OCTOBER 2008

virus BULLETIN

Fighting malware and spam

CONTENTS

ISSN 1749-7027

IN THIS ISSUE

BUG COLLECTOR

Peter Ferrie continues his series of analyses of viruses contained in the EOF-rRlf-DoomRiderz virus zine. This month's subject is W32/Harumf.
page 4

DIVERSIFY TO GROW

Hannah Mariner shares her thoughts on how allowing people from diverse professional backgrounds to enter the AV industry can help strengthen, prolong and add direction to the industry as a whole.
page 10

VB100 WINDOWS SERVER 2008

This month VB's testing team put 24 anti-malware products to the test on the server version of Microsoft's latest iteration of Windows. John Hawes has all the details on which products managed to secure a VB100 and which have a little more work to do.
page 12

vbSpam supplement

This month: anti-spam news and events, and Terry Zink continues his series of articles on backscatter with a look at some of the methods we have at our disposal to help combat this irritating type of spam.

'We are left with the alarming question as to whether privacy should be put before global security.'

Abhilash Sonwane, Cyberoam

THE COST OF ONLINE ANONYMITY

Minutes before the deadly bomb blasts that took place in Ahmedabad, India on 26 July 2008, an email claiming responsibility for the attacks was received by Indian authorities. The anonymous email was traced to the IP address of an American national living in Mumbai. The authorities now believe that the American's unsecured wi-fi network was used by the terrorists to send the email. The American citizen became a suspect just because he unintentionally left his wi-fi network open and unsecured.

In August, another email about the blasts was received. Investigations revealed that a proxy server was used to send the email. With some help from the service provider that hosted the server, investigators were able to determine that the email originated from an educational institute in the city of Vadodara. Analysis of the logs of the institute's unified threat management appliance enabled the investigators to trace the email to an internal IP address belonging to the institute's computer lab. Innocent students and faculty members were questioned as suspected terrorists.

More recently, Internet activist group 'Anonymous' was responsible for hacking into the *Yahoo!* email account of Alaskan governor and US Republican Vice-Presidential candidate Sarah Palin. The web proxy service ctunnel.com was used to by the group to preserve its anonymity.

These days, a large number of public places (airports, restaurants, cafes, hospitals and so on) offer free wireless networks. Home networks are often left open and unsecured by their users, because the average home user doesn't understand the technology and either leaves the wi-fi device in its default configuration or else does not configure it securely.

Criminals can simply sit in their cars outside a house, an office or a hotspot, and use the unprotected wireless network to carry out their sinister activities anonymously. The online activities of 'war-driving' criminals can be traced only to the IP address of the house, office or hotspot, putting the innocent home owner/office/hotspot manager under suspicion because of an insecure network configuration.

In the past, intelligence agencies could catch criminals based on the IP addresses of the emails they sent. The hard drives of the computers suspected of having been used for illegal activities provided the physical evidence needed to link the action to the criminal. However, new technologies are making it difficult to gather evidence.

Anonymous proxies enable criminals to conduct their online activities without revealing their real IP addresses. If the authorities want to trace the IP address of someone who has used the anonymous proxy they need the logs of the proxy server. The jurisdiction in which the proxy server is physically located plays an important role here. If it is located outside the jurisdiction of the investigating authorities, they have to rely on the cooperation of the local authorities at the other end, which can result in a dead end for the investigation.

Privacy is a basic human need and should be respected for every Internet user. However, as the movement for online privacy gathers pace, we are left with the alarming question as to whether privacy should be put before global security. The abuse of anonymity on the Internet is affecting many innocent lives, and victimizing Internet users.

Technology and the law need to keep pace with one another and with the changing times. The need of the hour is to engineer better technologies and frame better laws that allow users to enjoy their privacy while at the same time enabling authorities to trace criminal activities. But until that happens, there are several measures that can be taken by responsible citizens and corporations. For example, the hospitality industry should desist from providing Internet access without valid identity checks (mechanisms are available that allow this). The ISPs and vendors should undertake campaigns to educate home users as to how to configure wi-fi access points securely.

Cyberspace will continue to evolve and criminals will continue to look for new ways to abuse the loopholes left by technology and the law. However, proactive and responsible engineering and legislation can help prevent the misuse of technology.

NEWS

NO PHORMAL INVESTIGATION

UK police have decided not to undertake a formal investigation of telecoms company *British Telecom* (*BT*) and *Phorm*, the company it engaged to gather information on the web-surfing habits of its customers.

In July 2007 *BT* began a test with *Phorm* who used deep packet inspection at the ISP level to gather information on subscribers' web-surfing habits and subsequently deliver tailored advertising content. *Phorm* claims that it scrubs the content it stores of any personally identifiable information, and that it can also act as an anti-phishing measure as it prevents users from accessing sites on a list it maintains of known phishing sites. However, the problem for many was that the test was performed without the knowledge or consent of *BT*'s user-base.

Campaigners who took umbrage at the secretive actions of *BT* and *Phorm* compiled a dossier of evidence against the two companies and presented it to the City of London Police in July this year. However, following some informal questioning of *BT* executives and a report to senior officers, police interest has fallen and the case has formally been dropped. Police officers reasoned that there was no evidence of criminal intent on the part of *BT* or *Phorm* and that there would have been a level of implied consent from *BT*'s customers in relation to the tests, as the aim was to enhance the company's products.

POP-UP WARNINGS INEFFECTIVE?

A study in the US has suggested that computer users may largely ignore the pop-up windows that are used by some systems (such as *Vista*) to warn of unsafe computing use.

In the study, conducted by the Department of Psychology at North Carolina State University, 42 students were asked to rate a number of medical web pages for clutter – a cover story for the real purpose of the experiment which was to observe how they responded to pop-ups. Each student was presented with four pop-up windows which varied from warnings of programs executing or terminating to a flashing pop-up that added a browser status bar.

More than half of the students simply clicked 'OK' on the pop-up boxes almost automatically. The fact that their reaction times barely varied for the different types of box indicated that they were not bothering to read the contents. More than 40% said they just wanted to get rid of the box as quickly as possible.

While this was not a statistically significant study, it does raise questions as to the effectiveness of legitimate warning messages as well as highlighting once again a lack of awareness among users who seem happy to click on almost anything without a second thought.

Prevalence Table – August 2008

Malware	Type	%
Agent	Trojan	24.79%
Zbot	Trojan	17.39%
Suspect packers	Misc	13.94%
Dropper-misc	Trojan	13.04%
Delf	Trojan	6.15%
NetSky	Worm	5.92%
Downloader-misc	Trojan	4.86%
Autorun	Worm	4.57%
Mytob	Worm	2.82%
Iframe	Exploit	2.53%
Cutwail/Pandex/Pushdo	Trojan	2.53%
Virut	Virus	2.14%
Crypt	Trojan	2.06%
Mdropper	Trojan	1.90%
Bagle	Worm	1.40%
OnlineGames	Trojan	1.24%
Mydoom	Worm	0.85%
Basine	Trojan	0.78%
Zlob/Tibs	Trojan	0.57%
Grew	Worm	0.51%
Zafi	Worm	0.50%
Lineage/Magania	Trojan	0.45%
PWS-misc	Trojan	0.33%
Parite	Worm	0.28%
Small	Trojan	0.26%
Heuristic/generic	Misc	0.21%
Sality	Virus	0.19%
Stration/Warezov	Worm	0.14%
Murlo	Trojan	0.14%
Klez	Worm	0.09%
Inject	Trojan	0.09%
Mywife/Nyxem	Worm	0.07%
VB	Worm	0.07%
Others[1]		0.48%
Total		100%

[1]Readers are reminded that a complete listing is posted at http://www.virusbtn.com/Prevalence/.

MALWARE ANALYSIS

WHITHER THE HARUMF?

Peter Ferrie
Microsoft, USA

The second in our series of analyses of viruses contained in the EOF-rRlf-DoomRiderz virus zine is that of W32/Harumf.

'A' 'S'ILLY 'L'ITTLE 'R'EPEAT

The virus begins by decrypting the first stage of its body and attempts to transfer control to it using an address that it calculates from values in the PE header at the time of infection. This means that the virus is not aware of Address Space Layout Randomization (ASLR). If the infected file has been built to be ASLR-aware, then the virus will crash and the application will terminate. This is not a good way to start.

The decryptor is oligomorphic, having only very few variations, which are taken from a fixed set.

UN-SafeSEH

The first stage of the virus registers a structured exception handler, then intentionally causes an exception. This is an old anti-debugging trick which any good debugger can skip easily enough. Since the handler appears immediately after the call to the anti-debugging routine, it's a simple matter to step over the call and continue execution. However, the virus is not aware of 'SafeSEH', which overrides the legacy structured exception handling. If the infected file was built with SafeSEH, then the exception that the virus raises will cause the application to exit because the exception address will not match any known address.

One could be forgiven for thinking that we are looking once again at W32/Divino [1], since the two viruses were written by the same person, and they clearly share some code (and many of the same bugs).

The virus unregisters the handler and then decrypts its second stage, but the decryptor works only if the virus code is of even length. The reason for this is a combination of instructions and parameters that are not supposed to go together. There is a subtraction, then a comparison, and then a branch. The problem is in several parts. The subtraction is by two, and the branch is taken only if the carry and zero flags are both clear. This would work regardless of the size of the code, if it were not for the comparison. The comparison is made with zero, for which the carry flag can never be set, and the zero flag is set only if the result is zero. The zero flag cannot be set if the virus length is not even. The proper branch instruction would be one that checks the sign flag instead of the carry flag.

This problem is not present in W32/Divino because in that case, the decryptor uses an addition and a comparison of a value that is larger than zero, so the following branch works as intended, regardless of the size of the code.

I'M A LOCAL

The virus stores the selector of the local descriptor table onto the stack, and then reads four bytes and checks if the result is non-zero. A non-zero result should always occur because the location on the stack holds the previous stack frame when the process started, which is always an address above the 64 KB boundary. As a result, the top half of the stack frame will remain untouched and non-zero. This might be an anti-emulator trick for an emulator that stores four bytes instead of two. However, it seems more likely that what the virus author had in mind was to read only two bytes and detect whether the local descriptor table (LDT) is in use, but had to reverse the condition because of the extra bytes that the virus reads. The use of the LDT is a characteristic of virtual machines such as *VMware* and *VirtualPC*, along with *Norman*'s *SandBox*.

Next comes a specific detection for *Norman*'s *SandBox*, using a variation of a finding that was described in [2]. In this case, the attack is that *Norman*'s *SandBox* returns the same information for the CPUID instruction, regardless of the index that is specified.

BYTE, BYTE BABY

The virus retrieves an address from the stack that points within the kernel32 BaseThreadInitThunk() function. Using this as a starting point, the virus performs a brute-force search in memory for the 'MZ' header. The search is performed byte by byte, rather than on 64 KB boundaries, making it slow and inefficient. The virus does not register a structured exception handler for this operation. As a result, the technique fails on *Windows Vista64*. This is because the kernel32.dll in *Windows Vista64* uses a 64 KB section alignment, so the region between the file header and the first section are not mapped. Any attempt to access this memory will cause an exception which is not intercepted by the virus. If an exception occurs, the virus will crash and the application will terminate.

NEW YEAR'S RESOLUTION

The virus resolves a set of API addresses from kernel32.dll that are required to infect files, using the standard GetProcAddress() method. As a result, the names are clearly visible in the code. Despite this, however, three of the resolved functions are not used.

The virus also resolves a set of API addresses from advapi32.dll that are required to replace a registry value. The RegQueryValueExA() function address is also resolved but not used, because the virus does not care about the previous content of the data that it will replace.

At this point, the virus copies back the bytes replaced by the first decryptor and then begins the search for files to infect.

INFECTIOUS GROOVES

The virus allocates some memory to hold the name of the current directory. There is a bug in this code, however, which is that the memory is never freed. The virus enumerates all objects in the current directory, and looks for anything whose name ends with '.exe'. The virus assumes that such an object is a file. This is a minor bug, but it has no effect here. The reason it has no effect is because the virus attempts to load the object into memory. A directory will cause an error to be returned, which the virus intercepts. This also happens to filter out 64-bit files. How fortunate for the virus author.

CHECKS AND BALANCES

The virus attempts to find a particular resource within the file, whose presence is the infection marker. The resource is a data type with an identifier of 1234. If the resource is not found, then the virus checks within the file for the 'MZ' and 'PE' signatures and the presence of a resource data directory. Another bug exists here, which is that the 'PE' signature comparison is incomplete. The true signature is four bytes long, but the virus checks for only the first two bytes. Of course, the initial load of the file would fail if the file is not in Portable Executable format, so the check for the signatures is redundant.

The virus attempts to copy 512 bytes of data from the host entrypoint, but without checking if there are at least that many bytes available to copy. If the entrypoint is located less than 512 bytes from the end of the image, then the virus will crash and the application will terminate.

The virus walks the section table once to check for pure virtual sections. If any are found, then the virus will skip the file. The virus walks the section table again to check for a section whose name begins with '.rsr'. This is intended to find the '.rsrc' section, but because the full name is not checked, there could be other sections that are matched instead. This could cause trouble later. The virus also requires that this section is the last one in the image.

NOT VERY RESOURCEFUL

If all goes well, the virus unloads the file and then allocates some memory to hold a copy of the virus body. There is a bug in this code, which is that the memory is never freed. The virus encrypts the second stage at this point. Now we reach the 'feature' of the virus. The virus attempts to inject itself as a resource. It uses the resource-updating APIs to do that. However, there is a problem. In order to update a resource, one must specify its language. The virus uses a generic English language selection, which restricts the scope of infection. As a result, any file with multi-language user interface (MUI) resources (the default for many files in *Windows Vista*) will not be infected because the exact language (primary and sub-language) must match.

If the resource updating succeeds, then the virus opens the infected file and requests the file size. The virus allocates some memory to hold a copy of the entire file. There is also a bug in this code, which is that the memory is never freed. The virus walks the section table to find the section that contains the entrypoint, and walks the section table yet again to find the section whose name begins with '.rsr' (even though we know that it's the last section – no-one said that this code is optimal). There is yet another bug here. If the entrypoint is not in any section, then the virus will search beyond the end of the table. It will probably find something that covers the entrypoint value, but the results will be unpredictable.

REALLY 'NO EXECUTE'

The virus replaces completely the characteristics for the entrypoint section. It changes them to read/write/init, and does the same for the resource section. This act is not compatible with DEP, since without the Executable flag set in the section header, the contents of the sections cannot be executed on platforms that support DEP.

The virus searches the entire resource section to find the copy of itself. This is certainly simpler than parsing the resource data, but the virus searches only for the first four bytes of its code, which can easily match graphical data and other things. If the virus 'finds' itself, it encrypts the first stage. If the match was in fact false, then the results could be messy.

Finally, the virus copies the decryptor to the host entrypoint and writes the updated data to the file. Another bug exists here, which is that the file handle is never closed. The result is that one handle is leaked for each infected file.

At this point, the virus searches for another object and repeats the process until nothing more can be found.

BUT WAIT, THERE'S MORE

The local replication part of the virus ends here. Then begins the remote replication part. The virus begins by retrieving the process path name, and searches within the last eight

characters for the 'haru' string. The significance of the 'haru' string will be described below. Meanwhile, there are two bugs in this code. The first is that the comparison is case-sensitive. This bug is minor, since the virus is likely to have been the creator of the file. The second bug is that the virus does not verify the entire name. This has an effect later.

If the 'haru' string is found, then the virus wants to run 'explorer.exe' with the drive letter of the drive that contains the file. However, there is a bug in this code. The virus constructs the string on the stack, but does so below the current stack pointer, instead of allocating stack space. Then it calls the GlobalAlloc() function to allocate some memory to hold a copy of the string. The problem is that the API call destroys the string. In any case, the allocated length is also off by one byte, which causes heap corruption when the 'string' is copied there. There is also another bug, which is that the memory is never freed.

HARU ICHIBAN!

If somehow the string survived, then a directory listing is displayed for the specified drive. At this point, the virus checks if the payload should run, and also runs two other replication methods, before exiting silently. It is here that the effect of the incomplete 'haru' check appears. The problem is that if an infected file contained the 'haru' string in the name, then the host code will not be executed any more. Given that 'Haru' can be a person's name in Japanese (who remembers the author of LHARC?), and it also means 'spring' (the season), there is certainly the possibility of encountering files that contain the string. It may be a rare bug, but it is still a bug.

If the 'haru' string is not found, then the virus performs an additional check before checking if the payload should run, and running two other replication methods. For some reason, one of the methods is executed twice in both cases. Perhaps another method was intended to be included.

The additional check that the virus performs is whether the user is a member of the Administrators group. It uses the IsUserAnAdmin() function, which is documented by *Microsoft* as available in *Windows 2000* and later, but it appears to be present only in *Windows XP* and later. The function is a nice wrapper around code that checks the token membership. If the user is not a member of the Administrators group, then the virus displays the message 'You need Administrator Privilege to run this Application', and then exits. Otherwise, the virus attempts to retrieve the address of the InitializeSRWLock() function. This function was introduced in *Windows Vista*, and its presence or absence provides a method of determining the platform without the use of the GetVersionExA() function (whose results are currently being faked by some anti-malware emulators).

If the virus is running on *Windows Vista*, then the virus sets to zero the 'EnableLUA' value in the 'HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System' key, which will disable the UAC. However, this has no immediate effect as a reboot of the system is required for the change to be applied.

REMOTE CONTROL

The payload activates on the 9th of each month. It attempts to download a picture and place it in the root directory of the C: drive. On *Windows Vista*, standard users cannot write to that location, so the download fails in that case. If the download is successful, then the virus waits three seconds before displaying the picture. The picture is a banner that says 'Saddam's Family'. Rather than being a family photo, it's the logo of a heavy metal band which goes by that name.

The first additional replication method is that the virus copies itself as 'vista_crack.exe' to some P2P shared folders, assuming that they exist. The relevant P2P applications are *KaZaA Lite*, *KaZaA*, *EDonkey2000*, *ICQ*, *eMule*, *Gnucleus*, *KMS*, and *LimeWire*.

The second additional replication method begins by getting the bitmap of currently connected drives. For each drive, the virus allocates memory to hold the name of the current directory. There is a bug in this code, which is that the memory is never freed. The virus changes to the root directory of the drive, and then searches for files to infect. The virus also copies itself as 'harulf.exe' to the root directory of the drive, and drops an 'autorun.inf' file that contains a reference to the 'harulf.exe' file. This is the reason for the 'haru' check above. For removable media, the 'autorun.inf' file will run the 'harulf.exe' file when the drive is connected. Since the copied file is also an infected file, the virus does not want the host code to run at that point. The virus checks all 32 bits of the map, even though there can be only 26 drive letters. This might also be considered a bug.

CONCLUSION

Perhaps the funniest thing in this virus, even more than the numerous bugs, is the virus author misspelling his own name: 'coded by fakedmnded!'. Oops, 'i' did it again.

REFERENCES

[1] Ferrie, P. Prophet and Loss. Virus Bulletin, September 2008, p.4. http://www.virusbtn.com/vba/2008/09/vb200809-prophet-loss.

[2] Ferrie, P. Attacks on more virtual machines. http://pferrie.tripod.com/papers/attacks2.pdf.

FEATURE

THE HIDDEN COST OF COMPROMISE

Mary Landesman
ScanSafe, USA

Web-based malware is not a new phenomenon; the Internet as a whole has historically been the single largest factor enabling the global spread of malware and the web has increasingly proven a particularly successful vector. What is new is the overwhelming number of compromises of known, legitimate websites as a means to distribute web-hosted malware. Obviously these website compromises pose serious risks to the site owners and their visitors, but what of the less obvious threat they pose?

RISKY BUSINESS

Widespread compromises of known, legitimate websites were first reported in October 2007. While initial reports were largely confined to webmaster forums and newsgroups, by January 2008 the compromises had become numerous enough to attract international attention. By early spring 2008, the attacks had reached epidemic proportions – impacting hundreds of millions of pages across the web.

In a comparative study of May 2007 and May 2008, *ScanSafe STAT* looked at the increase in risk exposure due to web-based malware (focusing on customers that were common to both periods to preclude any bias related to changes in customer base). It was discovered that the risk of exposure to web-based malware had increased 220% in May 2008 compared to May 2007. Demonstrating the exponential growth of these attacks, *ScanSafe STAT* repeated the study, calculating the risk exposure the same customers faced in July 2008 compared to the previous May-to-May comparison. In July 2008, the risk of exposure to web-based malware had increased by 443% compared to May 2008 and by 1636% compared to May 2007.

A MALICIOUS MÉNAGE À TROIS

Web-based malware distributors face the same challenge as any other web property owner – how to drive traffic to their domain. As a result, today's attackers have less in common with traditional malware authors/distributors than with marketing pros specialized in search engine optimization (SEO).

By providing a means to monetize websites, advertising plays a key role in the economic viability and continued expansion of the web. Virtual landlords are effectively able to charge rent for their space, subleasing the attention spans of their hard-earned visitors in exchange for revenues from the advertising giants.

To embed third-party advertising (or other third-party content), web pages are coded with hidden iframes and external JavaScript references – components as crucial to the web's connectivity, growth and ongoing success as hyperlinks themselves. To deliver the ads, these external references load content from designated ad servers, enabling advertisers to reach multiple sites simultaneously.

To shortcut their SEO efforts, attackers initially leveraged the popularity of existing websites by inserting malicious advertising somewhere within a participating ad network. Rogue advertisers and poorly policed affiliate networks caught even legitimate advertisers off guard.

From 2003 through to 2005, malicious pop-ups delivering adware and spyware proliferated, leading to 'drive by download' and 'browser hijacking' becoming common household terms. Public outcry and consequent legal sanctions and improved technologies helped to stem the flow, though rogue advertising does still occur. During 2007, *ScanSafe* blocked malicious advertising foisted through the Miami Dolphins stadium website, *TomsHardware.com*, *Photobucket*, *MySpace* and hundreds of other sites. Attackers even targeted advertising on parked domains hosted by *NameDrive*, which, thanks to pre-existing links on non-parked sites, enjoyed considerable traffic despite no longer being active.

Today's attackers have taken an even greater shortcut, cutting out the middleman altogether. Rather than inserting a malicious ad in the advertising network, these attackers have gone straight to the source code of the target website. This direct form of compromise began largely as a manual effort. While successful, it was time-consuming for the attackers. To solve this problem, the attackers did what any development firm might do – they introduced automation. In an evolutionary sense, it was the introduction of automated tools and their subsequent availability that enabled website compromises to be rendered repeatedly en masse on a global scale.

The most predominant of these compromises have been those rendered through automated SQL injection attacks, the majority of which are currently carried out via the Asprox botnet. But while the SQL injection attacks have

understandably been the headline grabbers, all forms of website compromise have been on the increase.

THE BUTTERFLY EFFECT

The 'butterfly effect' is a term whose origins lie in chaos theory and which is often used to refer to the way in which even the smallest of events (such as the flap of a butterfly's wings) can set in motion a series of events that have far-reaching and often unexpected consequences – or at least consequences that appear far removed from the original action.

While the immediate risks posed by infection via website compromises are well established, thanks to the butterfly effect there are far-reaching consequences which aren't nearly as obvious. Chief among these are changes in user habits and the subsequent impact those changes may have on Internet advertising revenues in the long term.

According to the 2007 IAB Internet Advertising Revenue Report (published in May 2008) [1], Internet advertising revenues outpaced cable television, radio, broadcast television, as well as consumer and trade magazines in 2007, reaching $21.2 billion. At 41% of the total, search revenue was reportedly the single largest contributor. A *Nielsen Online* study [2] reports that in January 2006 there were 64.3 billion sponsored link advertising impressions on *Google* and *Yahoo* (including their extended advertising networks).

With 64.3 billion sponsored link advertising impressions and 41% of the Internet advertising revenues at stake, even the most subtle of ripples can have an impact.

BROWSER ADOPTION

It is virtually impossible to gauge web browser usage stats reliably. Sources that claim to do so are in fact merely reporting on the user agents presented by the browsers used by their own site visitors. There are many limitations to this approach. First, user agents can lie and the site itself may exert its own influence by optimizing the code in favour of one browser over another. Depending on the topic, a particular site might attract a particular demographic – an audience that is not necessarily reflective of the web as a whole.

Different browsers also access pages in different ways, which can skew browser usage statistics. For example, web pages consisting of multiple elements may, depending on the browser, be reloaded multiple times, thus resulting in an over count of page visits. Statistics-gathering challenges are also introduced by the use of proxies, shared IP addresses, and a host of other factors related to origin and relay that may artificially increase one browser's popularity over another.

Despite limitations in browser usage stats, there are still interesting trends to be found within the captured data. One example of useful browser usage stats is reported by w3schools.com which provides tutorials for website developers. The w3schools stats are taken from the site's own logs and thus are reflective only of a specific user base – i.e. the visitors to the w3schools site, who can be expected predominantly to be website developers and thus to be more technically savvy than the average web user.

According to the w3schools data, the use of *Firefox* increased 20% between January 2008 and August 2008. The largest growth occurred between February and May 2008 and thus cannot be contributed to the concerted marketing efforts surrounding the release of *Firefox v3.0* in June 2008. If the more technical, web-savvy visitors to w3schools are adopting *Firefox* over *Internet Explorer*, is it possible that this is due at least in part to heightened awareness of risk exposure brought upon by the mass website compromises?

NO SCRIPTS, PLEASE

It is not simply the browser but rather how it is configured that impacts on web usage. Ask most computer security gurus what steps home users can take to protect themselves while surfing online, and most will likely recommend using *Firefox* with the *NoScript* add-on. According to the *Mozilla* add-ons site, the *NoScript* add-on for *Firefox* has reached 27.3 million downloads, at an estimated 378 thousand downloads per week. There are multiple distribution points for this add-on, thus the numbers are not reflective of total downloads but are high enough to suggest widespread adoption.

As its name suggests, the *NoScript* add-on blocks JavaScript and other active content by default, enabling users to allow or deny scripts on a per-site basis. While users can elect to allow scripts globally, that option is labelled as dangerous (rightfully so) and thus it is unlikely to be selected. (Besides which, doing so would defeat the purpose of the *NoScript* add-on.) Other more viable choices include: 'Temporarily allow all on this page', 'Temporarily allow <visited site domain>', and 'Temporarily allow <third party domain>'.

Given these choices, how many users will elect to globally allow all scripts, or allow scripts from third-party providers (particularly advertising-related ones)? It doesn't seem far fetched to assume that the use of *NoScript* encourages the allowance of JavaScript only from the visited site while not allowing scripts from third parties which provide other content (including advertising) to the site.

As an example, in a DSLReports thread [3] regarding the *Firefox NoScript* add-on, the responses overwhelmingly favoured blocking all third-party content. As one poster commented '… I still block any third-party stuff. If I trust a site, I trust that site and not all third-party sites whose content the (sic) might use.'

NoScript is not the only add-on that *Firefox* users are adding to their protective arsenal. The *Mozilla* add-on site lists the ten most popular add-ons for *Firefox*, three of which are: *AdBlock Plus*, *NoScript* and *CustomizeGoogle* – all of which have either the express purpose of blocking advertising or include features that will, as a side effect, block advertising. Collectively, these add-ons have been downloaded by 60.8 million users.

The recently debuted *Internet Explorer 8* (beta) also gets in on the act, providing a feature dubbed 'InPrivate Blocking', which has a similar effect. According to *Microsoft*, 'Users are often not aware that some content, images, ads and analytics are being provided from third-party websites or that these websites have the ability potentially to track their behaviour across multiple websites.' If the 'InPrivate Blocking' feature is enabled, *Internet Explorer* will automatically block third-party content that it has observed across multiple sites.

InPrivate Blocking also 'helps prevent your browsing history, temporary Internet files, form data, cookies, and usernames and passwords from being retained by the browser, leaving no evidence of your browsing or search history.' While stopping short of blocking all JavaScript outright, by design, InPrivate Blocking would block the most widely deployed third-party advertising as well as third-party website analytics, examples of which include both *Google AdSense* and *Google Analytics*.

Google Chrome's 'Incognito' feature [4] is similar in spirit to *Microsoft*'s InPrivate Blocking feature – with one notable exception: omitted from the *Chrome* browser is the ability to block third-party content. Additionally, while *Chrome* uses virtual machine technology to sandbox JavaScript and other active content, it does not provide a means to disable it altogether, casting some doubt on its ability to fend off web-based malware attacks. Within days of *Chrome*'s release, researchers discovered two buffer overflow conditions which could enable the remote execution of arbitrary code and an out-of-bounds memory read error that left the browser in an unstable state.

THE TWO FACES OF JAVASCRIPT

Not all users will lament the inability to block JavaScript, regardless of the security implications. A browser that doesn't supply script blocking will likely be welcomed by members of traffic exchange networks, which rely on members to click through to other member websites to inflate page views. Administrators of some of those networks have gone so far as to forbid *Firefox* use among members. Their reasoning is concern that users will elect not to allow third-party advertising and thus click their way into higher referral benefits without contributing to ad impressions for the other members. But while fringe users may eschew ad blocking, with 60 million downloads and counting, it is hard to discount the notion that many normal web surfers are in favour of it.

Certainly it's too early to tell what sort of lasting impact the ongoing website compromises will have on browser adoption but it does seem likely that for many users security will be a deciding factor. It is impossible to say whether the increase of web-based malware (and subsequent increases of script-blocking technologies) played any role in *Google*'s decision two years ago to begin crafting a new browser. But it is certain that disabling JavaScript protects against the ill-tended effects of website compromise and equally certain that the $21.2 billion in Internet advertising revenues are largely dependent on the continued use of JavaScript.

CONCLUSION

Clearly, legitimate Internet advertising plays a critical role in the ongoing health and viability of the web, and has a significant impact on the global economy as a whole. Technologies and services that protect both the user *and* the advertiser should be viewed as imperative. And if Internet advertising revenues do take a downturn, ask yourself – is it due to recessionary conditions, or is it because the web is under attack?

REFERENCES

[1] IAB Internet Advertising Revenue Report. http://www.iab.net/media/file/IAB_PwC_2007_full_year.pdf.

[2] Sponsored Link Advertising on Google And Yahoo! Grows 16 Percent in Six Months: Yahoo's Sponsored Links Rise 21 Percent, Google's 14 Percent, According to Nielsen//NetRatings. http://www.nielsen-netratings.com/pr/pr_060216.pdf.

[3] DSL Reports Tech and Talk forum. http://www.dslreports.com/forum/r19479692-FireFox-NoScript.

[4] Google Chrome Releases blog. http://googlechromereleases.blogspot.com/.

OPINION

BROADLY SPEAKING: SKILL DIVERSIFICATION IN THE AV COMMUNITY

Hannah Mariner
HCL/CA, Australia

If someone asked you to describe your first day on the job, in 25 words or less what would you say? My response would be this: I was as green as Kermit. After weeks in Malware 101, the concepts were still mysterious and hazy. I pretended a lot.

A VERY SMALL CASE STUDY

Having never imagined a career in the AV industry, it happened in one of those serendipitous twists of life. A three-month maternity leave vacancy opened at *CA*, the software company at which I worked, and I stepped up to fill the temporary role of technical editor.

I researched. I read numerous articles and dictionary definitions and security blogs and websites to try to familiarize myself with the material I was about to work with. Yet, I was desperately unprepared. On my first day, I might have been able to tell you what a virus was. *Might* have. As for 'polymorphism', 'Browser Helper Object', 'rootkit' – well, you're kidding me. I hoped that my new team members were unaware of my feelings of fear at being lost, but as far as I was concerned, I was illiterate. If this was the alphabet, I was starting at 'A'. I had no real knowledge or understanding either of malware itself, or of the industry built around it.

Hold on to your assumptions, though. My placement in the role was not a misplacement, as you might first think. I did have things of value to bring to the team. I had a broad knowledge-base of communication styles and strategies; I could talk to people, one-on-one; I dealt with spelling misdemeanours as smoothly and naturally as James Bond delivers a self-introduction; I liked to sit with a clunky paragraph for an hour just playing, as though with pieces of a jigsaw puzzle, until the words suddenly and gloriously began to belong to one another. Those skills, utterly unrelated to malware, were enough to keep things going day to day; and six months later when things relating to malware and the anti-malware industry had sunk in a good deal more, those skills allowed me to bring a new and fresh perspective to my co-workers and, I hope, the product we represented.

A QUICK LOOK AT INDUSTRY TRAITS

Aside from any personal meaning my story might have to me or even any humanist meaning it might have to you, I think that parts of my anecdote are important in a larger, communal context, in the sense of industry diversification, innovation and survival: the recognition that generalized skills can be put to wonderful innovative use, especially in a niche, highly specialized industry. While it is becoming more common to meet people successfully contributing to the industry without specific security qualifications, it is true to say that the anti-virus field is difficult to enter unless you already have an IT background. And while it is a technical field requiring technically proficient minds, it's also, from my experience, a field that could reap sound rewards from looking for potential in applicants from non-IT backgrounds.

Recognizing this as an industry that tends to be closed, it is worth looking at some of the peculiarities that encourage this atmosphere. A caveat straight up: this is a broad-stroke piece based on a broad-stroke idea about widening the reaches of the industry in which I work, so some of these points will sound, well, broad.

- *The AV industry has a unique market position*. From the beginning, the anti-virus industry has occupied a very specific, defined and distinct pocket of the software marketplace. It has traditionally sought workers either with these specific skills, or with as close a skill-set match as possible.
- *The AV industry is mature.* When researching the historical annuls and putting numbers and years to things, this doesn't seem like an old community. Some of the veteran anti-virus companies like *Sophos* and *McAfee* are 20 years old or slightly over. However, this is a mature, well-established field with solid social, financial and professional structures, and heavily reinforced processes and practices.
- *The AV industry has active stakeholders.* Related to the point above, the industry has founders of sorts – pioneering researchers and managers who were there in the industry's early years and who remain actively involved and invested in the business now. You could say that the industry has a fair bit of 'living history'.
- *The AV industry values cooperative interpersonal networks and relationships.* This is an industry based on prized and tightly woven professional networks. By its very nature, the anti-virus industry has had to be positioned, particularly in the past, to summon immediate, coordinated, global responses to alarm bells; for example, in times of virus outbreaks. In the

past it has depended on interpersonal networks robust and secure enough to quickly communicate information and respond with solutions – and it still does.

- *The AV industry has an inner circle.* Being a community of people committed to serving a protective function, there has been at least the perception of there being an elite, inner circle of knowledge-holders, in stark contrast of course to another group, the knowledge-deprived.

THE PERCEIVED INNER CIRCLE

The last point is the one I'd like to pick up on a little more. It seems to stem from the conceptual themes behind the mission statements of most software security companies – that of protecting and defending those who don't know how, the defenceless and unaware; that of participating in a classic 'goodies vs baddies' set-up and of the importance of trade secrets in keeping ahead in the battle. As in all industries, but especially in this one, the difference between the knowers and the don't-knowers is crucial, and has been noted before. As pointed out by Peter Svensson on *Security Focus*, 'Ludwig, who went on to write *The Big Black Book of Computer Viruses* and similar collections, believes the anti-virus industry thrives on secrecy and mystique and is loath to spread knowledge.' [1]

The proposition of the knowers and the don't-knowers was also dealt with in a 2005 essay by Jessica Johnston, who explains the purpose of CARO, a 'very elite group of AV computer researchers created by the researchers themselves out of the necessity to share specialized, restricted and what they consider to be dangerous information' [2].

Having introduced CARO, Johnston presents us with the juxtaposition of CARO and REVS, a now defunct group that was 'started by a groundswell of frustration fuelled by the lack of information distributed by CARO when an actual global virus broke. REVS was an organization of AV vendors who shared information about viruses and virus outbreaks with each other.' Eventually REVS disbanded, prompted, Johnston claims, by the fact that 'people and organizations could not afford, literally and symbolically, to be out of the CARO information stream. The need to disseminate urgent and vital information about a global virus outbreak was repositioned by CARO as a dangerous attempt to spread secret information to untrusted and potentially unethical "anybodies".'

While it's difficult to ascertain how much of either of the former or the latter scenarios are true in practice, it is certain that the perception of the knowers and the don't-knowers is real. Arguments about the reality of the situation aside, the perception is in itself an area for examination, reflection and potentially, change.

WHY DIVERSIFY?

There is a notion in business theory which says that 'any innovation is founded on novel knowledge or a novel recombination of existing bits of knowledge' [3]. This idea of innovation through novel reapplication, with a specific focus on the anti-virus industry, was also touched on in a 1996 article by Sung Moo Yang, who makes a lengthy case for the idea that 'innovation of AV technology could come from existing theories and technologies that are applied to AV' [4].

Though on a small scale, my personal experience supports the concept that investing a little extra time into the development of talent from a non-IT background can actually bear fruit and be considered an investment in the literal sense, 'the commitment of something other than money (time, energy, or effort) ... with the expectation of some worthwhile result' [5]. I'm convinced that allowing people from diverse professional backgrounds into what can be an industry of knowers and don't-knowers, is one way forward and is one viable way – among others – to strengthen, prolong and add direction, vision and life to the industry as a whole.

REFERENCES

[1] Svensson, P. Antivirus industry steamed over virus article, college class 2003. http://www.securityfocus.com/news/5698.

[2] Johnston, J. Communications with Global Space: Negotiations of local/global tensions within the computer antivirus industry, p.6, 2005. http://www.mang.canterbury.ac.nz/anzca/FullPapers/12CultureCommFINAL.pdf.

[3] Ferreira, M. P.; Serra, F. A. R. Open and closed industry clusters: The social structure of innovation, p.11, 2008. http://www.globadvantage.ipleiria.pt/wp-content/uploads/2008/06/working_paper-24_globadvantage.pdf.

[4] Yang, S. M. Productivity, Technology and AntiVirus Industry. 1996. http://web.archive.org/web/19990428133500/http://www.intergate.bc.ca/personal/yang/avindust.html.

[5] Retrieved on 20 September 2008. http://www.thefreedictionary.com/investment.

COMPARATIVE REVIEW

WINDOWS SERVER 2008

John Hawes

The comparative review moves to an entirely new platform this month: the server version of *Microsoft*'s latest iteration of *Windows*. With the official release of the platform having been in February, there should have been plenty of time for developers and QA teams to ensure their products were fully integrated with the new environment.

This month's testing schedule saw a number of new challenges in addition to the usual time pressures and resource limitations. The breaking in of a new member of the testing team coincided happily with a series of significant adjustments to the standard line-up of testing tasks, more on which shortly. The range of products taking part continued to reflect the steady increase in diversity in the market. As always, the team entered the test lab hoping for smooth and speedy testing, but anticipating the gamut of problems including bizarre design, bewilderingly absent functionality and disappointing instability.

PLATFORM AND TEST SETS

The *Server 2008* platform shares a code base with *Vista*, with many tweaks and improvements in a variety of areas, but sensibly avoiding the rather showy and resource-hungry cosmetic adjustments which most users will identify with the new breed of *Windows* systems. The installation process follows the usual series of steps. Following the standard *VB* methodology, things were kept as simple as possible, with simple fileserver functionality added from the list of models available. Some driver software was required to activate networking and to get the most out of the graphical capabilities of the hardware in use, and some archiving tools were also installed to simplify the unpacking of the submissions, which as ever took on a wide range of formats. Unlike in the *Vista* desktop tests, no adjustments were made to the user set-up, and a user with administrative rights was logged in for all testing purposes, assuming that server administrators would need such rights to install core software to a system. With these tasks carried out, and a few tweaks to the display and desktop made for comfort and efficiency, images were taken of the identical systems and the test sample sets copied to the secondary hard drives ready for testing to begin.

As mentioned, the test sets saw some considerable evolution this month. Starting with the core of the VB100 sets, the WildList set was aligned with the July issue of the WildList, released about a week before the product submission deadline (2 September). The changes in the list from that used in the previous test included the disappearance of large numbers of older items, only to be replaced by an impressive swathe of new arrivals, the vast majority of which were trojans that target online gamers and most of these go by the fairly straightforward title of 'W32/OnlineGames'. A few of the more interesting items on the list were removed, including several of the W32/Virut variants, but enough of these highly polymorphic viruses remained to provide a frisson of danger for those products which had previously had difficulties providing full coverage of these items.

In the clean test set, a fairly large update was made with a swathe of software added. This included a selection of drivers and system tools acquired as part of the process of enabling the test systems and the new platform to interact, as well as a collection of packages downloaded as freeware or trial installations, this month focusing on web development tools. These enlargements of the test set were designed in part to expand the speed test collections, which are now approaching an acceptable size. The additions to the set were selected from software with reasonably significant manufacturers with reliable reputations, so were not expected to bring up a large number of false positives, but as ever with the growth of the set the chances of a mislabelling grew, and the older part of the set still seems to throw up occasional incidents.

The combination of these changes to the test sets with the new platform seemed to provide a pretty tough challenge for those vendors striving for the glory of a VB100 award, but we also paid attention to the additional information provided for our readers. The zoo collections saw another round of development towards a more flexible and relevant set of challenges, with the dwindling and less difficult test set of simple file-infecting viruses being retired to the legacy set for the time being. Replacing these was a substantial new selection of trojans, replacing entirely the set used in the last review with fresh samples gathered in the last two months. The set of worms and bots saw a small amount of updating, but we hope to implement a similar system of complete overhaul for each review in the near future.

Another upgrade was trialled this month, which is intended to add even fresher samples for each test, along with an element of retrospective testing to measure heuristic and generic detection capabilities. Preparations for this scheme – preliminary results of which we hope to present at the forthcoming *VB* conference in Ottawa – involved putting together a month's worth of new arrivals totalling well over 100,000 samples. The logistics of this looked set to be dwarfed by the difficulties involved in persuading a bevy of awkward and intractable products to produce usable results when scanning such a large set of samples in the very limited time available. Without further ado, we shut ourselves in the lab and got down to business.

Agnitum Outpost Security Suite Pro 6.5.2358.316.0607

ItW	100.00%	**Polymorphic**	75.64%
ItW (o/a)	100.00%	**Trojans**	75.67%
Worms & bots	99.93%	**False positives**	0

With this month's review running on a server platform, we expected most of the products to be dedicated to a server environment, but since many products designed for the desktop run quite happily in the same setting we accepted any such products which vendors saw fit to submit. First up on the roster alphabetically, *Agnitum* provided the same product as that entered successfully in several recent comparatives. Combining the company's own highly regarded firewall technology with a range of security extras including anti-malware detection provided by the *VirusBuster* engine, the product once again put in a solid performance, with a slick and well-designed interface and smooth, stable running.

Detection rates were reasonable, with somewhat below par coverage of the set of recent trojans but no problems in the WildList set. In the clean sets scanning times were fairly good, and an absence of false positives grants *Agnitum* its second VB100 in a row.

AhnLab V3Net 7.0.0.2

ItW	100.00%	**Polymorphic**	79.40%
ItW (o/a)	100.00%	**Trojans**	72.30%
Worms & bots	99.84%	**False positives**	0

AhnLab's *V3Net* product had some difficulties in the last comparative (see *VB*, August 2008, p.13), with the introduction of some engine upgrades causing some crashes during on-access scanning. The product provided for this test seemed pretty similar on the surface, with a simple and fairly attractive interface which kept some of its most useful controls hidden far away from where they might be expected to be found.

Some initial scanning results were safely obtained once the layout of the interface had been deciphered, but during on-access scanning of the trojan set blue screens were encountered, and repeated attempts to prevent this by the judicious removal of what were presumed to be offending samples proved fruitless. To get usable detection figures the set was eventually excluded from scanning entirely. By a chance mistake it was discovered that the list of executable file types did not include the .cmd extension used by some worms, which led to some worries until we found that the default setting was to scan all files regardless of type. The WildList was covered in full in both modes without further incident, and with speeds across the clean sets really quite good and false positives notably absent, *V3Net* makes the grade for a VB100 despite the wobbles.

Alwil avast! 4.8 Server Edition 4.8.985

ItW	100.00%	**Polymorphic**	92.25%
ItW (o/a)	100.00%	**Trojans**	94.20%
Worms & bots	99.78%	**False positives**	0

Bucking the trend seen so far, *Alwil* provided a server-specific product for this test. The interface showed little difference from that seen in recent desktop tests, other than by the fact that the rather funky pared-down interface provided by default in the desktop version was absent. However, this made little difference to testing, which generally requires the advanced options provided by the grown-up interface.

Detection rates across the sets were highly impressive as ever, and speeds were pretty good on demand, and reasonable on access. No problems were encountered covering the WildList, and without any false positives *Alwil* wins another VB100 award.

Arcabit ArcaVir 2008

ItW	90.58%	**Polymorphic**	86.54%
ItW (o/a)	90.58%	**Trojans**	66.48%
Worms & bots	99.44%	**False positives**	3

Arcabit returns once more to the VB100 test bench, having made its first appearance for several years in the last comparative review (*VB*, August 2008, p.13). The product was unchanged from last time, with the interface impressing with its simplicity and clarity of design. The developer's home market is hinted at by the fact that the option to switch into Polish is available from the system tray menu at all times.

Stability was similarly unimpeachable, even under the heavy strain of scanning large sets of new samples, and detection rates were fairly reasonable across the sets. However, a selection of samples recently added to the WildList were not detected, and in the clean set a small number of items were mislabelled as malware. Hence *Arcabit* does not qualify for a VB100 award this month, but continues to look likely to be a strong contender in the near future.

On-access detection rates	WildList viruses		Worms and bots		Polymorphic viruses		Trojans		Clean sets	
	Missed	%	Missed	%	Missed	%	Missed	%	FP	Suspicious
Agnitum Outpost	0	100.00%	2	99.93%	393	75.64%	1242	75.67%		
AhnLab V3Net	0	100.00%	3	99.84%	703	79.40%	N/A	N/A		
Alwil avast!	0	100.00%	3	99.78%	290	92.25%	447	91.23%		
Arcabit ArcaVir	93	90.58%	8	99.44%	165	86.54%	1799	64.76%	3	
AVG	0	100.00%	1	99.95%	52	90.75%	478	90.63%		
Avira AntiVir	0	100.00%	0	100.00%	0	100.00%	52	98.98%	1	
CA eTrust	1	99.998%	0	100.00%	172	91.82%	3476	31.92%		
ESET NOD32	0	100.00%	0	100.00%	0	100.00%	538	89.46%		
Fortinet FortiClient	0	100.00%	0	100.00%	0	100.00%	5000	2.06%		
Frisk F-PROT	0	100.00%	0	100.00%	125	95.66%	924	81.89%		
F-Secure	0	100.00%	0	100.00%	60	98.03%	466	90.87%	1	
Kaspersky	0	100.00%	0	100.00%	60	98.03%	287	94.38%	1	
Kingsoft	0	100.00%	16	99.10%	2119	41.19%	2605	48.97%		
McAfee VirusScan	0	100.00%	0	100.00%	0	100.00%	216	95.77%		
Microsoft	0	100.00%	0	100.00%	141	95.02%	1054	79.35%		
MWTI eScan Internet Security	0	100.00%	0	100.00%	122	96.00%	205	95.98%	1	
Norman Virus Control	0	100.00%	3	99.78%	1037	70.91%	788	84.56%		
Quick Heal AntiVirus	0	100.00%	45	95.16%	977	79.25%	3477	31.89%		
Redstone Redprotect	1	99.89%	0	100.00%	122	96.15%	481	90.58%	1	
Rising Antivirus	0	100.00%	4	99.64%	1333	60.04%	2260	55.73%		
Sophos Endpoint Security and Control	0	100.00%	0	100.00%	154	92.75%	625	87.76%		12
Symantec Endpoint Protection	0	100.00%	0	100.00%	0	100.00%	395	92.26%		
Trustport Antivirus	0	100.00%	0	100.00%	546	92.06%	155	96.96%		2
VirusBuster for Servers	0	100.00%	2	99.93%	392	75.77%	1281	74.91%		

AVG 8.0.169

ItW	100.00%	**Polymorphic**	90.75%
ItW (o/a)	100.00%	**Trojans**	94.96%
Worms & bots	99.95%	**False positives**	0

AVG also provided the same product for this test as for the recent *Windows XP* comparative: the most recent iteration of the company's suite as reviewed here a few months ago (see *VB*, March 2008, p.18). The new layout is something of an improvement on earlier versions, but remains a little awkward in parts, and getting everything running proved somewhat more fiddly than seemed strictly necessary.

Stability proved no problem throughout the main body of the tests, and although a few issues were observed when scanning the larger sets of infected items, it seems unlikely that such a situation would be very common in the real world. Detection rates were as splendid as ever, and speeds were on the good side of medium. With no false positives and no problems covering the latest WildList, *AVG* earns another VB100 award.

Avira AntiVir Server 8.1.0.1585

ItW	100.00%	**Polymorphic**	100.00%
ItW (o/a)	100.00%	**Trojans**	99.29%
Worms & bots	100.00%	**False positives**	1

Avira's server edition proved very different from the desktop version, with a console approach using the *Microsoft Management Console* as a base. This offered less straightforward access to such things as on-demand scans, as it is intended for sysadmins to set up regular scans of file shares to protect their networks rather than for the simpler

needs of the desktop user. However, configuration options were plentiful and reasonably accessible even for the demanding needs of a VB100 test run.

Detection rates were extremely high – approaching flawless, with the WildList detected effortlessly, and speeds likewise excellent across the board. Unfortunately, a single item in the clean set, which has gone many months without raising any suspicions, was labelled a trojan, and *Avira* thus does not qualify for a VB100 award this month.

CA eTrust ITM 8.1.637.0

ItW	99.998%	**Polymorphic**	91.82%
ItW (o/a)	99.998%	**Trojans**	26.35%
Worms & bots	100.00%	**False positives**	0

CA's *eTrust* product has barely changed in the last few years, with minor version changes little reflected in the product's layout or performance. Again intended more for sysadmins to set up and leave alone, the interface is not ideal for heavy interaction, but provides adequate tuning options for the VB100 test requirements. Implementation of archive scanning seemed not to function properly on access, despite an option to enable it, and logging as usual proved rather ungainly, with access to scan results from the interface itself all but impossible to use. The sluggishness of the interface was amplified by some difficulties scanning larger sets of infected items, which dragged to a halt on several occasions.

These things aside, scanning speeds were as remarkable as ever, and detection rates pretty decent in the more standard sets, if a little disappointing in the new trojans set. False positives were absent, but in the WildList a single sample of one of the W32/Virut variants was not detected, and thus *eTrust* does not make the required grade for a VB100 award this month.

ESET NOD32 Antivirus 3.0.672.0

ItW	100.00%	**Polymorphic**	100.00%
ItW (o/a)	100.00%	**Trojans**	89.00%
Worms & bots	100.00%	**False positives**	0

ESET's highly regarded flagship product was subjected to a major overhaul not long ago, and the stylish new look remains impressive both visually and in usability terms. Tweaking the controls to fit our needs was as usual a delight, and testing zoomed along at its usual rapid pace. Scanning of the extremely large new sets proved a little more sluggish, presumably as the product's strong heuristics kicked in, and on-access behaviour in the new trojan set was also a little odd, with many items not blocked on simple access but treated more severely when copying to the system or even browsing folders in *Explorer*.

Analysis of results showed the product's usual excellent detection rates and yet more splendid scanning speeds over the clean sets, and with nothing missed in the WildList set *ESET* adds yet another VB100 to its record tally.

Fortinet FortiClient 3.0.475

ItW	100.00%	**Polymorphic**	100.00%
ItW (o/a)	100.00%	**Trojans**	2.06%
Worms & bots	100.00%	**False positives**	0

Fortinet's product had a rather slow and lengthy installation process, and brought up one of the few query popups seen in this test, when *Windows* questioned the installation of a driver whose source it could not verify. Once up and running though, the interface presented few issues, being simple and straightforward and providing ample access to a wealth of configuration, as befits the more demanding requirements of a business environment.

Testing thus proceeded apace, with decent speeds and excellent stability even when scanning very large sets. Detection rates were as splendid as ever, but once again bizarrely let down by the trojan set, where detection was almost completely absent, leading to suspicions that some parts of the product were not fully functional. Nevertheless, with no false positives and full coverage of the WildList set, *Fortinet* gains another VB100 award.

Frisk F-PROT 6.0.9.1

ItW	100.00%	**Polymorphic**	95.66%
ItW (o/a)	100.00%	**Trojans**	85.39%
Worms & bots	100.00%	**False positives**	0

The *Frisk* product is simple in the extreme, with a very sparse and plain interface presented after the straightforward setup and obligatory reboot. Minimal configuration options kept work to a minimum, helped by zippy scanning speeds and low overheads, and detection was as usual excellent. A few crashes were observed while scanning large infected sets, including several during on-access scanning, but despite messages claiming the product had ceased to function it continued to block access to malware samples as if nothing had happened.

On-demand detection rates	WildList viruses		Worms and bots		Polymorphic viruses		Trojans		Clean sets	
	Missed	%	Missed	%	Missed	%	Missed	%	FP	Suspicious
Agnitum Outpost	0	100.00%	2	99.93%	393	75.64%	1242	75.67%		
AhnLab V3Net	0	100.00%	3	99.84%	703	79.40%	1414	72.30%		
Alwil avast!	0	100.00%	3	99.78%	290	92.25%	296	94.20%		
Arcabit ArcaVir	93	90.58%	8	99.44%	165	86.54%	1711	66.48%	3	
AVG	0	100.00%	1	99.95%	52	90.75%	257	94.96%		
Avira AntiVir	0	100.00%	0	100.00%	0	100.00%	36	99.29%	1	
CA eTrust	1	99.998%	0	100.00%	172	91.82%	3760	26.35%		
ESET NOD32	0	100.00%	0	100.00%	0	100.00%	561	89.00%		
Fortinet FortiClient	0	100.00%	0	100.00%	0	100.00%	5000	2.06%		
Frisk F-PROT	0	100.00%	0	100.00%	125	95.66%	746	85.39%		
F-Secure	0	100.00%	0	100.00%	60	98.03%	466	90.87%	1	
Kaspersky	0	100.00%	0	100.00%	60	98.03%	193	96.22%	1	
Kingsoft	0	100.00%	16	99.10%	2119	41.19%	2605	48.97%		
McAfee VirusScan	0	100.00%	0	100.00%	0	100.00%	216	95.77%		
Microsoft	0	100.00%	0	100.00%	141	95.02%	1054	79.35%		
MWTI eScan Internet Security	0	100.00%	0	100.00%	122	96.00%	205	95.98%	1	
Norman Virus Control	0	100.00%	0	100.00%	766	78.86%	649	87.29%		
Quick Heal AntiVirus	0	100.00%	45	95.16%	977	79.25%	3450	32.42%	1	
Redstone Redprotect	0	100.00%	0	100.00%	60	98.03%	467	90.85%	1	
Rising Antivirus	0	100.00%	3	99.75%	1333	60.04%	1801	64.72%		
Sophos Endpoint Security and Control	0	100.00%	0	100.00%	154	92.75%	575	88.74%		13
Symantec Endpoint Protection	0	100.00%	0	100.00%	0	100.00%	357	93.01%		
Trustport Antivirus	0	100.00%	0	100.00%	449	92.36%	131	97.43%		2
VirusBuster for Servers	0	100.00%	2	99.93%	392	75.77%	1080	78.84%		

Detection rates were as solid as ever, and with the WildList fully covered and no false positives detected in the clean set, *Frisk* survives a few stability issues to claim another VB100 award.

F-Secure Anti-Virus for Windows Server 8.00 build 123

ItW	100.00%	**Polymorphic**	98.03%
ItW (o/a)	100.00%	**Trojans**	90.87%
Worms & bots	100.00%	**False positives**	1

F-Secure joined the ranks of those providing a special server edition for this test, but after the customary fast and easy installation process nothing seemed very different from the standard desktop product seen in recent tests. The layout of the small window is pleasantly accessible, and allowed all the required tuning to get tests tripping nicely along. Thorough scanning is an available option, and in some cases the default and, with a multiple-engine approach, the speed tests took quite a while to get through. The manufacturer advises that archive scanning on access is best left switched off.

Logging once again left much to be desired, with the HTML log files that were produced regularly appearing curtailed to the point of uselessness, mainly when a large number of infections was found by a single scan. Some careful scan management eventually produced some excellent detection figures, with no problems in the WildList. Unfortunately, however, one of the new additions to the clean test set, a harmless Perl editing tool, was mislabelled as a member of the Hupigon trojan family, thus denying *F-Secure* a VB100 this month and boding ill for the several other products that share core components.

Kaspersky Anti-Virus for Windows Server Enterprise Edition 6.0.2.551

ItW	100.00%	**Polymorphic**	98.03%
ItW (o/a)	100.00%	**Trojans**	96.22%
Worms & bots	100.00%	**False positives**	1

Kaspersky's server version installs its basics as a bare protection system with no controls made available to the general user, but instead a special administration interface is provided for admins to manage system protection remotely. Again based on the MMC, this proved reasonably easy to navigate and access to the core controls was soon established.

Stability and logging presented no problems, and detection rates were highly impressive as expected, with a concomitant sluggishness in scanning times and overheads as files were subjected to close scrutiny. Unsurprisingly, the Perl tool which tripped up *F-Secure* also produced a false positive here, and thus *Kaspersky* is denied a VB100 award this time despite full coverage of the WildList samples.

Kingsoft AntiVirus 2008.2.22.11

ItW	100.00%	**Polymorphic**	41.19%
ItW (o/a)	100.00%	**Trojans**	48.97%
Worms & bots	99.10%	**False positives**	0

Kingsoft, proud holder of a brace of VB100 awards, has had some problems with stability in recent tests, with detection rates fluctuating wildly from one install to another. No such issues were in evidence this time around however, with a pleasantly designed interface providing ample controls in an easy fashion and scanning holding strong under a heavy onslaught of infected samples.

Detection rates were markedly improved in the set of worms and bots, but still lagging somewhat elsewhere, while the WildList was handled without difficulties. In the clean sets scanning speeds were remarkably slow in both on-demand and on-access measurements, but no false positives were raised and *Kingsoft* thus earns itself a third VB100 award.

McAfee VirusScan Enterprise 8.5.0i

ItW	100.00%	**Polymorphic**	100.00%
ItW (o/a)	100.00%	**Trojans**	95.77%
Worms & bots	100.00%	**False positives**	0

McAfee's product remains a stolid old trooper, unlovely perhaps, but efficient and businesslike with its sensible, unflashy design. Accessing the required controls proved no problem after much exposure to the same interface, and the tests were completed in excellent time, helped along by reasonable scanning speeds and an absence of any wobbliness or other unexpected behaviour.

Detection rates were excellent and reliable, and with no false positives or WildList misses *McAfee* also adds another notch to its VB100 bedpost.

Microsoft Forefront Client Security 1.5.1958.0

ItW	100.00%	**Polymorphic**	95.02%
ItW (o/a)	100.00%	**Trojans**	79.35%
Worms & bots	100.00%	**False positives**	0

Forefront, corporate big brother of *Microsoft*'s *OneCare*, has a slick and very *Windows*-y appearance, with an unsurprising but rather disappointing lack of serious configuration options. On demand at least the defaults were very thorough, with all files and all archive types scanned to an impressive depth, but nevertheless speeds were decent and tests completed in good time with no false positives to upset things.

Scanning the infected sets was similarly free from excessive difficulty, although in larger sets the product's insistence on using the event log as its only usable means of reporting caused some headaches, when large numbers of detections of a single variant tried to squeeze into a single event entry, overflowing it and losing some data. Nevertheless, results were eventually obtained, showing pretty good detection rates and complete coverage of the WildList, thus earning *Microsoft* another VB100 award.

MWTI eScan Internet Security for Windows 9.0.826.233

ItW	100.00%	**Polymorphic**	96.00%
ItW (o/a)	100.00%	**Trojans**	95.98%
Worms & bots	100.00%	**False positives**	1

MWTI's *eScan* is another product based on *Kaspersky Lab*'s *AVP* engine, and as such seemed at risk from the same minor misdemeanour which has brought a couple of products low this month. The installation was smooth, fast and simple, with an automatic scan of system areas and a reboot afterwards, and once running, the interface proved amenable, although accessing the browse function of the on-demand scanner often took rather a long time. As

Archive scanning		ACE	CAB	JAR	LZH	RAR	TGZ	ZIP	ZIP-SFX	EXT*
Agnitum Outpost	OD	X	√	√	X	√	√	√	√	√
	OA	X	X	X	X	X	X	X	X	√
AhnLab V3Net	OD	X/√	X/√	X/√	X/√	X/√	X/√	X/√	X	√
	OA	X	X	X	X	X	X	X	X	√
Alwil avast!	OD	√	√	√	√	√	√	√	√	√
	OA	X	X	X	X	X	X	X	X	X
Arcabit ArcaVir	OD	2	√	√	√	√	√	√	√	√
	OA	X/2	X/√	X/√	X/√	X/√	X/√	X/√	√	√
AVG	OD	X	√	X	X	√	X	√	√	X
	OA	X	X	X	X	X	X	X	X	X/√
Avira AntiVir	OD	√	√	√	√	√	√	√	√	√
	OA	X/√	X/√	X/√	X/√	X/√	X/√	X/√	X/√	X/√
CA eTrust	OD	X	X/√	X/√	X/√	X/√	X/√	X/√	X	√
	OA	X	X	1	X	X	X	1	X	√
ESET NOD32	OD	√	√	√	√	√	5	√	√	√
	OA	X	X	X	X	X	X	X	X	√
Fortinet FortiClient	OD	X	√	√	√	√	√	4	√	√
	OA	X	√	√	√	√	√	4	√	√
Frisk F-PROT	OD	1	√	√	√	√	√	√	√	√
	OA	1	√	2	√	√	√	2	2	√
F-Secure Internet Security	OD	X	5	5	5	5	2	5	5	X
	OA	X/√	X/5	X/5	X/5	X/5	X/2	X/5	X/5	√
Kaspersky Anti-Virus	OD	√	√	√	√	√	√	√	√	√
	OA	X/√	X/√	X/√	X/√	X/√	X/√	X/√	√	√
Kingsoft Internet Security	OD	X	X	√	√	√	√	√	X	√
	OA	X	X	X	X	X	X	X	X	√
McAfee VirusScan	OD	X	X	X	X	X	X	X	X	√
	OA	X/2	X/√	X/√	X/√	X/√	X/√	X/√	X/9	√
Microsoft Forefront	OD	√	√	√	√	√	√	√	√	√
	OA	X	X	1	X	X	X	1	1	√
Moon Secure	OD	X	X	X	X	X	X	X	X	√
	OA	X	X	X	X	X	X	X	X	√
MWTI eScan Internet Security	OD	√	√	√	√	√	√	√	√	√
	OA	√	√	√	√	√	√	√	√	√
Norman Virus Control	OD	X	X	√	√	√	√	√	X	√
	OA	X	X	X	X	X	X	X	X	√
Quick Heal AntiVirus	OD	X/2	X/5	X/5	X	2/5	X/1	2/5	√	X
	OA	X	X	X	X	X	X	X	X	X
Redstone Redprotect	OD	√	√	√	√	√	√	√	√	√
	OA	X/√	X/√	X/√	X/√	X/√	X/√	X/√	X/√	√
Rising Antivirus	OD	√	√	√	√	√	√	√	√	√
	OA	√	√	√	√	√	√	√	√	√
Sophos Endpoint Security and Control	OD	X	X/5	X/5	X/5	X/5	X/5	X/5	X/5	X/√
	OA	X	X/5	X/5	X/5	X/5	X/5	X/5	X/5	X/√
Symantec Endpoint Protection	OD	X	3/√	3/√	3/√	3/√	X/3	3/√	3/√	√
	OA	X	X	X	X	X	X	X	X	√
Trustport Antivirus	OD	X	√	√	X	√	√	√	√	√
	OA	X	√	√	X	√	√	√	√	√
VirusBuster for Servers	OD	X	X	X	X	X	X	X	X	X
	OA	X	X	X	X	X	X	X	X	√

Key:

X - Archive not scanned

√ - Archives scanned to depth of 10 or more levels

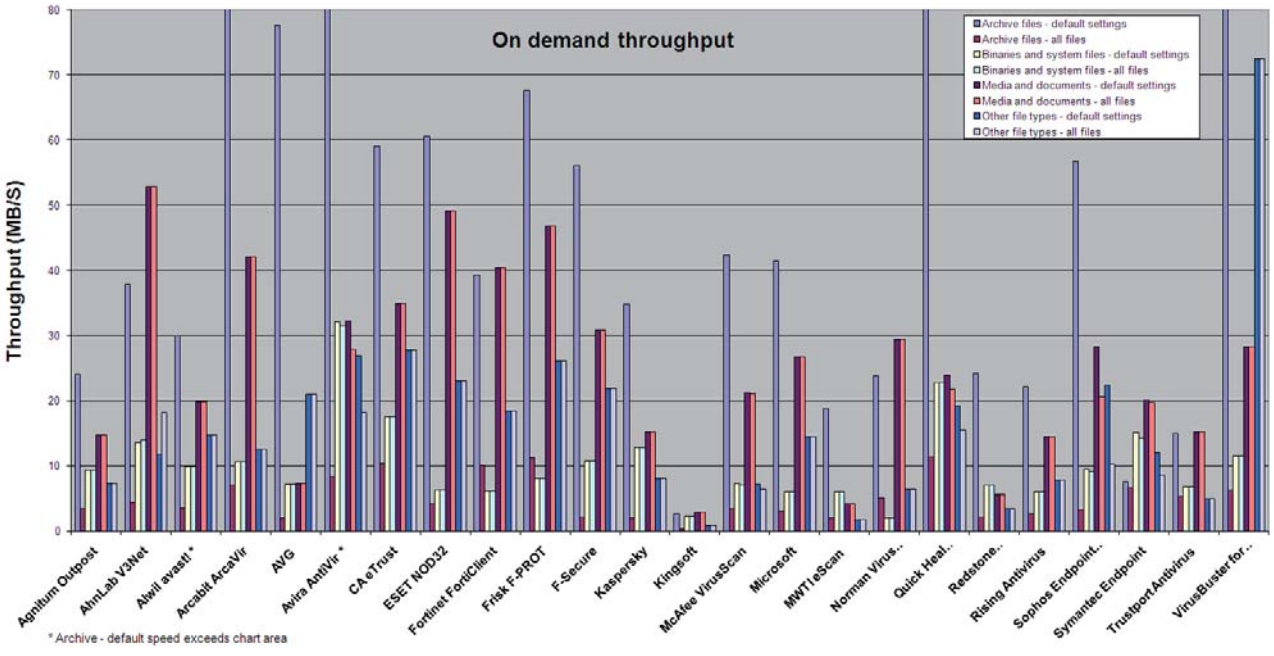
[1-9] - Archives scanned to limited depth

X/√ - Default settings/thorough settings

*Executable file with randomly chosen extension

expected, scanning speeds were less than stellar, but great thoroughness was evident in both the depth and breadth of file types scanned and in the excellent detection rates across the sets.

No problems were encountered in the WildList but, as feared, that pesky Perl utility once again popped up while scanning the clean sets, and this single false positive is enough to spoil *MWTI*'s chances of a VB100 this time.



Norman Virus Control 5.99

ItW	100.00%	**Polymorphic**	78.86%
ItW (o/a)	100.00%	**Trojans**	87.29%
Worms & bots	100.00%	**False positives**	0

After the appearance of a rather unusual new product from *Norman* in the last comparative, it came as something of a relief to see the more familiar version back once more for this test.

The product itself is not without its quirks, with on-demand scans necessitating the use of multiple windows to access configuration, scan design and actual running, but once we had refamiliarized ourselves with this things moved along nicely. Scanning extremely large infected sets proved a rather slow job, presumably as the 'sandbox' system delved deeply into malicious behaviours, but over the clean test sets speeds were splendid in some areas and at least decent in others. Detection rates were similarly reasonable, with no problems in either the WildList or the clean set, and *Norman* thus qualifies for a VB100 award.

Quick Heal AntiVirus Lite 9.50

ItW	100.00%	**Polymorphic**	79.25%
ItW (o/a)	100.00%	**Trojans**	32.42%
Worms & bots	95.16%	**False positives**	1

Quick Heal's product presents a chirpy, friendly face to the world, and continues to justify its name with rapidity in most areas. Installation was a breeze, with a complimentary pre-scan of system areas and no reboot required, and navigating the interface presented no shocks or pitfalls.

Scanning speeds were, well, quick, and overheads barely noticeable, while detection rates were only reasonable, with the trojan set particularly poorly covered. The WildList presented far fewer difficulties however, and a VB100 seemed assured, until a single item in the clean set, a component of the popular 'IrfanView' utility long lurking somewhere in the depths of the set, was mislabelled as a password-stealing trojan. As a result, no VB100 award is granted to *Quick Heal* this month.

Redstone Redprotect Anti-Virus 1.7.1.0

ItW	100.00%	**Polymorphic**	98.03%
ItW (o/a)	99.89%	**Trojans**	90.85%
Worms & bots	100.00%	**False positives**	1

Redprotect is another implementation of the *Kaspersky* scanning engine, aimed here at the managed service arena, and thus with little interaction from end-users intended. A rough engineer's interface is kindly provided to grant some access to the controls without having to resort to registry adjustments, but this was barely needed as sensible defaults were in place across the board. In an improvement on previous performances, the defaults seemed to function as expected throughout. At one point a scan was kicked

On-demand throughput	Archive files				Binaries and system files				Media and documents				Other file types			
	Default settings		All files		Default settings		All files		Default settings		All files		Default settings		All files	
	Time (s)	Through-put (MB/s)	Time (s)	Through-put (MB/s)	Time (s)	Through-put (MB/s)	Time (s)	Through-put (MB/s)	Time (s)	Through-put (MB/s)	Time (s)	Through-put (MB/s)	Time (s)	Through-put (MB/s)	Time (s)	Through-put (MB/s)
Agnitum Outpost	127	24.07	888	3.44	390	9.38	390	9.38	140	14.74	140	14.74	130	7.25	130	7.25
AhnLab V3Net	81	37.86	679	4.50	270	13.55	263	13.91	39	52.92	39	52.92	80	11.78	52	18.12
Alwil avast!	102	29.96	839	3.64	369	9.92	369	9.92	104	19.85	104	19.85	64	14.72	64	14.72
Arcabit ArcaVir	32	94.79	438	6.98	344	10.64	344	10.64	49	42.12	49	42.12	75	12.56	75	12.56
AVG	39	77.60	1450	2.11	509	7.19	509	7.19	284	7.27	284	7.27	45	20.94	45	20.94
Avira AntiVir	35	87.24	368	8.31	114	32.09	116	31.54	64	32.25	74	27.89	35	26.92	52	18.12
CA eTrust	52	59.02	294	10.40	209	17.51	209	17.51	59	34.98	59	34.98	34	27.72	34	27.72
ESET NOD32	50	60.61	734	4.16	579	6.32	579	6.32	42	49.14	42	49.14	41	22.98	41	22.98
Fortinet FortiClient	78	39.30	303	10.09	594	6.16	594	6.16	51	40.47	51	40.47	51	18.48	51	18.48
Frisk F-PROT	45	67.67	272	11.24	454	8.06	454	8.06	44	46.91	44	46.91	36	26.18	36	26.18
F-Secure	54	56.18	1383	2.21	339	10.79	339	10.79	67	30.81	67	30.81	43	21.91	43	21.91
Kaspersky	88	34.85	1489	2.05	286	12.79	286	12.79	136	15.18	136	15.18	117	8.05	117	8.05
Kingsoft	1139	2.68	7722	0.40	1574	2.32	1574	2.32	702	2.94	702	2.94	1126	0.84	1126	0.84
McAfee VirusScan	72	42.45	894	3.42	504	7.26	517	7.08	97	21.28	98	21.06	131	7.19	146	6.45
Microsoft	74	41.49	989	3.09	610	6.00	610	6.00	77	26.80	77	26.80	65	14.50	65	14.50
MWTI eScan Internet Security	162	18.87	1484	2.06	604	6.06	604	6.06	495	4.17	495	4.17	508	1.85	508	1.85
Norman Virus Control	128	23.84	600	5.09	1774	2.06	1774	2.06	70	29.48	70	29.48	147	6.41	147	6.41
Quick Heal AntiVirus	30	103.40	268	11.41	161	22.73	161	22.73	86	24.00	95	21.73	49	19.23	61	15.45
Redstone Redprotect	126	24.25	1385	2.21	522	7.01	522	7.01	363	5.69	363	5.69	269	3.50	269	3.50
Rising Antivirus	138	22.11	1125	2.72	611	5.99	611	5.99	143	14.43	143	14.43	120	7.85	120	7.85
Sophos Endpoint Security and Control	54	56.61	928	3.29	385	9.50	404	9.06	73	28.27	100	20.64	42	22.44	92	10.24
Symantec Endpoint Protection	407	7.51	459	6.66	243	15.06	258	14.18	103	20.04	105	19.66	78	12.08	110	8.57
Trustport Antivirus	204	15.01	568	5.38	537	6.81	537	6.81	136	15.18	136	15.18	190	4.96	190	4.96
VirusBuster for Servers	33	92.47	485	6.30	319	11.47	319	11.47	73	28.27	73	28.27	13	72.49	13	72.49

off with apparently no effect; while the number of files processed rocketed quickly upward, the number actually scanned and, more significantly, the number of detections, remained at zero. Restarting the job rectified things, and the issue was not repeated, but nevertheless it proved a little disquieting. Logging was also a little fiddly, with each handful of detections recorded in a separate XML file, which soon built up to an impressive number, requiring considerable processing power to draw out the required data, but with a little patience this was soon achieved.

As expected, detection results were generally excellent, and speeds more on the medium side, with again that single item in the clean set false alarmed on. Also here, a single sample in the WildList, an autorun type worm, was rather surprisingly not picked up on access, pushing a VB100 award still further from *Redstone*'s reach this month.

Rising Antivirus 2008 20.59.22

ItW	100.00%	**Polymorphic**	60.04%
ItW (o/a)	100.00%	**Trojans**	64.72%
Worms & bots	99.75%	**False positives**	0

Rising, flushed with success after achieving its first VB100 award in the last comparative review, returns to the test bench with what seems to be an identical product. The slick and smooth installer led to a similarly clear and usable interface, accompanied by a cavorting lion cartoon on the desktop, which greatly entertained the new member of the testing team with its antics.

Speeds were a little below par, and detection rates slightly on the patchy side in the polymorphic and trojan test sets, but stability was rock solid throughout the test. No problems were encountered in the WildList, and with no false positives generated either, *Rising* takes home its second VB100 in a row.

Sophos Endpoint Security and Control 7.3.5

ItW	100.00%	**Polymorphic**	92.75%
ItW (o/a)	100.00%	**Trojans**	88.74%
Worms & bots	100.00%	**False positives**	0

Sophos's core product continues a long run with no visible changes, despite much activity in the company's portfolio, and remains a pleasant midpoint between corporate sterility and cartoonish glossiness. As remarked previously, the installer offers the exciting prospect of removing competitors' products from the system before getting underway, and soon has things up and running without the need for a reboot. The initial, fairly lax settings can easily be upped to cover a more thorough range of file and archive types, with some even more in-depth configuration tucked away in a super-advanced section. Scanning moved along at a pleasant pace with no upsets or shocks.

Detection rates were mostly pretty good, and speeds decidedly so. With no problems in either the WildList or the clean set, beyond a fair number of samples flagged as using unusual packing techniques, *Sophos* is awarded a VB100.

Symantec Endpoint Protection 11.0.2020.56

ItW	100.00%	**Polymorphic**	100.00%
ItW (o/a)	100.00%	**Trojans**	93.01%
Worms & bots	100.00%	**False positives**	0

Symantec's product, once dignified and humourless, has veered to the other extreme, with a curvy, gaudy design clearly aimed at the less business-like business user. With the change has come an inevitable reduction in the wealth of options available, but the product remains generally stable and solid.

Opening large logs from within the interface brought the system to a near halt on several occasions, with several long periods of unresponsive, transparent windows to be endured before the required data could be accessed. However, once acquired and parsed, with a great deal of extraneous material discarded, results were much as expected. Speeds were reasonable on demand and very good on access, detection rates pretty high with complete coverage of the WildList, and with no false positives evident *Symantec* earns a VB100 award.

Trustport Antivirus 2.8.0.3007

ItW	100.00%	**Polymorphic**	92.36%
ItW (o/a)	100.00%	**Trojans**	97.43%
Worms & bots	100.00%	**False positives**	0

Trustport's multi-engine approach has fluctuated greatly of late, both in the range of engines available and its success in *VB* testing. Now the company seems to have settled on just two engines: those of *AVG* (here still labelled *Grisoft*, in defiance of the firm's recent name change) and *Norman*. The *AVG* engine appears to be enabled at all times, with the *Norman* engine an extra which is on by default but can be deactivated.

Aside from some strange use of English in the installation process, and some issues with the logging of outsize test sets, no major difficulties were encountered. Speeds were not the best, thanks to the doubling up of engines, but detection rates were highly praiseworthy. In the clean sets, a couple of items were highlighted as using suspicious packing techniques, in wording which came dangerously close to being adjudged false positives, but these were not in the end deemed to be full false alerts.

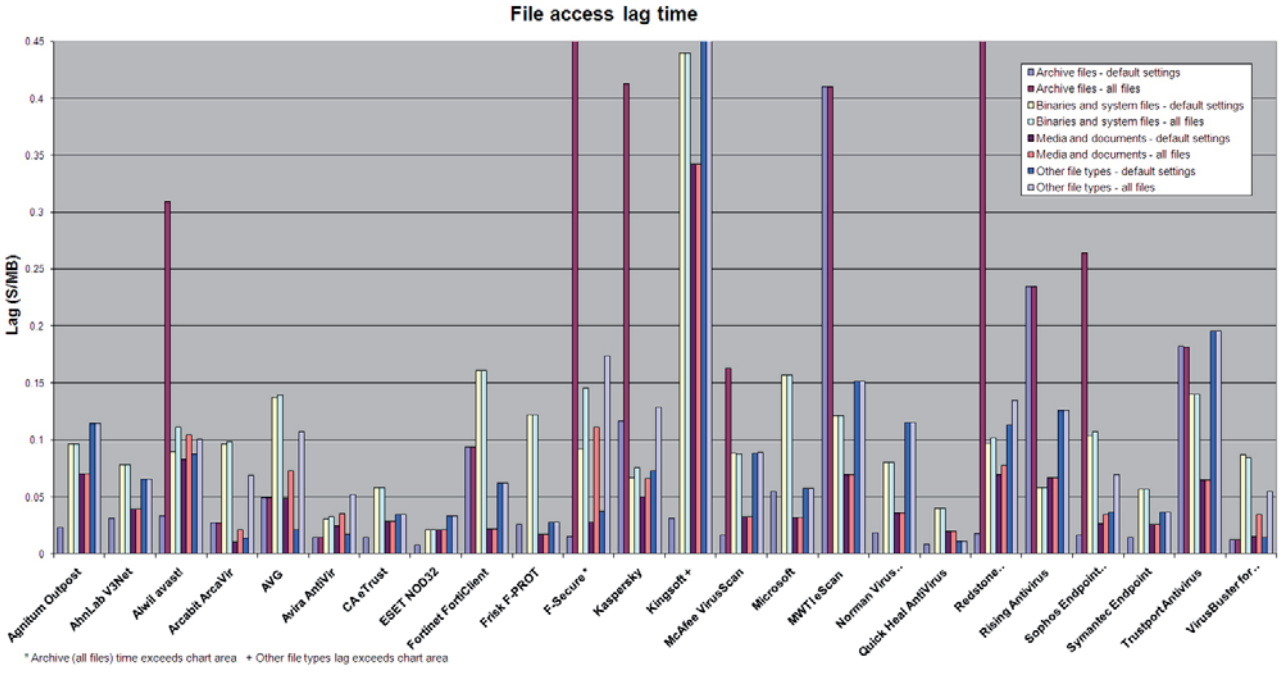
With no other problems *Trustport* scrapes through to a VB100 award after some rocky results in recent months.

File access lag time	Archive files				Binaries and system files				Media and documents				Other file types			
	Default settings		All files		Default settings		All files		Default settings		All files		Default settings		All files	
	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)	Time (s)	Lag (s/MB)
Agnitum Outpost	73	0.02	N/A	N/A	369	0.10	369	0.10	171	0.07	171	0.07	127	0.11	127	0.11
AhnLab V3Net	97	0.03	N/A	N/A	303	0.08	303	0.08	108	0.04	108	0.04	81	0.07	81	0.07
Alwil avast!	103	0.03	948	0.31	346	0.09	424	0.11	197	0.08	242	0.10	102	0.09	114	0.10
Arcabit ArcaVir	85	0.03	85	0.03	370	0.10	377	0.10	48	0.01	69	0.02	32	0.01	84	0.07
AVG	154	0.05	154	0.05	521	0.14	528	0.14	126	0.05	177	0.07	39	0.02	120	0.11
Avira AntiVir	48	0.01	48	0.01	128	0.03	134	0.03	76	0.02	99	0.04	35	0.02	69	0.05
CA eTrust	47	0.01	N/A	N/A	228	0.06	228	0.06	85	0.03	85	0.03	52	0.03	52	0.03
ESET NOD32	25	0.01	N/A	N/A	94	0.02	94	0.02	70	0.02	70	0.02	50	0.03	50	0.03
Fortinet FortiClient	290	0.09	290	0.09	606	0.16	606	0.16	71	0.02	71	0.02	78	0.06	78	0.06
Frisk F-PROT	81	0.03	N/A	N/A	463	0.12	463	0.12	62	0.02	62	0.02	45	0.03	45	0.03
F-Secure	48	0.01	1630	0.53	354	0.09	548	0.15	84	0.03	255	0.11	54	0.04	183	0.17
Kaspersky	360	0.12	1265	0.41	261	0.07	292	0.08	129	0.05	162	0.07	88	0.07	141	0.13
Kingsoft	98	0.03	N/A	N/A	1626	0.44		0.44	733	0.34	733	0.34	1139	1.19	1139	1.19
McAfee VirusScan	53	0.02	501	0.16	340	0.09	338	0.09	93	0.03	93	0.03	102	0.09	103	0.09
Microsoft	169	0.05	N/A	N/A	590	0.16	590	0.16	92	0.03	92	0.03	74	0.06	74	0.06
MWTI eScan Internet Security	1258	0.41	1258	0.41	460	0.12	460	0.12	170	0.07	170	0.07	162	0.15	162	0.15
Norman Virus Control	60	0.02	N/A	N/A	311	0.08	311	0.08	100	0.04	100	0.04	128	0.12	128	0.12
Quick Heal AntiVirus	27	0.01	N/A	N/A	163	0.04	163	0.04	66	0.02	66	0.02	30	0.01	30	0.01
Redstone Redprotect	56	0.02	1390	0.45	372	0.10	390	0.10	170	0.07	187	0.08	126	0.11	147	0.13
Rising Antivirus	719	0.23	719	0.23	229	0.06	229	0.06	164	0.07	164	0.07	138	0.13	138	0.13
Sophos Endpoint Security and Control	54	0.02	811	0.26	397	0.10	410	0.11	80	0.03	97	0.03	54	0.04	85	0.07
Symantec Endpoint Protection	46	0.01	N/A	N/A	223	0.06	223	0.06	79	0.03	79	0.03	54	0.04	54	0.04
Trustport Antivirus	558	0.18	558	0.18	529	0.14	529	0.14	160	0.06	160	0.06	204	0.20	204	0.20
VirusBuster for Servers	41	0.01	41	0.01	334	0.09	324	0.08	56	0.01	97	0.03	33	0.01	71	0.05

VirusBuster for Servers 6.0 build 205

ItW	100.00%	**Polymorphic**	75.77%
ItW (o/a)	100.00%	**Trojans**	78.84%
Worms & bots	99.93%	**False positives**	0

VirusBuster brings up the rear of the test as usual, with much the same product as seen in numerous previous tests and few explicit nods to the server environment. The layout is somewhat esoteric and fiddly, and was not popular with the new member of the team, who was tasked with tackling its strange design to set up a series of scheduled scans over



a long weekend, but once the right technique was hit upon testing was completed tolerably easily, with no serious problems.

Scanning speeds were pretty impressive, quite startlingly so in scanning miscellaneous file types on demand, but on access the option to enable archive scanning seemed not to function as promised. Detection rates were mostly reasonable, though not so hot in the trojan set, but with no false positives or WildList misses *VirusBuster* completes this comparative on a high, winning a VB100 award.

CONCLUSIONS

Another month, another comparative review, this one rendered rather special by the new additional help available in the testing lab, which enabled the review to squeeze in under the wire just before the team heads off to Ottawa for this year's *VB* conference. It was a pretty close call however, with many products taking far longer to get through the test than expected, mainly due to instability under heavy pressure and unexpected, even downright contrary behaviour.

The instability and bad behaviour was most in evidence in the additional testing running parallel with this month's test, trialling a new setup we hope to have fully operational soon. The trial has shown some serious difficulties with persuading some products to behave themselves properly when called on to do their very utmost, meaning that some minor tweaks to the test design may be required prior to the official introduction of these tests, in order to ensure useful data can be obtained and presented in a reasonable time frame.

In the main body of the test, things were much as usual. A few products had some issues with the WildList, with the very pesky W32/Virut#10 once again raising its ugly head after many months on the list. The main reason for products being denied certification, however, was the generation of false positives, with only a handful of files tripping up a sizeable number of products. This was mostly thanks to several products including the same single engine, which in turn mislabelled a single file. This is an indicator of the toughness and the unforgiving nature of the VB100 system, and what makes it such a sought-after and widely respected scheme. Those products that managed to pass should hold their heads up high, while those who didn't quite make it this time, all highly regarded and reliable products, will likely find themselves back up on the podium soon.

Technical details:

All products were tested on identical systems with *AMD Athlon64* X2 Dual Core 5200+ processors, 2 GB RAM, dual 80 GB and 400 GB hard drives, running *Microsoft Windows Server 2008* (32-bit).

Any developers interested in submitting products for VB's comparative reviews should contact john.hawes@virusbtn.com. The current schedule for the publication of VB comparative reviews can be found at http://www.virusbtn.com/vb100/about/schedule.xml.

END NOTES & NEWS

SecTor 2008 takes place 7–8 October 2008 in Toronto, Canada. The conference is an annual IT security education event created by the founders of North American IT security usergroup TASK. For more information see http://sector.ca/.

The 3rd International Conference on Malicious and Unwanted Software (Malware '08) will be held 7–8 October 2008 in Alexandria, VA, USA. The main focus for the conference will be 'the scalability problem'. For more details see http://isiom.wssrl.org/.

Black Hat Japan 2008 takes place 7–10 October 2008 in Tokyo, Japan. Training will take place 7–8 October, with the Black Hat Briefings taking place 9–10 October. For full details see http://www.blackhat.com/.

Net Focus UK 2008 takes place 8–9 October 2008 in Brighton, UK. The event deals with issues of security, personnel, compliance, data privacy, business risk, e-commerce risk and more. For details see https://www.baptie.com/events/show.asp?e=160&xyzzy=2.

The third APWG eCrime Researchers Summit will be held 15–16 October 2008 in Atlanta, GA, USA. eCrime '08 will bring together academic researchers, security practitioners and law enforcement representatives to discuss all aspects of electronic crime and ways to combat it. See http://www.antiphishing.org/ecrimeresearch/.

The SecureLondon Workshop on Computer Forensics will be held 21 October 2008 in London, UK. For further information see https://www.isc2.org/cgi-bin/events/information.cgi?event=58.

RSA Europe 2008 will take place 27–29 October 2008 in London, UK. This year the conference celebrates the influence of Alan Mathison Turing, British cryptographer, mathematician, logician, biologist and 'the father of modern computer science'. For full details see http://www.rsaconference.com/2008/Europe/.

Hack in the Box Security Conference 2008 takes place 27–30 October 2008 in Kuala Lumpur, Malaysia. This year's event will see new hands-on sessions designed to give attendees a closer and deeper understanding of various security issues from physical security bypass methods to the security of RFID and other wireless-based technologies. For more information see http://conference.hackinthebox.org/.

Hacker Halted Malaysia 2008 takes place 3–6 November 2008 in Selangor, Malaysia. For more information see http://www.hackerhalted.com/malaysia.

CSI 2008 takes place 15–21 November 2008 in National Harbor, MD, USA. For online registration see http://www.csiannual.com/.

The SecureDubai Conference on Emerging Threats takes place 4 December 2008 in Dubai, United Arab Emirates. Sessions will engage in the devastating effects and developments of DDoS attacks and how to avoid them, email encryption and the social engineering threat communities pose to a company. For full details see https://www.isc2.org/cgi-bin/events/information.cgi?event=81.

The 2nd Annual Chief Security Officer Summit will take place 8–10 December 2008 in Geneva, Switzerland. The summit aims to bring together security directors from across Europe, Africa and the Middle East to tackle the most critical and strategic security challenges at the highest business level. For more information see http://www.mistieurope.com/cso.

ACSAC 24 (the Applied Computer Security Associates' Annual Computer Security Conference) will be held 8–12 December 2008 in Anaheim, CA, USA. For details see http://www.acsac.org/.

AVAR 2008 will be held 10–12 December 2008 in New Delhi, India. The 11th Association of anti-Virus Asia Researchers International Conference will be hosted by *Quick Heal Technologies Pvt*. See http://www.aavar.org/avar2008/index.htm.

VB2009 will take place 23–25 September 2009 in Geneva, Switzerland. For details of sponsorship opportunities and any other queries relating to VB2009, please email conference@virusbtn.com.

SUBSCRIPTION RATES

Subscription price for 1 year (12 issues):

- Single user: $175
- Corporate (turnover < $10 million): $500
- Corporate (turnover < $100 million): $1,000
- Corporate (turnover > $100 million): $2,000
- *Bona fide* charities and educational institutions: $175
- Public libraries and government organizations: $500

Corporate rates include a licence for intranet publication.

See http://www.virusbtn.com/virusbulletin/subscriptions/ for subscription terms and conditions.

Editorial enquiries, subscription enquiries, orders and payments:
Virus Bulletin Ltd, The Pentagon, Abingdon Science Park, Abingdon, Oxfordshire OX14 3YP, England
Tel: +44 (0)1235 555139 Fax: +44 (0)1235 531889
Email: editorial@virusbtn.com Web: http://www.virusbtn.com/

vb Spam supplement

CONTENTS

NEWS & EVENTS

SOLOWAY REVEALS MOTIVATIONS

Convicted spammer Robert Soloway has spoken out about his life as a prolific spammer in an interview conducted by *NBC News* just prior to the start of his almost-four-year jail sentence. In the interview Soloway estimated that he had been responsible for sending over 10 *trillion* spam messages, most of which he says probably originated from his home PC. Asked about how he felt at the time about the impact his actions had on recipients of his missives, he said he didn't care, and that he felt that anyone who didn't like receiving the messages could simply delete them.

Soloway claims to have made $20,000 a day from spamming at the height of his career – a large proportion of which he spent on property, designer clothes, extravagant holidays and luxury cars. Asked about his motives, Soloway's answer was simple: 'pure greed'. You don't say.

EVENTS

Inbox/Outbox 2008 takes place 25–26 November 2008 in London, UK. The event promises to deliver keynotes from industry experts, more than 30 free seminars each day, practical advice on new developments and common challenges, detailed case study sessions using real-life examples of best practice as well as unique insights into emerging technologies and the latest industry initiatives. For details see http://www.inbox-outbox.com/.

The 15th general meeting of the Messaging Anti-Abuse Working Group (MAAWG) will be held in San Francisco, CA, USA, 17–19 February 2009. The meeting is open to members only. The 16th and 17th general meetings will be held 9–11 June 2009 in Amsterdam, The Netherlands, and 27–29 October 2009 in Philadelphia, PA, USA, respectively. For full details see http://www.maawg.org/.

FEATURE

THE PROBLEM OF BACKSCATTER – PART 2

Terry Zink
Microsoft, USA

Last month, we introduced the problem of backscatter spam, describing what it is and why it is such a problem. We also looked at why it is so difficult to stop. Fortunately, the situation is not hopeless and this month we will look at some of the methods we have at our disposal to help combat this irritating type of spam.

SPECIAL CONTENT FILTERING

One technique we can use to help combat backscatter is to block all NDR messages, or at least tag the phrases and characteristics that commonly occur in NDR backscatter as inputs to a spam filter decision. The Spamnation website [1] recommends some fields to examine, in order of usefulness:

Field	Test	String
From	contains	Mailer-Daemon
From	contains	postmaster@
From	contains	Mail Administrator
Subject	contains	Returned mail
Subject	contains	Failure notice
Subject	contains	Blocked by our bulk email filter
Subject	starts with	Delivery Status Notification
Subject	starts with	Undelivered Mail Returned to Sender
Subject	starts with	Undeliverable
Subject	starts with	Delivery Notification
Body	contains	Status: 5.1.1
Body	contains	Status: 5.7.1

If a spam filter were to look for the above characteristics and block mail based on them, there's a good chance that it would block a very healthy portion of NDR backscatter. The problem is that it would also block a lot of legitimate NDR mail.

Another, riskier, blocking strategy would be to block all mail with a null sender, <>, in the SMTP MAIL FROM field. As many MTAs will send bounces with null senders

(so as not to receive a bounce back if the NDR cannot be delivered), blocking all mail with a null sender will succeed in blocking the bulk of NDRs. However, there are some drawbacks to this:

1. Like the first strategy, blocking on null senders means you will block some legitimate NDRs.
2. Not all null sender messages are NDRs. Some other messages, such as automated reports, use them as well. Other types of quickly written processes, or processes that generate email alerts also use null senders, simply for convenience. Out of office (OOF) notifications also have null senders. Thus, if you block on null senders, you are likely to incur substantial collateral damage.
3. This method is not guaranteed to catch all NDR backscatter. Some MTAs will bounce messages with something like bounces@example.org, postmaster@example.com, or similar.

There are some mail delivery systems that already refuse to route mail with an empty sender due to the excessive abuse of backscatter. These systems are prone to the false positive problems described above. You can get around this by setting inbound 'allow' policy rules for other characteristics, for example, you could have 'allow' rules for certain IPs and 'reject' rules for null MAIL FROMs. The inbound 'allow' rule could supersede the 'reject' rule. The drawback of this strategy is that it creates large numbers of 'allow' rules that become impossible to manage because of all the exceptions to the rule.

So, content filtering based upon From/Subject/Body examination is one way to fight backscatter, albeit with false positive problems. If you have a handle on who you want to accept mail from (no pun intended) then you can create exceptions. But remember, managing exceptions can become a real pain after a while.

USE SPF

Another trick for combating backscatter is to use SPF records. SPF records are designed to help combat backscatter on the theory that the recipient mail server will be able to figure out that your server didn't send it. Here's how it works:

- Bob has his own mail server and creates an SPF record for the domain bobsdomain.com:

  ```
  v=spf1 ip4:256.18.19.0/24 -all
  ```

  Properly interpreted, this means that any message that comes 'from' bobsdomain.com must originate from the IP range 256.18.19.0 – 256.18.19.255.
- Notorious spammer I.M. Obnoxious sends a message to my email address at my domain tzink99@example.com and says it is from my other domain, tzink99@example.org; he spoofs the sender. But tzink99@example.com doesn't exist. My mail server can't do the recipient lookup in real time so it has to '250' accept the body contents of the message.
- Upon trying to deliver the message, my mail server figures out that tzink99@example.com doesn't exist. In a normal world, my mail server would send a message back to tzink99@example.org (with null sender <> in the MAIL FROM field) with a '55x' level error indicating that the message could not be delivered.

However, my mail server is smarter than that. When it accepts the message, it does some spam filtering. It sees that the sending IP for the message is 288.41.18.19. It then looks up the SPF record for example.com and sees it is 256.18.19.0/24. It determines that the sending IP is not in the SPF range for the domain in the MAIL FROM. My mail server sees that any mail coming from example.org that fails the SPF check has a hard fail, '-all'. It assumes that the message is spoofed/forged and decides not to send an NDR back to tzink99@example.org, the purported sender. Instead, it simply drops the message. No backscatter is sent back to me.

The use of SPF records is a way to avoid contributing to the backscatter problem, but this technique is entirely dependent upon the recipient MTA. The recipient MTA must perform an SPF check on the message and then decide not to take its normal course of action on non-deliverable mail. Thus, logic must be built into the MTA to perform custom actions depending on the authentication results.

Not every MTA will do this. SPF checks require DNS queries which are somewhat computationally expensive. It is quicker and easier simply to check to see if the message can be delivered and then take action, rather than check, verify, and then take action. Still, using an SPF record with '-all' in it means that you have given receiving MTAs the help they need in order to determine whether you actually sent the message. Whether or not they use this information is up to them.

CHECK TO SEE IF YOU SENT THE MESSAGE IN THE FIRST PLACE

There is another way to combat backscatter – check to see if you sent the message in the first place. We have already seen that NDR messages and backscatter contain a notice from the bouncing email server as well as all, or part, of the original message. We can use this bounce message to figure out whether or not you sent the message in the first place.

Suppose that my email address is tzink@example.com, and my mail server is mail.example.com. My mail server always sends mail like this:

```
HELO: mail.example.com
MAIL FROM: tzink@example.com
RCPT TO: someguy@example.org
DATA
<etc>
.
QUIT
```

If someguy@example.org were to see this message in his inbox and look at the message headers, he would see a line something like the following:

```
Received: from mail.example.com (mail.example.com
[188.24.229.80])
```

Properly interpreted, the part in parentheses says that this SMTP transaction came from 188.24.229.80 and the mail server HELO'ed as mail.example.com. The recipient mail server did a reverse DNS lookup of 188.24.229.80 and it said mail.example.com. Thus, mail coming from me has Forward-Confirmed Reverse DNS set up.

Suppose the message headers said the following:

```
Received: from host.example.com (unknown
[188.24.229.80] helo=example.com) ...
```

My mail servers don't do it that way. They don't HELO with nothing (that's what unknown means).

```
Received: from [188.24.229.80] (port=12345
helo=example.com)...
```

My mail servers don't HELO with example.com or with a port in the HELO. My reverse DNS is not the IP that I sent it from. Suppose it said the following in the bounced body content Received headers:

```
Received: from mail.example.com (HELO mail.example.
com) [123.123.122.101]...
```

This one comes close, but notice that the IP it came from is not one of my IPs. I can see that the IP should have failed an SPF check (and the recipient mail server should have detected this and not bounced it... tsk, tsk). What if the body contents had said this:

```
Received: from host.example.com (EHLO example.com)
...
```

I also don't HELO with an EHLO. These are all examples of anomalies in the way that my mail servers send mail. You could spot similar abnormalities in the Message-ID tag. If it doesn't conform to the way I generate them, then I know that I didn't send it and that the bounce message did not originate from me.

This method of fighting backscatter is to do it when it hits your inbound server and handle it in a different way from the regular inbound filtering. Verify first that it is a bounce message, for example by looking for the Content-Type: multipart/report; report-type=delivery-status; header.

Next, look for some tell-tale characteristics that say the message came from you. What do the Received headers look like? The HELO line is usually pretty distinctive, and so is the Message-ID. Do you sometimes insert a special X-Header into your outbound messages?

If the message is a bounce message and lacks some of these distinctive features, you can be fairly certain that the message is backscatter (i.e. bounce-back spam that did not originate from you). The advantage of this feature is that you don't have to rely on someone else to do the spam filtering as in the SPF model. Messages that are uniquely yours are difficult to spoof and it is unlikely (though possible) that a spammer would spoof something unique to you in order to spam you. Of course, you wouldn't actually accept a message simply because it looks like it comes from you, you would only not reject it. You would need a stronger method of authentication.

The downside is that you have to insert special distinctive features into your outbound mailer that you can recognize, and you need to have special handling for NDRs that implement custom logic that you wouldn't normally perform on inbound messages. Rejecting mail in this way again imports additional risk because some MTAs will bounce messages back to you and won't conform to the Pirates' Code (see part 1 of this article, *VB*, September 2008, p.S2) of sending DSNs; they may not send back all of the headers or they may modify some of them. In this case you could end up rejecting messages that legitimately came from you, when the recipient MTA just did a lousy job of letting you know.

DON'T MAKE THE PROBLEM WORSE

It would be remiss of me not to include a section on how mail administrators can make sure they do not contribute further to the problem of backscatter [3].

1. ***Don't accept mail, and then bounce***. The primary problem of general backscatter is when email servers accept a message, discover they can't deliver it and then send a bounce notification back to the person who 'sent' the message without verifying that they really sent the message. Remember that if a recipient mail server cannot deliver the mail and figures this out during the SMTP conversation, it rejects delivery and the sending mail server has to generate the bounce. After the recipient mail server has accepted it, it's too late. Make sure you configure your mail server to not do this.
2. ***Don't use Challenge/Response (or allow your users to)***. While Challenge/Response spam filters fix your spam problem, they irritate and annoy everyone else. They offload the spam filtering onto the sender (I send you a message and you expect me to filter it? Hey, filter your own mail!). But even worse, if a spammer spoofs an email address and you send a challenge back to the

sender in the MAIL FROM, you are sending piles of challenge notifications to people who never sent mail. In other words, *you* become a source of backscatter.

3. ***Configure your virus scanner to silently strip or discard malware instead of sending a notification back to the sender***. Viruses and worms that send via email do not have valid sender addresses, they are spoofed. If your mail filter catches the malware and sends a notification back to the sender, it goes to the wrong person, an innocent third party. This is backscatter. This is the same as the Challenge/Response problem.
4. ***Be careful regarding autoresponders, out-of-office notifications, etc***. The issue with autoresponders is the same as the above: a spammer sends a mail to your autoresponder which bounces a message back to the 'sender' (who thinks 'Who is this guy? I don't care if he is out-of-office!').

   (I have mixed feelings on this one. While on the one hand I can see the point of those who are against the use of autoresponders, on the other hand I find OOF notifications very useful. When I send an email to someone at my company and I get an OOF notification, I am glad to know that their response will be delayed. Similarly, when I go on vacation and people send me mail, it is useful to let them know that my response will be delayed. So, in the business world I believe that there is a place for the autoresponse/OOF notification. Maybe you should only send auto-responses to senders who pass a DKIM or SPF check.)

Even if we all can't stop backscatter, let's not make it worse by contributing to it. Don't send mail automatically if you can't deliver it or without verifying that whoever sent the message actually sent it.

AN IDIOSYNCRASY

There are a number of competing mail transfer agents out there, including *Microsoft Exchange*, *Postfix*, *Sendmail*, *qmail*, *Exim*, and so forth. I am not an expert on MTAs but I know a few things about them.

Qmail is a mail transfer agent that runs on Unix. It was written by Daniel J. Bernstein [4] as a more secure replacement for the popular *Sendmail* program. When first published, *qmail* was the first mail transport agent that was written with security in mind. Other MTAs with security concerns addressed have been written since then.

Qmail is designed to accept mail for all of its domains and not perform any recipient validation. If the recipient doesn't exist, it generates an NDR. In other words, it accepts, then bounces. This, of course, can generate backscatter.

Another thing about some older versions of *qmail* is that it doesn't put in the following header in a bounce:

```
Content-Type: multipart/report; report-type=delivery-status;
```

I know this because recently I had to investigate a customer complaint. They claimed that because the bounce message did not have that header (which is required by the Pirates' Code), the message was not backscatter; rather, it was a spammer *spoofing* backscatter and getting through the filters. While it's certainly possible that a spammer would do this, it turned out to be a *qmail*-generated bounce message that did not include the header.

Qmail does put this in the bounce message, however:

```
Hi. This is the qmail-send program at mail.someorganization.com.

I'm afraid I wasn't able to deliver your message to the following addresses.

This is a permanent error; I've given up. Sorry it didn't work out.
```

It also inserts the following header:

```
Received: (qmail 9999 invoked for bounce);
```

So, it is quite interesting that some *qmail* implementations do not include the Content-Type header but do include the above bounce message. It does not quite comply with the RFC (or Pirates' Code), but it's part way there.

The moral of the story is: if you are running *qmail*, don't just use the default implementation. There are some patches out there [5] that will fix the accept-then-bounce behaviour.

SUMMARY

In this article we have seen some techniques for minimizing the backscatter issue. In the next article, we will examine a technique that has a much greater success rate at blocking backscatter – Bounce Address Tag Validation.

REFERENCES & NOTES

[1] http://www.spamnation.info/notes/guides/BackscatterFAQ.html.

[2] http://www.postfix.org/BACKSCATTER_README.html.

[3] I have borrowed and paraphrased these suggestions from: http://www.spamresource.com/2007/02/backscatter-what-is-it-how-do-i-stop-it.html.

[4] http://cr.yp.to/qmail.html.

[5] See http://marc.theaimsgroup.com/?l=qmail&m=108605073822238&w=2 or http://qmail.jms1.net/patches/validrcptto.cdb.shtml.