this 'rule' or 'protection' switched on by default. However, this will not slow Opaserv down on the user's 'home' or 'corporate' network. To resolve this would require the addition of a 'custom' rule.

Desktop Anti-Virus

Yes, desktop (and perimeter) anti-virus packages have their part to play and, once they have been updated to detect the new variants of Opaserv, they can and do block the Opaserv 'dropped files'.

However, it seems that, although the anti-virus packages block the files, the end-user is left to clean out WIN.INI after each infection attempt – in most (if not all) cases, anti-virus programs do not stop Opaserv from modifying this file.

As I have been seeing up to 1000 such modifications each day, I imagine that this soon becomes a major chore for the Opaserv-afflicted, but AV-protected, user.

Caught Red-Handed

Intrusion Detection System

Yes, you can use an IDS as an anti-malware detection and/or blocking tool.

It is fairly straightforward to create custom malware signatures or rules (especially for SNORT). These can then be used either to log attempts (for later remediation) or to block/drop specific requests or connections.

This is very useful when dealing with new malware for which the anti-virus companies have not yet added detection, or for blocking 'self-updating' or 'phone-home' malware and related risks.

SMB Lure

Since Opaserv first appeared (approx. two months ago), my Internet-facing SMB Lure has trapped over 14,000 samples, yes samples, as I have made some major changes to the basic SMB Lure design that was created by John Morris of *Nortel Networks*. These changes include:

- Sample capture, once the dropped/modified file is completed.
- Intrusion Detection with custom malware signatures (logs IP addresses).
- Integrity checking, so that changed, added or removed files can be handled accordingly.
- MD5 hash table to compare trapped samples against known MD5 hashes of specific malware and variants.

And more improvements are planned.

Router/Firewall/Proxy Logs …

Regular reviews of log files looking for (unusual) quantities of outbound traffic on port 80 to the known 'Opaserv update' or 'non-mainstream' websites:

- http://www.opasoft.com/
- http://www.n3t.com.br/

How Opaserv can be Beaten or Held at Bay

Patching, Patching, and More Patching

Regular system maintenance is imperative. If you didn't learn this from Kak, CodeRed and Nimda, as well as the myriad other malware that have exploited security holes in either an operating system or an application, then you are forever cursed to suffer from such malware until you learn the lesson that malware history has desperately been trying to teach you.

Don't Share

Stop sharing, or change the way you use shares. Windows shares are useful, but should be used as a last resort, and never across the Internet! If you must use Windows shares, then:

1. Ensure that your system is fully patched.
2. Do not share the whole hard disk; share the directories that you need only.
3. Do not allow write access unless you really need to.
4. Use 'User-level access control' rather than 'Share-level access control' as this is slightly more secure.
5. Use a complex password of at least eight characters, and use non-alphanumeric characters too, not just alphanumeric.
6. Use a firewall and/or unbind Netbios over TCP.

Unbind Netbui/Netbios

Unbinding Netbui/Netbios over TCP/IP from Internet interfaces (i.e. Modem/xDSL/Cable) is strongly recommended. See http://www.mikeshardware.com/howtos/howto_disable_netbios.html for details of how to achieve this.

If you must use Netbios over TCP, then install a firewall. Configure the firewall correctly to block Netbios traffic to/from the 'Internet' by default, then set up specific rules to the IP addresses that you require Netbios over TCP for on your 'home' or 'corporate' network.

Conclusions

W32/Opaserv has surprised many people by becoming so widespread. It is highly likely that its success in terms of propagation has been noticed by other malware authors, and that they will add 'SMB' as an extra infection vector to their upcoming releases.

We (at least, the 'great unwashed') should be thankful that the current versions of Opaserv have not been destructive, nor had a nasty payload.

The fact that you are on a 'corporate' network does not mean that you can't or won't see Opaserv.

SMB Lure-based trap systems are useful additions to corporate networks, as they can log the IP addresses of 'internally' infected systems which are looking for new victims to infect. This will allow remedial action to be taken more quickly than would be possible when relying on the network team to notice and/or mention 'strange' or enlarged quantities of SMB traffic and large numbers of outbound requests to a specific site.

Like Roger Thompson's *WormCatcher* (see *VB*, December 2001, p.4), SMB Lure is a very useful early-warning system, and compliments *WormCatcher* well. It is well worth the minimal trouble in setting it up and maintaining it.

Both early warning systems should be considered seriously by organisations to be included as part of a defence-in-depth approach to malware.

With simple modifications SMB Lure can easily be turned into a semi-automated share-aware sample capture system. This is especially useful when Internet facing.

Anti-virus products should block the WIN.INI modifications, as this blocks the worm itself. This would not be difficult to implement, and the inclusion of this feature would help avoid unnecessary panic and anger among the AV product user base.

Finally, have we seen the last of the Opaserv variants? I do not think so … at the time of writing, my trapped sample count has exceeded 18,000.

FEATURE 3

Infected or Affected, Mobile Users Are Being Plagued

Mary Landesman
Antivirus Guide, About.com, USA

In October 2002, The Mobile Data Association (MDA) released astonishing figures for SMS text messaging use in the UK. According to its report, 'The total number of chargeable person-to-person text messages sent across the four UK GSM network operators during September was 1.43 billion.' As the report explains, this translates into an average of two million text messages per hour.

Clearly, text messaging could be considered a lucrative industry. If these had been *Verizon Wireless* customers, the total revenue generated might have been as much as $240,000 per hour – or $5.7 million per day. Certainly not chump change. It's exactly these types of figures that have American wireless executives salivating at the prospect of engendering the same enthusiastic use in the US.

The Radicati Group estimates that the number of SMS users will rise from 576 million in 2002 to 1.36 billion by 2006. Analyst *IDC* forecasts the number of wireless SMS messages soaring from 1.4 billion messages in 2002 to a whopping 42 billion in 2006.

Whether these figures are accurate remains to be seen. *Radicati* also estimates computer virus damage costs for 2002 in excess of $21 billion (virus writers had better get busy), rising to over $56 billion in 2006. The reality is that, while sometimes such forecasts turn out to be little more than indicators of the fact that the lens being looked through is a little cloudy, they do provide a valuable barometer of trends. And all forecasters agree that, since cellphone use in the US – and in particular SMS text messaging – falls far behind that in Western Europe and some Asian countries, the United States presents a significant growth opportunity.

Derailing the Gravy Train

However, there's a dark side to SMS text messaging that could derail the gravy train. Users of SMS devices are increasingly being victimized by both spam and email worms. Currently, of course, email worms don't spread *from* mobile devices, but they can sometimes be sent *to* them – and these messages bring tangible, hard costs to the victims in the form of a monthly bill.

Deborah Tapp, owner of the 'Better Than a Taxi' service in Suffern, New York, knows first-hand the pain of Klez. To keep track of business, her email is forwarded automatically to her mobile.

According to Deborah, she is inundated with Klez. She explains, 'My service includes 500 [free] text messages and I purchased a plan for 500 more ($2.99) each month. Everything in excess is billed at 5 cents each. My excess bill (from Klez) was $75 this past month!'

Though Deborah complained to *T-Mobile*, her service provider, she described them as unsympathetic. Other SMS messaging users report similar experiences, receiving suspiciously worded text messages which contain familiar wording such as 'A Humour Game' or 'A Funny Game'. Deborah has resorted to blocking the messages, but as each is slightly different, the reactive approach has been of little help. Currently, Deborah's plight, and that of others like her, remains unresolved.

Warnings

While Klez did not target SMS text messaging users *specifically*, viruses that do target users of this technology directly may not be far off.

In a February 2002 press release concerning the JS/Coolnow.A IM worm, Natasha Staley, anti-virus consultant at *Sophos*, warned, 'Instant messaging platforms may be a fast and convenient way of keeping up to date with your friends, but they can also be used for virus transmission. With an increasing number of worms infecting IM applications, managers should ensure that only those with a legitimate business purpose are allowed access to these platforms.' It seems some were not paying attention to Ms. Staley's caution.

In June 2002, *Microsoft* teamed up with several European providers to offer Instant Messaging forwarding capabilities to mobile devices, for a charge. If a user is offline when Instant Messaged, the message will be sent automatically via SMS messaging.

Regardless of whether Instant Messaging worms can *infect* SMS text messaging users, it certainly seems they could deliberately *affect* them, in much the way Klez seems to have done accidentally – and for a fee. In his *Vnunet* article 'Instant Messenger goes mobile', Nick Farrell estimated that as many as 31 million subscribers could be affected: the new service was adopted in Belgium, The Netherlands, Switzerland, Denmark, Austria, Turkey and Norway. Fortunately, it appears that some did heed the *Sophos* warning – according to Nick, the UK chose to opt out of the plan.

Of course, a costly DoS from IM worms is not the only drawback to such services. While IM is itself a non-secure medium, consider the even greater security risk involved in having corporate communications forwarded in plain text to mobile devices.

The case of Philip Nourse, recently convicted for (among other things) persuading two *mm02* employees to forward his ex-girlfriend's text messages to him, underscores the vulnerability of non-encrypted data sent through third parties. The problem of espionage may increase if Instant Message forwarding is adopted by the corporate world.

In their report 'Corporate Messaging and Collaboration Deployment and Procurement Plans, 2002–2004', *The Radicati Group* disclosed that 45% of businesses surveyed used Instant Messaging. How many subsequently adopt SMS forwarding remains to be seen, but history has shown that the more prevalent the technology becomes, the greater the likelihood of its becoming a target for miscreants.

Bargain Advertising

Coupled with the malicious code and security risks affecting mobile devices is the ubiquitous problem of spam. Mike Musgrove of the *Washington Post* reported on this problem as far back as April 2000. With the increase in technologies such as *Outlook SMS* from *Upside Wireless*, which makes sending SMS text messages via email a mass-mailers' dream come true, the problem can only worsen. This is not to say there are not legitimate business reasons for having such a capability; rather that spam may be an unfortunate side-effect of these advances. The same can be said for websites that offer free sending of text messages, many of which have been reported to collect, and later sell, the SMS addresses to bulk mailers. Even *AT&T* unwittingly contributes to the problem by issuing predictable, sequentially numbered IDs followed with @worldnet.att.net, providing a virtual harvesting opportunity for the spammers.

There is little doubt that spam itself is big business and that SMS messaging provides another tier of opportunity to bulk mailers. Current response rates to SMS message advertising is estimated to be from ten to fifteen per cent on average. Compared to direct mail (one to three per cent) and email spam (less than three per cent), SMS spam is seen by many as bargain advertising.

Fortunately, as *BBC News* reported in July 2002, there are organizations such as ICSTIS (the Independent Committee for the Supervision of Standards of Telephone Information Services) in the UK which provide some help in reigning in the SMS spam industry. However, in other countries this is not always the case. The same article notes that 'in Japan, where recipients rather than senders are charged for messages, spam is a much bigger problem with nine out of every ten messages on the *DoCoMo* network estimated to be spam.' Even those protected by ICSTIS suffer, as 'unsolicited texts dupe people into phoning premium rate numbers' using such ploys as 'a romantic message from a mystery admirer.'

Clincher

The market clincher may be that text messaging in parts of the world other than the US caught on long before the worms and spammers jumped into the fray. Early adopters were able to cosy up to the technology and embrace the power of it prior to being presented with the unpleasant aspects. Not so in the US where adoption has been slow – creating a situation wherein new customers may likely be greeted by miscreant text long before their first 'SUP' or 'HRU' comes in from a friend.

Already overwhelmed with telemarketers on their phones, viruses and spam clogging their email, and junk mail in their mailbox, will the US population embrace this medium? Or will they view it as hostile and intrusive, thus shutting down a fledgling industry before it even takes flight?

In his 1995 book *The Road Ahead*, Bill Gates forecasted that senders of unsolicited mail would be required to pay the recipient to read it. That prediction has not come to pass, though one has to wonder if a hoax will soon be developed around the notion. Nor has it been possible to stop ISPs effectively from granting the necessary circuitry through which spam flows. Though both *AT&T* and *PSINet* were embarrassed two years ago when it was discovered they were entering into lucrative 'pink contracts' with spammers, it is unlikely that embarrassment alone – or fear of discovery – will prevent such behaviour from occurring now or in the future with other ISPs. Further, by taking advantage of open relays and other vulnerabilities, spammers have means other than willing service providers to advance their cause.

The Radicati Group projects anti-virus revenues for 2002 will approach $1.5 billion. They project content filtering and anti-spam revenues for the same year at $460 million and $88 million respectively. It is not surprising that anti-virus revenues are strongest, considering both real and perceived damage costs justify the need for such protection.

Will spam prove to be the same double-edged sword as the virus writers, on the one hand creating a plague and on the other an industry? One has only to look at the strategies employed by the miscreant FriendGreetings to recognize that the methods employed by the two groups – virus writers and spammers – may not be that distinct from one another. Scampaigns such as these will undoubtedly continue, redefining the term 'malicious' by turning SMS messaging into a potential minefield of explosive charges. If this aspect is combined with DoS'ing worms and security breaches, SMS messaging may find it hard to soar.

Perhaps even worse than SMS failing to take off will be the impact it has on businesses choosing to adopt the technology in spite of the security perils, expense, and DoS capabilities inherent with SMS Messaging. At present, few solutions exist to defend against these risks and most are designed to protect against viruses *infecting* mobile devices and doing nothing for those *affecting* mobile devices. As long as the risks inherent in SMS messaging remain clear, and the solutions murky, the best plan to migrate to SMS messaging may be to make no plans at all.

INSIGHT

Hooked on a Feeling

Larry Bridwell
ICSA Labs, USA

I was born in 1949 in Anderson, South Carolina and grew up in the small textile producing town of Honea Path –population 900 (if you include the dogs, cats, horses and cows). After graduating from high school I moved to Florida to attend Southeastern College in Lakeland, from where (after changing my major course of study several times) I graduated with a BA in secondary education with a concentration in history.

Faith and Firefighting

After I had finished college I remained in Florida. I married my wife Debbie and I spent 12 years working as a professional firefighter in Sarasota, Florida. During my tenure with the City of Sarasota Fire Department, I served as a firefighter, paramedic, and company officer.

I have always been very active in the churches I have attended and I am very deeply committed to my faith. Around 1982–1984 I became aware of either a 'calling' or 'deep desire' (depending on your theological viewpoint!), to return to university and study theology in order to be better able to minister to others. So, in 1985, I resigned my commission with the fire department and returned to graduate school.

I enrolled as a second career student at Gordon-Conwell Theological Seminary in Hamilton, Massachusetts, where I studied theology and education, and in 1986 I graduated *summa cum laude* with a Master's degree in educational philosophy.

My general assumption was that, after completing my Master's degree, I would either continue with my studies or enter the clergy full-time. Although doctoral studies were a possibility, that did not seem to be the best road – I had a wife and three children to support, and our savings had run out. So, after graduating in 1986, I started looking for employment.

I returned to my home town and pursued positions in ministry and teaching. While waiting for a position to present itself, I started my own decorative tile and masonry business and worked in my local church, leading a small home Bible studies course. This turned out to be a six-year 'temporary' situation.

In 1992, a friend contacted me and asked if I would move from sunny Florida to Mechanicsburg, Pennsylvania and take a position as an associate pastor. After careful consideration, my wife and I decided we would make the move.

I enjoyed four years as a professional clergyman. I found it a very fulfilling experience – developing educational programs, planning events, working with volunteers, teaching, counselling, and so on. The highlight of the four years for me was being able to perform my daughter's wedding ceremony.

Introduction to the AV Industry

I got into the anti-virus industry before I had a virus – or at least before I *knowingly* had one. After four years in the clergy I decided that I would like to return to the business world. By coincidence, one of the founders of *NCSA/ICSA Labs* was an Elder in one of the churches I had served. He heard that I was looking to return to the business world and offered me a temporary position as a contracted employee – to help with 'accounts receivable' of all things!

In January 1996 I received an email from an anti-virus developer with whom I was in communication. The email arrived with a .doc attachment and, when I opened it, I discovered that the attachment contained rather more than the information I had requested – I had also received the 'gift' of WM/Concept! Of course, the anti-virus software on my machine detected it straightaway, but I found it an interesting introduction to the anti-virus industry. Needless to say, the developer – never to be named – was somewhat embarrassed by the incident.

When the work on my initial project for *NCSA* was complete, I was offered a permanent position, which I happily accepted. I had become really quite interested in the area of computer security and I jumped at the chance to learn more about it. In August 1996 I was asked whether I would be interested in becoming a consortium manager. *NCSA* had been sponsoring AVPD (the Anti-Virus Product Developers' Consortium) since 1991 and had just launched a similar consortium for firewall developers. I fully expected to be working with the firewall group. However, as soon as I had accepted the offer I was told to get my passport up to date in order to attend the next AVPD meeting – in Brighton, UK, and the *Virus Bulletin* conference. It took only that one meeting for me to become hooked by the industry.

I entered the industry with very little technical knowledge and no security background – some would say I still have very little technical knowledge! I have always felt that if

you are honest with people, listen to them, and treat them with respect, you will find more often than not that they will be accepting and helpful.

So, at my first meeting with the AVPD group, I just told the members who I was, what I wanted to do, and that I needed them to help me do it. And, for the most part, they did help.

Some of those at my first AVPD meeting (the more vocal ones at least) included Alan Solomon, Sarah Gordon, Graham Cluley, Jimmy Kuo, Joe Wells, Rob Stroud, and Glenn Jordan. All of these people and others really did help to educate me in the ways of the industry. I have made many friends in the industry since that time and have truly enjoyed the ride!

Today

These days I am Content Security Programs Manager at *ICSA Labs*. Currently, *ICSA Labs* has three testing and certification programs for content security products: anti-virus products, desktop firewall products, and Internet filtering products. My primary duties are to manage the individual product group consortia, develop and maintain certification criteria, and manage special projects for the product groups.

I have managed the AVPD (which deals with the testing and certification of anti-virus products, consumer education and special projects) since 1996. Also, I helped to launch the *ICSA*/*Microsoft* MVI (Macro Virus Initiative) in 1997 – an initiative to facilitate communication between *Microsoft* and the AV industry and to provide a reliable means of transferring information both proprietary and public from MS to the anti-virus industry. I still manage the MVI web forum and email list. I have been a member of EICAR since 1997 and am involved with several public and private email lists on viruses and security.

Home Life

Outside of work, I enjoy both archery and fishing on occasion, but above all I enjoy spending time with my wife and with my family.

Debbie and I have been married for 31 years. We have three children: Margo, who is married and has two children; Erik, who is married with one child (the youngest of our grandchildren – born November 2002) and Joel, who is in college. Sadly, I don't get to see my grandchildren as often as I would like to since Debbie and I live seven hours away from them.

AV Industry Views

I believe that the anti-virus industry will continue to change and adapt as the technical environment evolves – as it has done since its birth. I believe we will continue to see improvements in current anti-virus detection and recovery technologies, but the changes that are on the horizon in the way people and businesses use technology, new devices, faster access in both wired and wireless networks, etc. will result in the development of new and better anti-virus protection than that which exists currently.

The anti-virus industry itself will survive as long as the people and companies within it remain flexible and willing to adapt, and as long as they continue to cooperate with one another to meet the challenges presented by new viruses and malicious code.

I am certain that virus writing will remain with us. The relative ease of editing existing script viruses to 'create' new viruses or virus variants, coupled with the curiosity of youth, will continue to produce virus writers.

However, I believe we can no longer stereotype virus writers as 'pimply faced teenagers' – as demonstrated by the arrest and prosecution of the Melissa author David Smith. Authors of malware come in all ages and genders, with varying levels of programming skills.

I believe that we have seen some slowing of writing and releasing viruses as a result of the legal actions taken against a few virus writers in recent years. Whether this trend will continue has yet to be seen. I think that the mere fact that some jurisdictions are even considering legal action helps – while it may not be *the* answer, it is certainly a deterrent to some.

I think that sentencing should probably be meted out based on the individual case. Things to take into consideration may be the writer/distributor's age, intent, type of damages, extent of damages, and so on. However, I do not believe that legal actions alone will stop the practice of virus writing completely. After all, it is illegal in most places to write graffiti on public buildings, break street lights, or throw litter from an automobile, but it still happens.

ICSA Labs' annual virus surveys seem to indicate that companies are increasingly using anti-virus products at the desktop, file server and the email gateways, as well as using more filtering techniques to block or quarantine malware or email attachments of certain file types. Awareness is there, but more user education is needed.

Future

I believe the current AV technologies and products do a very good job of protecting us from known viruses and much of the new and previously unknown malicious code. I do hope to see improvements in more generic types of virus protection going forward to help guard against the new malware that will inevitably come.

I truly enjoy the work I do and the people with whom I work both at *ICSA Labs* and in the anti-virus industry and I have no plans to leave either. My work has given me the opportunity to meet some quality people throughout the world, create strong friendships with them, and to be part of an industry that seeks to be a help and service to others. I find that fulfilling as well as challenging.

PRODUCT REVIEW

Ahnlab V3Net for Windows Server SE

Matt Ham

Ahnlab is one of a number of vendors whose products are recent additions to *Virus Bulletin*'s standalone and comparative tests. Korean in origin, *Ahnlab* specialises in anti-virus software, with an additional interest in the PKI security market.

By anti-virus company standards *Ahnlab* is a relative newcomer, having been founded in 1995. Initial appearances, however, are those of a well established company and a mature product. The company may not be as well known to readers as many others in the industry – this can be attributed almost entirely to the fact that *Ahnlab* is Korean in origin. Although the company has recently expanded into the Western hemisphere, its geographical origins raise the potential for differences in both approach and implementation of the product, thus making it an interesting subject for review.

As far as products are concerned, *Ahnlab*'s focus is very much towards *Windows*-based systems. There is a *V3Net Group* product for *Notes*, though this operates only on a *Windows*-based *Domino* server. This *Windows*-centric approach is more common in those companies which have entered the anti-virus market relatively recently, and is understandable in that *Windows*-specific scanners can be produced by re-using code directly from, for example, schedulers or reporting tools.

The lack of cross-platform capabilities may be convenient for developers, but is not a great selling point where companies with diverse operating systems in place are concerned.

Web Presence

Ahnlab's main website is at http://www.ahnlab.com/, which acts as a portal into several country-specific sites. The company has offices in Korea, China, Japan and the United States, while Brazil and Australia are home to distributors.

The introduction to the website consists mainly of recent press releases and virus news and alerts, with direct links to a somewhat non-standard selection of recent viruses – presumably reflecting the *Ahnlab* perspective of what is important in the world of anti-virus.

Downloads are remarkably limited in these days of free trials for almost every piece of software. Other than these minor points, however, the website is useful and fairly similar to others of its type.

Documentation and Help

Rather than being context-sensitive, the internal help file provided with *V3Net* is of the monolithic document sort. However, it is quite brief in its descriptions of the product's functions and settings, though this policy has both advantages and disadvantages – while thoroughly detailed information is provided in some larger help files, wading through such information can prove tiresome.

Many of the features within *V3Net* are named in an original or potentially ambiguous way and, as such, it was necessary to refer to the help file regularly, giving it a good test as to the adequacy of its comprehensiveness.

Where it was necessary to use this function in the main program, the brevity of the descriptions did not impede their ability to provide the information required. However, the help file was found to be lacking in two major places within the program.

Since there are many configuration settings whose function may not be apparent – neither would the circumstances under which they might be useful – it would be useful to have a help function available during setup. The lack of an accessible help function both here and in the areas where program configuration is set after installation was a noticeable and regrettable omission.

The manual provided was stated to be an early copy of a more recent version and came with warnings that the translation might not be of the highest standards. The most amusing instance of poor translation comes early in the document, with the statement, 'This user's guide is written only in Korean' – which would seem a classic example of an oxymoron.

The manual is home to many all-too-familiar declarations of quite how good the scanner is, stating that the product has the fastest update and scan engine capabilities 'among many domestic and international products'. Whether these statements are quite as self-promotional and debatable in the original language is unclear. There is also a guide to WARP technology in *Star Trek* – which is a little bizarre, even taking into consideration the fact that the *V3Net* engine is named WARP.

The technical contents of the manual are less likely to cause controversy. The description of installation for the product and the associated walk-through and screenshots go a long way to nullify any complaints about the lack of online help available for this stage of the process.

Similarly, the descriptions of features and procedures is far more detailed and organised than that in the online help, and is complementary rather than being simply a transposition of the same information. The contents are fully

hyperlinked in the electronic version of the manual, which is always very useful, although the same has not been done in the index. However, the index is more complete than those typically found in anti-virus manuals.

An FAQ and glossary are also included in the manual – though these are prime candidates for checking by *Ahnlabs*' translators. Overall, however, the manual is useful and sufficiently comprehensible even in this beta state.

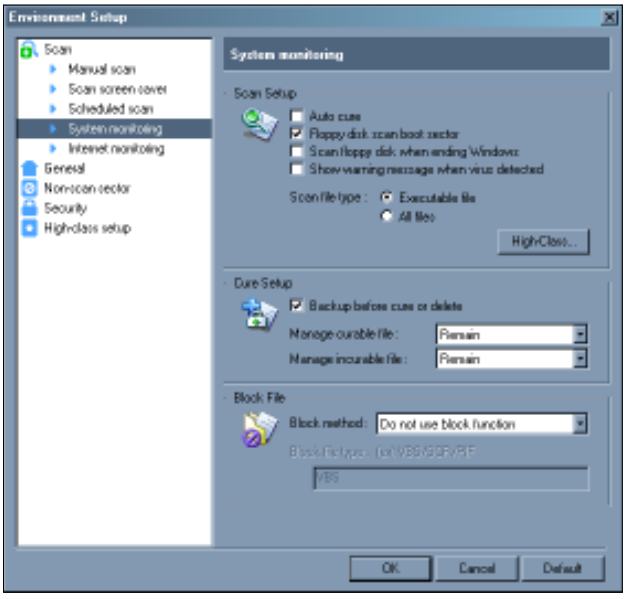
Installation

The server product, *V3Net*, has remarkably light requirements as far as hardware is concerned, claiming that it works on machines as feeble as a 486 DX. However, since there is also a requirement for *Windows NT 4 Server* or higher, it is difficult to imagine that such a slow machine will ever be a realistic target for installation.

In this case *V3Net* was installed on *Windows 2000 Server*, with the usual licence agreement and the inputting of serial numbers starting off the process. At this point, where many products have an exhaustive list of options, *V3Net* asks only for confirmation of the install location before installation commences.

Since the test server is not connected to the outside world, registration on the *Ahnlab* website failed, both during installation and when triggered as a result of a dialog at the end of installation. After registration (or lack thereof), options are offered to 'Execute a Smart Update' and 'Execute a Scan Window'. These offers were declined, and at that point installation was complete as far as file transfer was concerned.

The lack of configuration options during the installation process is made up for after the event, with the Environment Setup Wizard being launched automatically straight after installation.



As a default this wizard offers a scanner which remains visible after a clean scan, does not attempt disinfection automatically and does not scan inside compressed files. Since this is a server product it would seem very likely that at least the compressed files option would need to be changed in a real-world installation. In its default state the product is set to scan executable files rather than all files.

Other default settings refer to the configuration of disinfection options – which, surprisingly, are not greyed-out when disinfection is inactive. The options offered here extend to whether backups should be made before disinfection or deletion, and individual settings for disinfectable, non-disinfectable and compressed files. Again, the compressed file settings were not greyed out when compressed file handling was inactive.

Somewhat strangely, no disinfection option was visible for the compressed files, though a separate option provides the opportunity to recompress disinfected ZIP files. This looked to be an area in which experimentation could be profitable.

Similarly obscure were the options to 'manage executing file', where the choices available were 'upon user confirmation' (the default), with alternatives of 'remain', 'cure upon termination' and 'reboot after cure'. Further references to 'executing files' led me to believe that this is a slightly garbled translation of executable file. Unfortunately there is no help function at this point in installation to confirm my assumption.

In addition to the immediately obvious options, the setup wizard offers a sub-dialog entitled 'High-Class'. This strange name is used to define scanning further for compressed files and the file types to be scanned. In this case the compressed-file commands are greyed out when compressed file scanning is deselected (rendering all the more mysterious the lack of such a feature in the main wizard). By default this offers: scanning inside multi-compressed files to only one level; scanning of executable compressed file types, such as DIET and LZEXE; and incremental scanning so that only changed objects are scanned.

As for compressed file types, 27 formats are supported, of which ACE, ARJ, LZH, RAR and ZIP are the total of those activated if compressed scanning is activated in its default setting.

Settings for executables in the default state include categories for Executing File, Macro and Script, all of which can be selected independently. To these blanket categories can be added user-defined extensions, or indeed the blanket categories may be replaced entirely by a custom list.

The second page of the setup wizard deals with the scanning activities that are to be activated. In a default installation system monitoring (the file-scanning on-access function), *Explorer* integration and scanning of files downloaded through web browsers are all active. Scanning can also be set up as a job which starts when the screen

saver is activated – areas to scan can be user-defined. Finally in this page Office Protector can be activated – a feature which offers automatic scanning of files in both *Office 2000* and *IE 5.0* in addition to the nebulous area of '(an)other environment'.

A third page follows this part of the configuration process, this relating to the activation of scheduled scans. Targets and times may be selected, much as would be expected. User details for the scan may also be added, so as to activate scanning with sufficient rights. Time periods can be set from once only, through monthly to daily or upon machine startup.

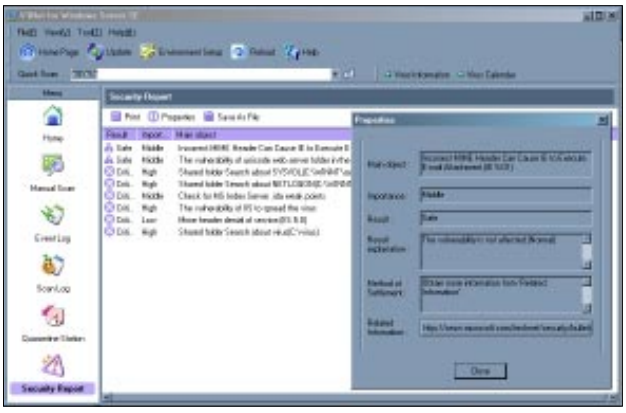
Next in the wizard's pages is the selection of areas to be excluded from scanning – which is unremarkable. The page that follows offers security functions, all of which are inactive by default. The reality of this is rather less grandiose than I had imagined, being simply a method for applying a password to the uninstallation or deactivation of real-time monitoring for *V3Net*.

With the completion of this page of the Environment Setup Wizard the installation process is complete. This results in two subsections being added to the start menu, one offering the Smart Update Utility and its uninstaller, while the other offers Environment Setup plus *V3Net*'s main program and an uninstaller for the main program. A direct link to the main scanner is also installed to the desktop.

Updates

The update features present in *V3Net* can be activated in two ways, either through the use of executable patch files or through downloads from designated sites. These latter sites may be located either on a networked machine or the *Ahnlab* site.

The update via patch method worked perfectly and involved simply the execution of the patch file. The local download method involves the construction of a local directory, the files within being derived from the patch – a task performed from within the updater. However, use of this installation download directory or the website equivalent was fraught with problems when it came to transferring parts of the update. This process did not successfully complete and the *Ahnlab* technical staff were consulted. At the time of publication discussions were still ongoing.



Features

Launching *V3Net* results, when appropriate, in a reminder to update the definition files. When this has been passed the GUI opens, with the layout being the, by now, classic left-hand bar with icons controlling the content of a right-hand pane. There are minor additions to this layout in the default view, however, since a selection of drop-down menus and various controls are accessible at the top of the screen.

A warning for the faint-hearted ought to be issued regarding the colour scheme used by *V3Net*. Although the default view is a standard pale blue, selecting other controls on the left-hand bar changes not only the content of the right-hand pane but also the colour scheme. The colours include a virulent lime-green and a particularly fetching violet.

As has become customary, the investigation of features began with the assorted drop-down menus and icons. The drop-down menus are divided amongst File, View, Tool and Help.

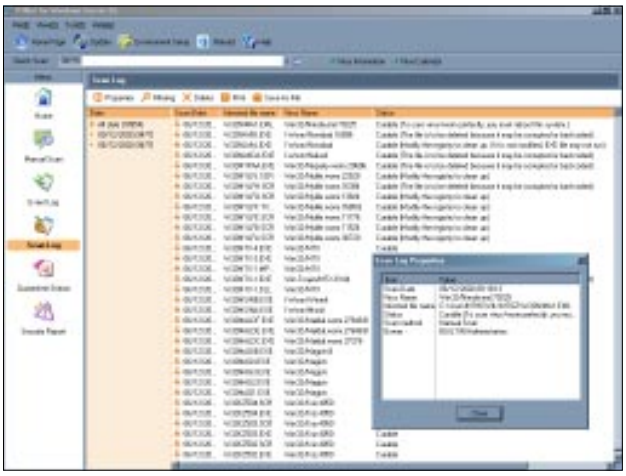
Of these, File is limited to the option of an update, discussed above, and exiting the program. View is a method of swapping between the contents of the main display panes without the use of their associated icons, and also offers hyperlinks to virus information and virus calendar web pages. That these are links is not particularly surprising given that this is designed as a server product and that broadband penetration is more advanced in Korea than in any other country.

The Tools drop-down menu is a little more varied. First, this may be used to toggle the activation of scanning while screen savers are activated and also within *Explorer*. In addition it can be used to toggle real-time scanning both of the system and of Internet activity. A link to the *Ahnlab* homepage is another addition to the random collection of features in this category. Finally, scheduled scans may be set up from here and the Environment Setup application launched.

The last of these drop-downs is entitled Help and launches the help feature. Online registration can also be performed here and program installation viewed. The program information area is home to a classic mistranslation, with 'serial number' rendered here as 'consecutive number'.

With all drop-downs out of the way, the top icon bars can be inspected. As is often the case, several of these icons offer functionality which is also covered by the drop-down menus. Of these a Home Page link, Update, Environment Setup, Virus Calendar, Virus Information and Help option may thus be passed by quickly.

This leaves only two objects, the first of which, Reload, can be described briefly, since it is simply a method of refreshing the view which is currently in the main pane.



This leaves only the Quick Scan option, which provides a means by which areas may be scanned by means of standard DOS command line parameters. As an example the default value in this command was C: /a /s.

This covers the commands available from areas outside the main control where the view selections are entitled Home, Manual Scan, Event Log, Scan Log, Quarantine Station and Security Report.

Home, as the name suggests, is the default starting view and is more informational than control-dominated. Information provided includes the current engine version, the status of on-access monitoring for system and Internet scans and both scheduled updates and scans. Information is also provided concerning the last performed scan. The scan, update and scheduler status information links to the appropriate control area for each of these features.

The next view, Manual Scan, does not increase the amount of control available as significantly as might be expected. Unlike many competing products the interface here is minimalist, with the controls being simply for selecting the areas to be scanned and for scanning those areas.

In-built area selections provide the option to select individual drives, folders or local hard drives. Additional groups can be constructed and saved by the user. Somewhat strangely these user-defined scans are added to the pre-existing selection of selectable drives rather than being immediately visible from the interface.

Although unpleasantly lurid to look at, the Event Log view contains complete details of scans, updates, the starting of on-access scanning and exit codes for certain applications. Filters may be applied so that only events from certain days or time periods are displayed, and the log file entries may be viewed separately for the categories of normal, error or warning. For administrative purposes the log can be saved, printed or cleared.

Still on the log file theme, the Scan Log view offers all the options for filtering and administration that are offered by the Event Log view. In addition, a Properties feature is included. Saving to file is a speedy process and can be delivered as the standard comma and tab separated text.

Details available in the log files are scan date (which includes the time), file name, virus name, scan method, owner of the scan process and status. Status is most often the relatively simple 'curable'. Filtering by incurable, i.e. files that are unable to be disinfected, produced all results that were labelled as curable with some proviso. These were mostly worms with a selection of older standard set samples. In the first case the curable status came with the proviso that the Registry must be modified in order for full clean up to occur, while in the latter case most of the files were labelled as corrupted. In some cases the descriptions had clearly been crafted specifically for the particular virus family. For example, W32/Qaz has a status description explaining that notepad.com must be renamed as notepad.exe after disinfection.

One oddity in the Status field is that, in many cases, a specific virus name is given for a sample, but the status is set as 'New Virus', along with a request to send the file to *Ahnlab* for further analysis. On a more functional level, it is somewhat irritating that actions cannot be taken upon files marked as infected from the Log view.

The penultimate view is the Quarantine Station. This view is very similar to the Scan Log view, though additional features include restoring to the origin of the file, moving to a temporary folder or permanent deletion.

Finally we reach the Security Report View, without having spied the usual morass of configuration to be seen in a scanner GUI. Though not directly related to the traditional concept of an anti-virus scanner, this type of feature is becoming increasingly common in products as a pre-emptive precaution against new threats.

Seven possible security holes were listed which could be used by malicious code. On the machine running *Windows 2000 Advanced Server* with *SP2*, two of these potential vulnerabilities were declared to be safe, while a further five were declared unsafe, with two of these being instances of shared folders.

In cases where vulnerabilities are detected, a brief overview is given of what the possible implications are and a direct link is supplied to an appropriate patch. One level of control is offered, that is to apply the previously chosen action upon a selected portion of the report list, rather than the alternative of the entire list.

Features – Environment Setup

It has been noted at several points in the review so far that the number of controls supplied within *V3Net* is minimal. The reason for this becomes apparent when the sister program, the Environment Setup, is examined. This program is used for the setting of the majority of those options usually found within the main body of a scanner.

The Environment Setup program is also very much linked with the functionality of the Environment Setup Wizard. In this case, however, the pages encountered in the installation are bound together with other controls, resulting in a GUI featuring the omnipresent split pane with left-hand view controls. Controls in this case are divided amongst Scan, General, Non-Scan sectors, Security and High-class setup.

As overall controls, accessible at any point, the program may be exited, defaults applied or current changes cancelled. One distinct drawback is the absence of any help function within this program.

Scan has its own right-hand pane view, in addition to five sub-views. The parent view contains on-access scanning activation for all relevant areas. In addition it further refines what exactly the scanning of *Windows Explorer* entails when activated.

Contracted Menu and Tools may be selected independently for access, though it is not readily apparent what each of these options actually does. Reference to the manual revealed that the contracted menu is the ability to scan from a right-click of a file. Tools, on the other hand, adds a V3 icon to the *Explorer* interface which can be used to trigger updates, launch the main scan engine and similar *V3Net*-related features.

The sub-views also provide further control over scan tasks. On-demand and screen saver scans can each be configured separately with the same choices available here as during installation. The similarity also extends into the sub-view for Scheduled scanning.

The views for System Monitoring and Internet Monitoring, on the other hand, are very much new territory, despite the overall on or off activity of these features having been defined already in other control areas. System monitoring is the traditional on-access scan within *V3Net* and in its original configuration offers scanning of executable files and floppies. Automatic disinfection, scanning of floppies when *Windows* exits and the option to show a warning message if a virus is detected can be added to the default settings.

Continuing these options, the action to be taken on infected files is determined here and files can be designated to be blocked from execution automatically, based on extension. In addition there exist 'High-Class' options for defining more precisely which file types are to be scanned, adding user-defined extensions to the scan list and adding files which are not to be scanned based on extension.

Where Internet monitoring is concerned a limited subset of these options – automatic disinfection, scan executable or all files, high-class options and treatment of infected files – is offered.

General setup is the next view available. Most of these options are concerned with the operation of the various log files within the program. Logs – which include the previously mentioned scan log and event log – are on by default and may be activated and their size limits set.

There is also a control here for the maximum size of the back-up folder, equivalent to a quarantine. Also activated in the default settings are the options of showing event messages, displaying an icon on the task bar and making a sound if a virus is detected. These are also removable at this point. User-defined sounds may be assigned to viral detection, though for testing purposes the sound was turned off.

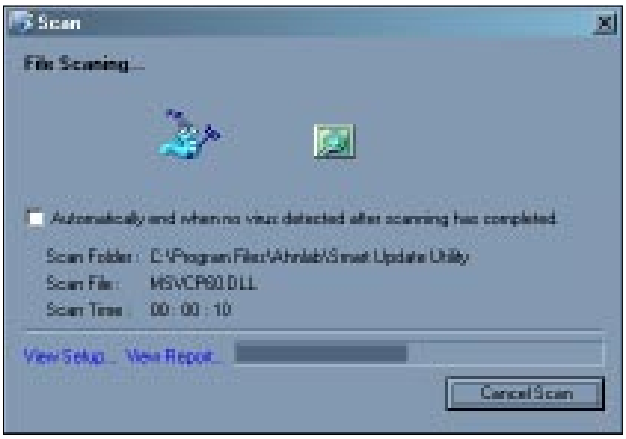
The exclusion settings, here termed Non-Scan Sectors, and Security Settings views are identical to those encountered in the installation process. This leaves another High-Class setup view – not to be confused with those accessible elsewhere. This commences with macro disinfection settings. By default all macros within a file will be deleted when a macro virus is detected – a situation which may or may not be desirable depending on the organisation. It is possible, therefore, to set this so that only the viral macros are deleted.

The High-Class options include the settings that determine which objects are to be scanned by default when scanning on demand. These areas include memory (which cannot in fact be deactivated), *V3Net* itself and the boot sectors of the machine. More confusingly, there is also a setting to scan 'process/Back Orifice' which appears to be for scanning processes active at the time – though the addition of Back Orifice is slightly odd.

Operation

First, simply as a test of reporting and log files, the scanner was set to check the whole of the *VB* test set. Default settings were used, with the following exceptions: all settings for disinfection were disabled, sound was disabled on disinfection and the on-access scanners were disabled.

The report file generated was not vast, being simply a breakdown of number of files scanned, infected and



disinfected divided by MBS, DBS, Executing Process and File, the first two clearly being boot record scans. With the large size of the test set involved it was not a great surprise that the process of exiting the report area was rather slow (since this is the point at which the log file is generated).

The other option at this point is to disinfect files that are found to be infected. This is preceded by a disclaimer – which introduces some possible frustration, since it informs the user that treatment will be according to the options defined in the Environment Setup and does not offer any opportunity to change these on a case-by-case basis.

Tests

For detection testing a scan was performed over the *VB* test that was used in the most recent comparative review (see *VB*, November 2002, p.16).

The scanning process is accompanied by an animated pitchfork-wielding worm which progresses towards a computer monitor where it is transformed into a halo-wearing version of itself. Whether the transformation is to angel or ghost is not apparent and the creature soon vanishes into the heavens as the animation restarts.

Such distractions aside, the scanning rate on infected files was good, though W97M/Splash.A, a macro virus which inserts junk to provide ever larger new infectious versions during replication, showed definite sluggishness in the larger samples.

Detection was the mixed performance that can, by now, be expected of a program that is new to the *Virus Bulletin* tests. Performance in the ItW test set was very good, with detection very much reduced in other areas. Of particular note were misses of large sections of the polymorphic set and older viruses in the standard set.

By and large, detection rates were higher in the macro set and amongst the newer standard samples, though the ItW set showed by far the highest detection rate. Since this product is due to be tested in forthcoming *VB* comparative reviews, a detailed analysis of scanning detection results was not performed.

The standard clean set tests were used for testing the scanning speed of *V3Net* over clean files. The scanning engine date, hardware and platform used in these tests were all appropriate for inclusion in the November 2002 comparative review, making results here comparable directly with results from that review.

Scan times in the default setting were at the faster end of the results attained over a range of products: 60 seconds for the files in the executable clean set and 11 seconds for the files in the OLE clean set. Since archive handling is not a standard feature this was activated for the scanning of the ZIP archived portions of the clean set, the times here being 287 seconds for clean executable zips and 68 seconds for clean executable OLE files. No false positives were encountered.

The setting for the cleaning of archive files was also examined. For these tests ZIP archives containing W32/Heidi.A were used, as this worm inserts itself within ZIP files as part of its normal activity. The settings were changed so as to include scanning within archives and the reconstruction of archives after scanning. When scanned using deletion, the result – as would be hoped – was that all archive files were left intact, and the Heidi file was deleted.

Conclusion

The overall appearance – including the colour scheme – of *V3Net* are very typically Korean in nature, though its features are common to all products of its type.

The feature set in *V3Net* is not quite standard, however. For example, the lack of some method of control over the configuration of the program from the scanning interface is almost unique amongst current anti-virus programs. (Of programs that are reviewed in *VB* regularly, only *Norman Virus Control* has as radical a control method.) This is something of a double-edged sword. If the same scans are to be run repeatedly, the lack of clutter in the scanning GUI is well worth the need to set a single scanning policy in advance. For users who wish to change configuration frequently, however, the same feature is more of a hindrance than a help.

The product's detection rates are comparable with other recent additions to the *Virus Bulletin* testing stable and only time will tell whether VB 100% awards will be thick on the ground or just out of reach for *V3Net*.

Technical Details

Test environment: Three 1.6 GHz Intel Pentium machines with 512 MB RAM, 20 GB dual hard disks, DVD/CD-Rom and 3.5-inch floppy drive, running *Windows 2000 Server Service Pack 2*.

Developer: *Ahnlab Inc*, 8F Valley Bldng, 724 Suseo-Dong, Kangnam-Ku, Seoul, Korea 156-744.

Tel +82 2 2186 3000; Fax +82 2 2186 3001; email v3support@ahnlab.com; web http://www.ahnlab.com/.

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, 6 Kimball Lane, Suite 400, Lynnfield, MA 01940, USA

Tel (781) 9731266, Fax (781) 9731267

END NOTES AND NEWS

The Black Hat Windows Security 2003 Briefings take place 26–27 February 2003 in Seattle, WA, USA. The Briefings comprise six tracks across two days and follow two days of Black Hat Windows Security Training (24–25 February). Register before 16 January 2003 for reduced registration rates. See http://www.blackhat.com/.

The 12th Annual SysAdmin, Audit, Networking and Security Conference (SANS) takes place 7–12 March 2003 in San Diego, USA. The conference will feature 12 tracks, night activities, a vendor exhibition, and additional special events. See http://www.sans.org/.

Infosecurity Italy will be held in Milan, Italy, 12–14 March 2003, for details see http://www.infosecurity.it/.

CeBIT, one of the world's largest information technology trade fairs, runs for one week in Hanover, Germany from 12–19 March 2003. All aspects of IT are catered for, with well over 7000 exhibitors. For full details see http://www.cebit.de/.

SACIS Expo (Security, Audit & Control of Information Systems) takes place 25–26 March 2003 in Istanbul, Turkey. Hear about the latest information security and audit developments from IT security professionals, and meet with product developers and academics. Early registrations qualify for a discount of up to 20%. For details see http://www.smartvalley.net/sacis/.

RSA Conference 2003 takes place 13–17 April 2003 at the Moscone Center, San Francisco, CA, USA. General sessions feature special keynote addresses, expert panels and discussions of general interest. Optional tutorials and immersion training sessions will provide the basics of e-security technology, enterprise security and security development techniques. See http://www.rsaconference.net/.

Information Security World Asia takes place 23–25 April 2003, at Suntec Singapore. For details of what is claimed to be Asia's largest and most dedicated security technology and solutions exhibition see http://www.informationsecurityworld.com/2003/iswa_SG/.

Infosecurity Europe 2002 takes place 29 April to 1 May 2003, Olympia, London. A free keynote and seminar programme alongside almost 200 exhibitors is expected to attract more than 7,000 dedicated security visitors. See http://www.infosec.co.uk/.

EICAR 2003 will take place 10–13 May 2003 in Copenhagen, Denmark. The 12th Annual EICAR Conference combines academia, industry and media, as well as technical, security and legal experts from civil and military government, law enforcement and privacy protection organisations. Call the conference hotline +45 4055 6966/ +44 709 211 1950 or check http://conference.eicar.org/ for details.

Black Hat Europe 2003 takes place 12–15 May 2003 at the Grand Krasnapolsky, Amsterdam, the Netherlands. For more details see http://www.blackhat.com/.

The DallasCon Wireless Security Conference takes place 24–25 May 2003 in Plano, Texas. A two-day wireless security course precedes the conference, including hands-on lab experience and lectures. For full details see http://www.DallasCon.com/.

***Virus Bulletin* is seeking submissions from those wishing to present at VB2003, the Thirteenth Virus Bulletin International Conference**, which will take place 25–26 September 2003 at the Fairmont Royal York hotel in Toronto, Canada. Abstracts of approximately 200 words must reach the Editor of *Virus Bulletin* no later than Friday 21 February 2003. Submissions received after this date will *not* be considered. Abstracts should be sent as RTF or plain text files to editor@virusbtn.com. *VB* also invites suggestions for speakers you would like to hear from at VB2003. Please send speaker nominations, along with details of why you would like to hear the speaker to editor@virusbtn.com. More details, including a list of suggested topics for papers can be found at http://www.virusbtn.com/conference/.

Correction: in *Virus Bulletin*'s Windows 2000 Advanced Server comparative review (see *VB*, November 2002, p.16) the product submitted by *Cat Computer Services* was listed as *QuickHeal X Gen 6.05*. This should in fact have been *QuickHeal X Gen 6.07*. *VB* apologises for the mistake.

The head of the UK's National Hi-Tech Crime Unit has warned that the level of organised crime on the Internet is rising sharply. At the start of the UK's First Strategic Stakeholders e-crime congress, held last month, DCS Les Hynds said, 'I believe we must challenge the existing misguided perception that hi-tech crime is somehow less serious than its mainstream equivalent.' See http://www.nhtcu.org/.