JANUARY 2007

virus BULLETIN

Fighting malware and spam

CONTENTS

IN THIS ISSUE

A MERRY DANCE

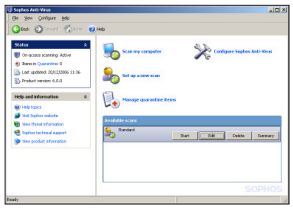
OSX/Macarena is the first parasitic infector of Mach-O files. Peter Ferrie has all the details.
page 4

PREPENDING CONUNDRUM

W32/Nubys-A looked, at first glance, like a trojan downloader. However, most samples contained not one, but several legitimate PE files in the appended data. Samples with one appended executable would have suggested a prepending virus, but why several? Robert Poston makes sense of the conundrum.
page 6

PRODUCT REVIEW

John Hawes takes an in-depth look at the latest version of *Sophos*'s full cross-platform, multi-component suite, *Sophos Enterprise Security*.



page 15

vbSpam supplement

This month: anti-spam news and events; and Gordon Cormack reports on the TREC 2006 spam filter evaluation track.

ISSN 1749-7027

'The malware research community [is] the authority with regard to assisting newcomers in the adoption of safe practices.'

Ryan Hicks, Earthlink, USA

DÉJÀ VU ALL OVER AGAIN

Since the turn of the century the malware landscape has been changing steadily. Previously, most attention and effort was focused on the problems created by malicious self-replicating code. Even though trojans had existed and caused problems for some time, viruses and worms were considered the only threats worthy of attention. However, the situation has since changed.

Initially, there was debate in the research community as to whether or not early trojans (e.g. simple keyloggers, autodialers, etc.) constituted enough of a threat to warrant detection and cleaning. But as vendors started adding trojans to their definition sets, another problem arose. Certain companies were producing software, allegedly within legal bounds and/or with user consent, but which could otherwise be considered malware. The combination of the AV industry's reluctance to detect trojans and the legal wrangling left a gap that was later filled by the anti-spyware industry.

But the separation in focus didn't last long. Anti-virus (AV) vendors created their own anti-spyware products through acquisition, in-house development, or both, and anti-spyware vendors began adding anti-virus capability through partnerships or in-house development. The two sides of the industry have come closer together and will likely soon become indistinguishable.

The rise of the anti-spyware industry was not limited simply to technological or product development. Difficult policy and law enforcement issues also needed to be resolved. While viruses and worms can be said always to be unwanted, spyware is not as easily classified.

At the forefront of addressing these issues is the Anti-Spyware Coalition (ASC). Among the myriad issues with which the ASC is concerned are issues of which the AV industry and research community already has a vast amount of knowledge and experience: sample sharing and safe handling, participant vetting, and control of information dissemination.

The transition of the AV community from focusing on self-replicating malware to the inclusion of non-replicable malware is still under way, and already yet another threat has become a significant problem: phishing. The Anti-Phishing Working Group (APWG) brings together policy makers, law enforcement bodies, customers and vendors to decide the issues related to phishing. Like the anti-spyware community, anti-phishing efforts are faced with issues that are well known to the AV industry: sample sharing and safe handling, participant vetting, and control of information dissemination.

It is apparent that this represents a massive duplication of effort. Organizations at the forefront of the latest software security issues are spending time and effort developing policies and procedures that the AV vendor and research community already has in place. Even though the AV industry is well represented in the ASC and the APWG, the technical and procedural efforts should be more visibly led by the AV research community.

For nearly two decades, the AV research community has developed proven procedures for every aspect of malware research. The newer threats of spyware and phishing will require new policies, best practices, and new laws as to the investigation and prosecution of offenders. However, the concerns regarding the sharing of samples with trusted community members, the safe handling of those samples, vetting and acceptance of new members in the research community, and the dissemination of sensitive information, remain the same.

New organizations such as the ASC and APWG are being created to address the greater issues of how to deal with new threats. While the malware research community may not be expert in the creation of policy or law enforcement, we are the authority with regard to assisting newcomers in the adoption of safe practices. As such, it is incumbent on the malware research community to take the lead and establish a means by which newcomers can benefit from our knowledge and experience.

NEWS

SONY ROOTKIT SETTLEMENT COSTS ESCALATE

Sony BMG Music Entertainment's ill-advised use of hidden digital rights management (DRM) software on its CDs in late 2005 has cost the company $5.75 million in settlement fees.

Last month *Sony* agreed to pay a combined total of $1.5 million to settle lawsuits filed by the states of California and Texas over its use of hidden DRM software on CDs. Two days later, the company agreed to pay settlement fees to a further 40 states to end the investigations into its use of the copy protection programs.

In late 2005, Mark Russinovich of *Sysinternals* (now *Microsoft*) was first to pick up on the security risks concerning the copy protection software, revealing that the software was using rootkit cloaking techniques (see *VB*, December 2005, p.11). According to the Massachusetts Attorney General, more than 12 million CDs shipped containing the software.

Residents of each of the US states that have settled with *Sony* are entitled to up to $175 in refunds for damages that may have been caused to their computers while attempting to uninstall the software. *Sony* has set up a website (http://www.sonybmgcdtechsettlement.com/) with information for consumers on the matter.

MMS MOBILE PHONE EXPLOIT RELEASED

Last month saw the publication of proof-of-concept code exploiting a vulnerability in the popular mobile phone Multimedia Messaging Service (MMS).

Security researcher Collin Mulliner discovered the vulnerability over six months ago and reported it to software vendors, but having received no satisfactory response, chose to publish the exploit at December's Chaos Communication Congress in Berlin.

The vulnerability resides in the SMIL (Synchronized Multimedia Integration Language) protocol used in MMS messages. Region tags in MMS SMIL are vulnerable to buffer overflow causing arbitrary code execution.

So far only two devices have been confirmed as vulnerable: the *IPAQ 6315* and *i-mate PDA2k*, but it is believed that other devices running *Pocket PC 2003* and *Windows Smartphone 2003* are also likely to be at risk. However, as researchers at AV firm *F-Secure* were quick to point out, exploitation would be difficult in any device – since an attacker would need to guess the correct memory slot where the MMS processing code is executing and send appropriate exploit code – and malicious MMS messages would therefore be more likely to crash a device rather than infect it.

Prevalence Table – November 2006

Virus	Type	Incidents	Reports
W32/Mytob	File	3,507,477	28.67%
W32/Netsky	File	3,337,363	27.28%
W32/Bagle	File	2,413,696	19.73%
W32/MyWife	File	1,048,023	8.57%
W32/Zafi	File	441,357	3.61%
W32/Mydoom	File	402,981	3.29%
W32/Lovgate	File	372,897	3.05%
W32/Bagz	File	357,269	2.92%
W32/Parite	File	74,444	0.61%
W32/Stration	File	43,793	0.36%
W32/Tenga	File	30,716	0.25%
W32/Mabutu	File	27,704	0.23%
W32/Klez	File	26,011	0.21%
W32/Funlove	File	24,350	0.20%
W32/Elkern	File	20,861	0.17%
W32/Valla	File	11,850	0.10%
W32/Reatle	File	10,211	0.08%
W32/Bugbear	File	9,000	0.07%
VBS/Redlof	Script	8,987	0.07%
W32/Maslan	File	8,046	0.07%
W32/Agobot	File	7,996	0.07%
W32/Sober	File	7,529	0.06%
W32/Lovelorn	File	6,156	0.05%
W32/Dumaru	File	4,892	0.04%
W32/Sality	File	3,713	0.03%
JS/Kak	Script	3,676	0.03%
W32/Plexus	File	2,005	0.02%
W32/Gurong	File	1,812	0.01%
W97M/Thus	Macro	1,593	0.01%
W32/Rontokbro	File	1,482	0.01%
W32/Chir	File	1,366	0.01%
W95/Tenrobot	File	1,117	0.01%
Others[1]		12,955	0.11%
Total			100%

[1]The Prevalence Table includes a total of 12,955 reports across 60 further viruses. Readers are reminded that a complete listing is posted at http://www.virusbtn.com/Prevalence/.

VIRUS ANALYSIS 1

DO THE MACARENA

Peter Ferrie
Symantec Security Response, USA

On 31 October 2006 we received a sample of the first parasitic infector of Mach-O files, OSX/Macarena. The file had previously been uploaded to a popular VX site. In contrast to OSX/Leap, which relied on a resource fork to contain the virus code, Macarena understands the Mach-O file format sufficiently well to parse the necessary structures correctly and inject its code directly into a file.

MACH-O FORMAT

Every Mach-O file begins with a header structure. That structure is called the mach_header. It begins with a magic number, whose value depends on the architecture on which the Mach-O file will execute. Though it is declared as a 32-bit value, it is easier to consider it as a sequence of four bytes. Thus, for the 32-bit *Intel* x86 architecture, the value is 0xCE 0xFA 0xED 0xFE. For the 32-bit *PowerPC* architecture, the value is 0xFE 0xED 0xFA 0xCE ('feed face'). For the 64-bit *PowerPC* architecture (currently the only supported 64-bit format), the value is 0xFE 0xED 0xFA 0xCF.

Following the magic number is a value specifying the CPU family. For the *Intel* architecture, the value is 7. For the 32-bit *PowerPC* architecture, the value is 0x12. For the 64-bit *PowerPC* architecture, the value is 0x1000012. While the *Intel* and *PowerPC* architectures are the most common types that will be seen, other CPU values can be specified, such as the *VAX*, *Motorola* 680x0, *MIPS*, *ARM*, and the *Sparc*. These CPU values exist because the underlying operating system is based on a variant of *BSD*, which supports these CPUs. There is also a value that specifies the CPU subtype, to specify the required CPU more exactly. For the *Intel* and *PowerPC* architectures, a special value exists to specify that the file can run on any member of that architecture family.

The filetype field specifies the internal file format. The three most common types are: Object, Executable and Library. There are other types, such as Core, which usually contains crash-dump information; and Symbol, which contains symbol information for a corresponding binary file.

The next two fields relate to the array of 'load commands' that follow the mach_header structure. The first field contains the number of those load commands, and the second field contains their size.

The last field in the 32-bit mach_header structure contains a set of flags that describe some optional characteristics that can affect the loading of the file (the 64-bit *PowerPC* format has an additional reserved field for alignment purposes, but is otherwise identical to the 32-bit format). Most of the flags relate to file linking, and their effects are not relevant to the description of the virus.

Load commands exist to allow a file to specify various different characteristics within the file, including the memory layout and contents. Some of these characteristics include 32-bit and 64-bit segment descriptions, symbol table descriptions, dynamic library descriptions, dynamic linker descriptions, entrypoints for executables and libraries, and framework descriptions. Each load command contains a field that specifies the type of the command that follows, and the size of the command that follows. This allows an application to skip any command that it does not understand, or that it does not find interesting.

As far as the virus is concerned, only the segment descriptions and the executable entrypoint are relevant.

SEGMENTS

Segments are described by a structure called the segment_command. The segment_command structure begins with a segment name, followed by the address and size of the segment itself. There are two address fields, and two size fields. The first address and size fields are the virtual values (the address and size in memory), the second address and size fields are the physical values (the offset and size in the file). The term 'segment' in Mach-O files is roughly equivalent to the term 'section' in the *Windows* Portable Executable format (but in Portable Executable files, the address and size fields are in the reverse order). Interestingly, Mach-O files also contain 'sections', and are described in detail below.

All segments must be aligned on a 4kb boundary, otherwise a bus error occurs when attempting to load them. This is documented in *Apple*'s ABI for Mach-O files.

Following the address and size fields are two protection fields. The first field specifies the maximum protection that a segment can acquire. The second field specifies the initial protection that a segment can acquire. The possible protection values are: Read, Write and Execute. Currently, *OSX* does not implement 'W^X' protection (a method for the mutual exclusion of writable and executable protections, to limit the ability of some types of exploits to execute), though this might be implemented in the future. The first version of Macarena uses Read/Write/Execute protection for the segment in which it resides. Perhaps in response to the possibility of 'W^X', the second version of Macarena uses Read/Execute protection alone.

The next field in the segment_command structure contains the number of section data structures that follow the current

segment_command structure. The final field in the segment_command structure is a set of flags. One possible flag specifies that the segment should be loaded to the top of memory; another possible flag specifies that the segment contains no relocated data.

The 64-bit version of the segment_command structure is identical in format to the 32-bit version of the segment_command structure, but with all of the address and size fields expanded from 32 bits to 64 bits.

SECTIONS

Sections are regions of memory that subdivide a segment. They are described by a section structure, and the sections within any given segment follow the segment_command structure immediately. Sections begin with a section name, followed by the name of the segment that contains it. The next four fields are the address in memory, the size and offset in the file, and the section alignment. The next two fields contain the offset of any relocation data, and the number of relocation items. The final three fields are a set of flags, and two fields whose interpretation depends on the type of section. Usually these last two fields will contain a value of zero.

The 64-bit version of the section structure is identical in format to the 32-bit version of the section structure, but with only the address and size fields expanded from 32 bits to 64 bits. This causes a slight limitation: while a segment can refer to file data beyond the 4Gb range, a section cannot.

It is legal to have a segment that contains no sections. In fact, most files contain an example of this: the __PAGEZERO segment describes a 4kb region of memory with no protection attributes set. It is intended to contain no file data, and thus be simply a virtual memory region that will cause an exception if it is accessed for any reason. Its purpose is to allow interception of certain invalid pointer usage, since that is a sign of a programming bug.

DOING THE MACARENA

While the __PAGEZERO segment is intended to contain no file data (size in file field has a value of zero), there is no reason why it cannot contain file data. Since it is really a segment like any other, if the file offset and size fields are set to any legal value, and if the segment protection flags are changed to at least Read, the segment becomes accessible. If the segment protection flags are changed to Executable as well, then code can be executed directly from there.

This is exactly what Macarena does. When infecting a file, it pads the file size to a multiple of 4kb (a segment requirement, as noted above), then appends itself. The __PAGEZERO segment is altered to point to the virus code that starts immediately after the padding, and the segment protection flags are changed as described above, depending on the version of the virus. The change to the segment protection flags acts as the infection marker.

THREADS

The final piece of the puzzle involves how the virus gains control. The method is straightforward – the UnixThread load command contains the initial values for all of the CPU registers for the specified architecture. This includes the Instruction Pointer (EIP for the *Intel* architecture, and SRR0 for the *PowerPC* architecture). By altering the Instruction Pointer register to the required virtual address, the code at that location will be executed when the file is loaded. Macarena changes the Instruction Pointer register to zero, the start of the __PAGEZERO segment. This is apparently an unexpected value for some tools such as *GDB* and *IDA*, with the result that the virus code is not shown.

LIFE, THE UNIVERSE, AND EVERYTHING

Macarena is a simple virus. When executed, it enumerates the files in the current directory, and for any file of normal executable type, the virus will attempt to infect it, if it has not been infected already. This algorithm was obviously sufficiently simple for someone to learn enough *PowerPC* assembler to port and release a *PowerPC* version of it a week later. The *PowerPC* version is functionally identical to the *Intel* version, apart from infecting files for the 32-bit *PowerPC* architecture instead.

Universal files describe multiple architectures, allowing an executable to run on multiple platforms. They are not actually Mach-O files themselves. Rather, they are archives that contain multiple Mach-O files. Since this is the more common format for files on the *OSX* platform, it is likely that we will see viruses that understand the Universal file format and can infect the target architecture within them.

If that should happen, we might need to learn some new moves.

OSX/Macarena

Type:	Parasitic, direct-action Mach-O infector.
Size:	528 bytes (.A), 504 bytes (.B), 840 bytes (PPC).
Payload:	None.
Removal:	Delete infected files and restore them from backup.

VIRUS ANALYSIS 2

THE GREAT PREPENDER: W32/NUBYS-A

Robert Poston
Sophos, UK

When samples of W32/Nubys-A first came in for analysis it looked, at first glance, like a trojan downloader. However, it was spreading rapidly across the network of a large international company, and most samples contained not one, but several legitimate PE files in the appended data. Samples with one appended executable would have suggested a prepending virus, but why several? It did not make sense.

A CLOSER LOOK

Analysis of the malware code revealed two major pieces of functionality:

- A thread to download and execute six files from a website. Samples of these files were obtained, and found to be password stealers for popular online games.
- A subroutine, periodically called from a timer, using the API call *WnetEnumResourceA* to find executable files to infect on the network. The virus extracted the correct icon from the target host, and then prepended itself to the file.

So this was indeed a prepending virus, and its purpose was clear: to spread the password-stealing trojans silently, as widely as possible, without drawing attention to itself. Infected hosts should appear to run as normal.

A SILLY MISTAKE

The author got one thing wrong. Upon finding a new host to infect, a prepending virus would normally just prepend its own viral code. However, this author's infection routine prepended the whole of the currently executing viral file – including any previously infected hosts. Furthermore, the infection marker 'by USA!' (see Figure 1) was appended each time. The URL for the downloads was also part of the appended data, but appeared just once. Figure 2 illustrates the resulting file structure after each generation of infection. Except for the marker, which is of fixed length, each piece of appended data was prefixed by a decimal text string giving its length.

```
000D60A0 0000 0000 0000 0000 0000 0000 0000 0000 ................
000D60B0 0000 0000 0000 0000 0000 0000 0000 0000 ................
000D60C0 0062 7920 5553 4121 0000 0031 3633 3834 .by USA!...16384
000D60D0 3000 0000 004D 5A90 0003 0000 0004 0000 0....MZ.........
000D60E0 00FF FF00 00B8 0000 0000 0000 0040 0000 ...............@..
000D60F0 0000 0000 0000 0000 0000 0000 0000 0000 ................
000D6100 0000 0000 0000 0000 0000 0000 0000 0000 ................
000D6110 00F8 0000 000E 1FBA 0E00 B409 CD21 B801 .............!..
000D6120 4CCD 2154 6869 7320 7072 6F67 7261 6D20 L.!This program
000D6130 6361 6E6E 6F74 2062 6520 7275 6E20 696E cannot be run in
000D6140 2044 4F53 206D 6F64 652E 0D0D 0A24 0000  DOS mode....$..
000D6150 0000 0000 00E9 6021 8FAD 014F DCAD 014F ......`!...O...O
```

Figure 1: The infection marker, 'by USA!', followed by a length string and a new host.

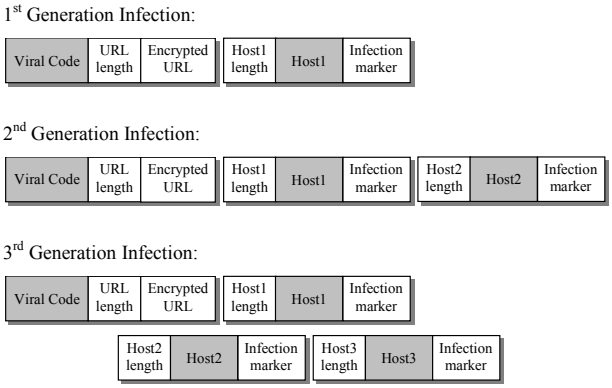


Figure 2: File structure for three generations of infection.

This explains the chains of appended files observed at the beginning. Each successive infection produced a longer and longer sequence. It also explains why several of the samples had the same sequence of initial hosts.

THE COST OF MISTAKES (1)

Each time an infected file was executed, the virus would execute its downloading and infection routines then drop and execute a temporary copy of Host1 (using a filename of '~' followed by the original file name).

This is where the virus writer's mistake becomes clear: for second and subsequent generations it should be the final host that is executed. For example, suppose the first generation is an infected copy of notepad.exe. For the user, this file will appear to execute as normal. However, if the infected notepad.exe itself infected, say, *Internet Explorer* then when the user next attempted to run *Internet Explorer* they would find notepad executing instead.

This would at least raise the alarm that something has been tampering with files, which is not what the virus author wanted. It is also unfortunate for the infected user, since it will break many applications.

THE COST OF MISTAKES (2)

It is doubtful that malware authors do much quality control testing – they don't care if they mess up other people's computers. Conversely, anti-virus companies need to produce a response that is fast, yet accurate. For a new

trojan or worm it is often possible, after only a few minutes' analysis, to confirm the malicious nature of the file and to have detection quality control tested and published soon afterwards. For a virus, detection is not enough. Simply deleting infected files would also delete the original hosts, so where possible disinfection is provided to restore infected files to their original form.

Thus a detailed analysis of the infection mechanism is required, and a way must be found to reverse its actions. For a complicated polymorphic virus analysis this may take days, while for a non-polymorphic prepender, like W32/Nubys-A, a response can usually be made within a few hours. However, care must still be taken to understand what is going on. When the virus writer makes mistakes, the anti-virus researcher must be careful not to fall into the same trap.

For W32/Nubys-A this means that, instead of restoring the first appended host, the final one must be restored. One possible strategy to locate this host would be to scan backwards from the end of the file for an MZ and PE header. However, this would be slow and, in certain situations, unreliable. Thankfully, the infection mechanism of W32/Nubys-A supplies sufficient information for a much better disinfection routine. The viral code is of a fixed length, so it is possible to locate and read the length of the encrypted URL, and from there to calculate the position of each successive length field until the last one is reached. This identifies the correct part of the file to restore.

CONCLUSION

Mistakes made by malware and spam authors are nothing new. There are thousands of damaged variants of the Netsky worm that fail to execute. There are spam campaigns that send out millions of gibberish emails. For their authors, these careless mistakes are not a problem. They are not accountable to anyone, and they do not take responsibility for the effects of their creations.

However, for the researchers who take on the responsibility of clearing up other people's mess, this can create some interesting challenges. Writing a disinfection algorithm for W32/Nubys-A is just one example of the importance of a flexible and extensible anti-virus engine. It is thanks to such technologies that most problems have a solution.

W32/Nubys-A

Type:	Non-polymorphic prepending virus.
Aliases:	Trojan-Downloader.Win32.Agent.bam.

FEATURE

THE REAL MOTIVE BEHIND STRATION

Ivan Macalintal
Trend Micro, USA

Recently the anti-virus and computer security industry has focused increasingly on targeted trojan attacks, trojan downloaders and spyware/adware rather than the mass-mailers that plagued cyberspace just a few years ago. However, just as it seemed that mass-mailers were dying away, a new breed emerged: Stration (aka Warezov, or Strat).

The first variant appeared on 16 August 2006 and was given the detection name WORM_STRATION.A. After only two months *Trend Micro* had received well over 150 variants, the most recent of which (at the time of writing this article) is WORM_STRAT.EQ, detected on 25 October.

At first, the behaviour of the Stration worms was perplexing. They exhibited features much like those used by previous mass-mailers, but there were differences:

- The worms exhibited bursts of 'spiked attacks' or continuous massive spamming in short time frames.
- Stration's downloader components used various top-level domains as infection vectors.
- Stration appeared to have a financial motive, unlike previous worms whose only purpose was to spread to as many computer systems as possible, as quickly as possible.

This paper attempts to reveal the underlying motive of the seemingly random and nonsensical outbursts of the Stration worm.

ANALYSIS OF THE THREAT

On the surface, the Stration attacks look like a pointless series of worm propagation, but further investigation shows that this is not the case. Let's take one recent variant's behaviour as an example: WORM_STRAT.DV.

The worm is downloaded from one of the many URLs that Stration uses as infection vectors. Once executed, it drops a number of Dynamic Link Library (DLL) files in the system directory:

file	size
attstat.dll	143,360 bytes
confatt.dll	53,248 bytes
attprf32.dll	53,248 bytes
attperf.exe	40,960 bytes
attmgr32.dll	356,352 bytes
atrconf.exe	49,152 bytes

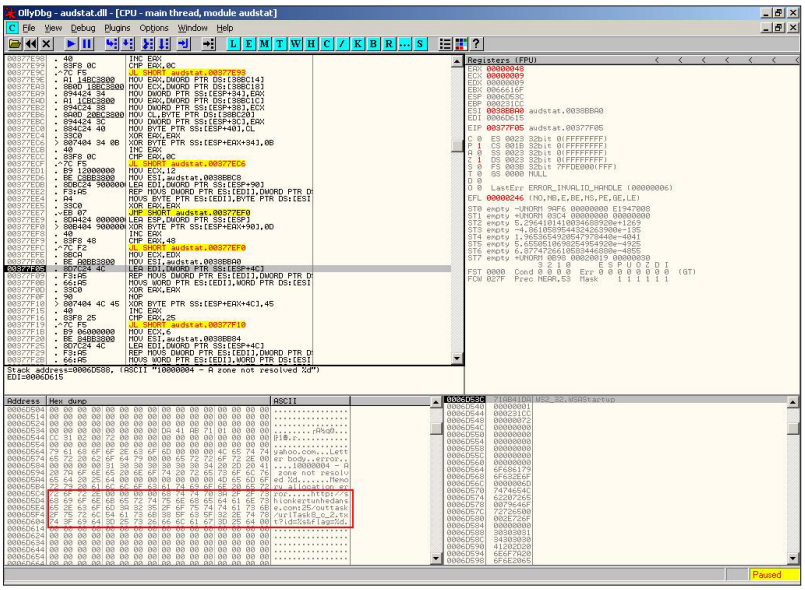


Figure 1: audstat.dll decrypts URLs in memory.

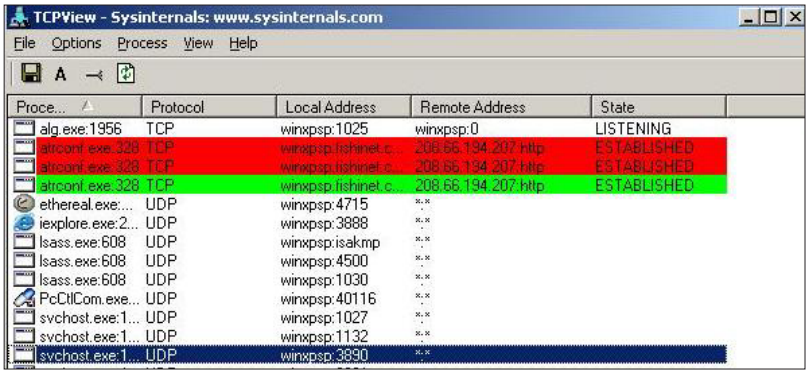


Figure 2: TCPView screenshot.

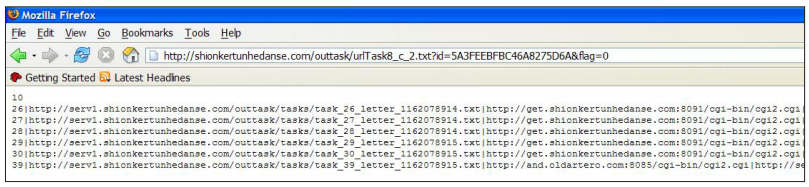


Figure 3: Loading the URL in a browser.

We will take a closer look at two files: audstat.dll and atrconf.exe. After execution, audstat.dll decrypts a number of URLs in memory, as shown in Figure 1.

The second file in question, atrconf.exe, connects to the URL using IP address 208.66.194.207, as shown in Figure 2. The decrypted URL is http://shionkertunhedanse.com:25/outtask/urlTask8_c_2.txt?id=%s&flag=%d.

Figure 3 shows the result of loading the URL into a browser. The contents are dynamic, changing upon every reload of the page. The following is an example of the content:

```
26|http://serv1.shionkertunhedanse.com/
outtask/tasks/
task_26_letter_1162078914.txt|http://
get.shionkertunhedanse.com:8091/cgibin/
gi2.cgi|http://
serv1.shionkertunhedanse.com/
report2.cgi|1||http://
mail.oldartero.com:8888/cgi-bin/put|

27|http://serv1.shionkertunhedanse.com/
outtask/tasks/
task_27_letter_1162078914.txt|http://
get.shionkertunhedanse.com:8091/cgibin/
gi2.cgi|http://
serv1.shionkertunhedanse.com/
report2.cgi|1||http://
mail.oldartero.com:8888/cgi-bin/put|

28|http://serv1.shionkertunhedanse.com/
outtask/tasks/
task_28_letter_1162078914.txt|http://
get.shionkertunhedanse.com:8091/cgibin/
gi2.cgi|http://
serv1.shionkertunhedanse.com/
report2.cgi|1||http://
mail.oldartero.com:8888/cgi-bin/put|

29|http://serv1.shionkertunhedanse.com/
outtask/tasks/
task_29_letter_1162078915.txt|http://
get.shionkertunhedanse.com:8091/cgibin/
gi2.cgi|http://
serv1.shionkertunhedanse.com/
report2.cgi|1||http://
mail.oldartero.com:8888/cgi-bin/put|

30|http://serv1.shionkertunhedanse.com/
outtask/tasks/
task_30_letter_1162078915.txt|http://
get.shionkertunhedanse.com:8091/cgibin/
gi2.cgi|http://
serv1.shionkertunhedanse.com/
report2.cgi|1||http://
mail.oldartero.com:8888/cgi-bin/put|
```

Let's look at the first two URLs in the first row. Figure 4 shows what we will see when the first URL is loaded into a web browser.

THE MOTIVE

Looking at Figure 4, we can see that it's an email template. We saved the contents of the template and renamed it with an .EML extension. The result is shown in Figure 5 – it's an image spam advertising Viagra and other drugs.

Figure 6 shows the result of loading the second URL into a web browser. This URL resolves to a site containing a list of email addresses. These are the target recipient email addresses that Stration uses to send the image spam. The number of email addresses found here continues to increase at the time of writing this article. They are gathered from the Internet via blogs, forums and mailing lists, as well as from infected PCs.

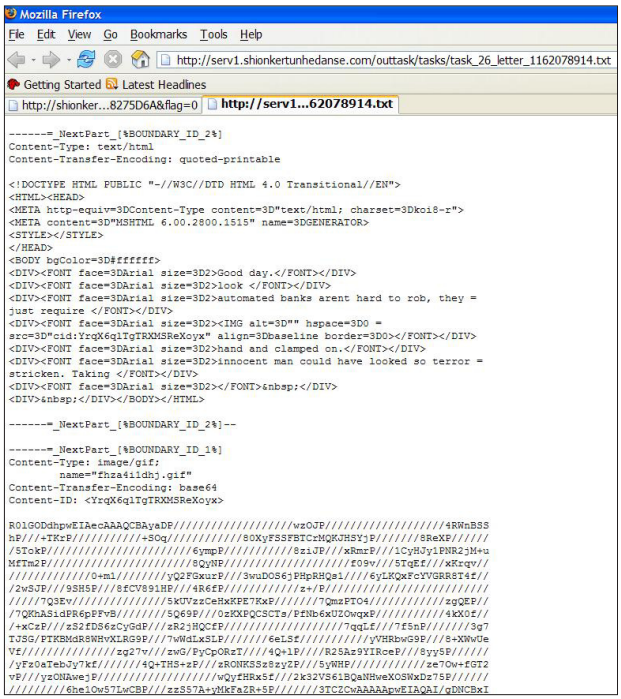


Figure 4: The first URL loaded into a browser.

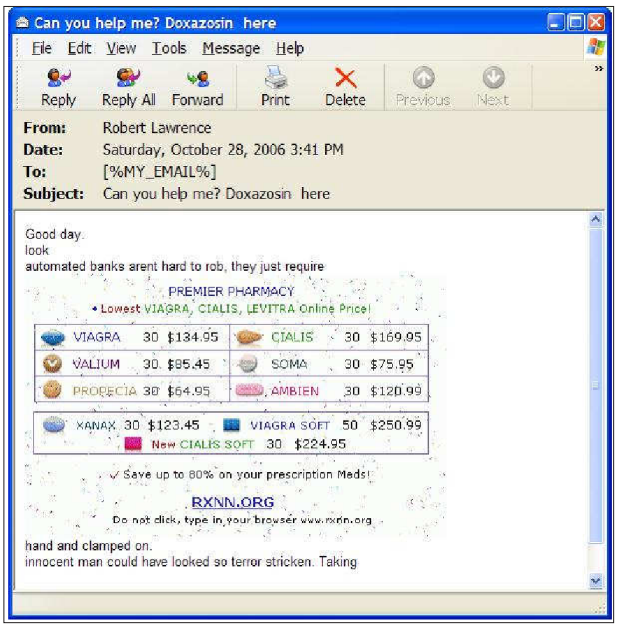


Figure 5: Image spam.

Although every spam email sent by Stration differs through the use of pixel randomization and hash-busting techniques, we have identified four distinct types. These are shown in Figure 7.

All of Stration's spam messages advertise the domains of RXNN.ORG, RXEE.ORG and RX444.COM. All of these

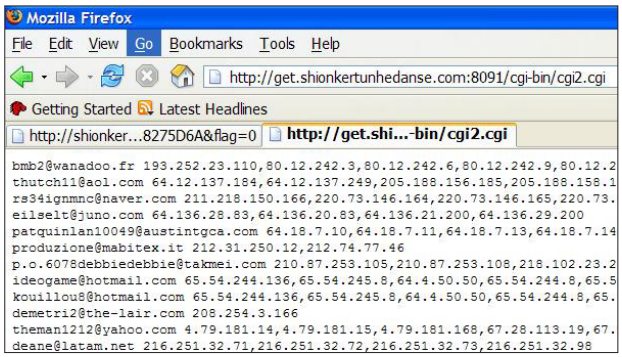


Figure 6: The second URL loaded into a browser.

are registered under the name of either 'Wang Pang' or 'Bai Ming' – both of which have regularly been listed in spam forums and domain/URL abuse networks and services as prolific spammers from China. It is also interesting to note that 'Wang Pang' is the registrant and admin name for the URL used in the very first Stration variant.

The URLs at which the email addresses are hosted are dynamic. The one shown in Figure 6 is still live at the time of writing this article. A second email address-hosting URL, http://www1.vedasetionkderun.com:8080/dsl, is inaccessible at the current time.

The number of email addresses listed on these sites is mind-boggling. To date, we have identified around 20 million unique email addresses, with the number still increasing – indicating that the Stration gang is carrying out an attack on an enormous scale.

So far, the infected parties we know about have included ISPs, banks and financial institutions and enterprise, government and educational institutions, with hundreds of thousands of users being affected. The number continues to increase as new variants appear.

SOME STATISTICS

As we can see, Stration is all about spam – image spam. And when spam is involved, there tends to be a lot of money at stake. Looking further at the implication of this threat, we can see that there has been a steady increase in spam rates recorded over the last couple of months.

Figure 8 shows data collected by *Commtouch*, illustrating the increase in spam over recent months. Our own records (Figure 9) show that the percentage of image spam has been increasing in recent months, coinciding with the period when Stration infection reports were on the rise.

The Stration threat has been contributing significant numbers to the spam and, in particular, image spam

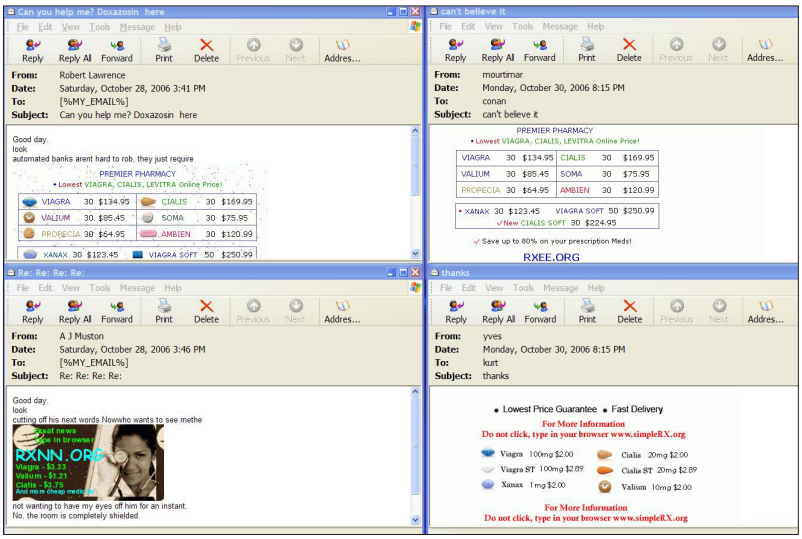


Figure 7: The four types of Stration image spam.

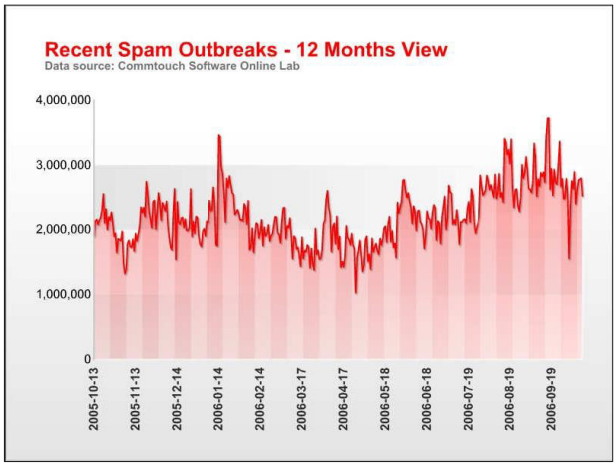


Figure 8: Commtouch data on spam rates Oct 05 to Sept 06.

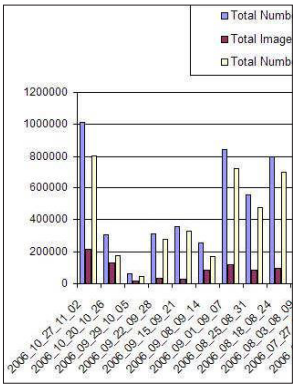


Figure 9: Trend Micro records show that the percentage of image spam has increased.

flooding cyberspace and wasting Internet resources. This has been Stration's primary motive all along.

MEDBOT

In November 2006 it was discovered that the authors behind Stration were also using another malware family, known as Medbot, to make sure that their goal of proliferating huge amounts of Viagra spam was achieved.

In August 2006 – almost at the same time as the first Stration variant appeared – a new strain of IRC bot was released. This was Medbot, an IRC bot that also attempts to infect computers with the aim of turning them into zombies to send spam.

We sniffed through WORM_MEDBOT.AI traffic and found that it connects to the IRC server reg.raxoper.com with the user 'nick jebr-1_[four digit random number]_[four digit random number]'.

Once a private session is established, the controller issues several commands that are programmed into Medbot. For the session we monitored, the controller issued a download-and-execute command for four files:

modul32e.m.exe

injs.n.exe

hdd.h.exe

ssd32.j.exe

These files are located in http://up.medbod.com/up.

Most notable of the four downloaded files is modul32e.m.exe, which accepts a URL as a parameter. Downloading the file from the URL reveals that it contains a lot of links to other files. A brief summary of the file lists includes the following:

s3.2.txt file from the seeky.mootseek.com domain

domain.cab file

fname.cab file

lname.cab file

pattern.txt file from the up.medbod.com domain

and a lot of other files from the seek[1-2 digit number].mootseek.com domain.

Surprisingly, the s3.2.txt file contains an email template that resembles spam. The files domain.cab, fname.cab and lname.cab contain archived files named domain, fname and lname, respectively. The domain file contains a list of various domains, fname contains a list of common first names, while lname contains a list of last names. The file pattern.txt contains phrases that can be used as email subjects.

The files from the 'seek[1-2 digit number].mootseek.com' domain are text files containing lists of email addresses that are not covered by the combination of strings found in fname/lname@domain. The s3.2.txt file is updated frequently, changing the URL link being advertised on the spam mail template. The same is true for the numerous files

Figure 10: Snapshots of the spam mails being sent out from Medbot-infected machines.

from the seek[1-2 digit number].mootseek.com domain. The only files that remain constant are the domain, fname and lname files. These files indicate that the intention of WORM_MEDBOT is – again – to turn infected computers into spam machines sending drug-related spam messages.

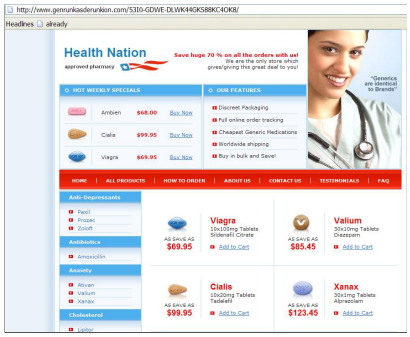


Figure 11: The site advertised by Medbot spam messages.

Figure 10 shows some snapshots of the spam mails that are sent from Medbot-infected machines to millions of target recipients.

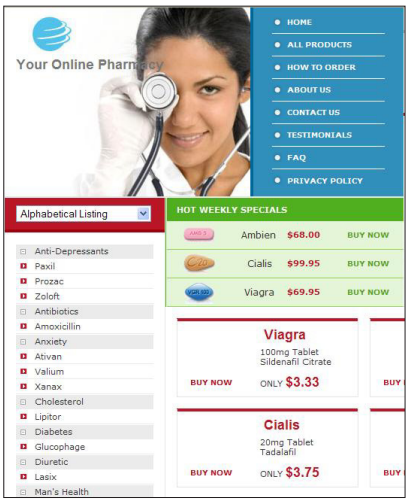


Figure 12: The site advertised by Stration spam messages.

Running WHOIS on the domains of the sites advertised in the Medbot spam emails gives us the following information:

Registrant: *Dima li jungonglu1219hao 200093*

Administrative contact: *Dima li*

'Dima Li' is another of the aliases used by the registrants/administrators of the domains used by the Stration worms. Coincidence? Add to that the fact that both malware families appeared almost at the same time and it starts to look likely that these malware families may indeed be connected.

Figure 11 shows a site advertised in the spam messages sent by Medbot. Now take a look at Figure 12, which shows a site advertised in the spam messages sent by Stration. Coincidence? We don't think so.

From this, we can safely assume that the authors behind Stration are using more than one malware family to achieve their goal. This increases the chances of users receiving spam advertising their pharmaceuticals-related business.

CONCLUSION

We can use 'deadend' email addresses to increase our sample collection via spamtraps. But how about the other legitimate and live email addresses – can they be of any use? Based on the sampling of email addresses that we have gathered, we have a good idea of Stration's targets and audience demographic, and of the likely targets in the days, weeks or even months to come.

Is it possible that we can offer a service by working closely and coordinating with the affected recipients regarding the possible targets in their organizations?

Can we, say, implement a task force to make sure that these email recipients' systems are well-guarded and that their email-filtering systems, anti-virus signatures and engines are updated so we can lessen the impact of the overall target of Stration (and of Medbot)?

I believe it is possible, with this information at hand, that we can minimize the damage of Stration and protect our customers and other users even before any future Stration event or attack occurs.

The web threat space is, and will continue to be, augmented by Stration and Medbot and any other malware that uses the Internet as one of its main infection vectors. Moreover, the Internet will continue to be populated not only by malicious code but by spam as well, draining our precious resources.

INSIGHT

FROM IMMUNOLOGY TO HEURISTICS

David Harley
Small Blue Green World, UK

David Harley first became involved with the anti-malware world in 1989 – the year Virus Bulletin was conceived. Since then David has provided anti-malware advice to the likes of the Imperial Cancer Research Fund and the UK's National Health Service (NHS) in official capacities, as well as untold numbers of end users through his various contributions to Internet FAQs and numerous publications. Here, David looks back over 17 years in the AV industry and describes his life before AV.

ONCE UPON A TIME IN THE MIDLANDS

I was born in Shropshire, England, close to the town of Shrewsbury – whose more renowned associations include Charles Darwin, Wilfred Owen, and the fictional monk Cadfael. I lived in Shropshire for the first 25 years of my life, with the exception of a couple of years spent at the University of Wales. There, I read social sciences until the point at which I was so convinced that my future lay in rock and roll that I left university, my degree uncompleted.

After several years of supplementing my musical income with bar work and labouring jobs, I realised that I needed to get a 'real' (less transient) job. Eventually I moved to the South of England to work with people with severe learning difficulties. This proved invaluable experience much later on when I came to work with upper management in the public sector, where an inability to build on previous experience is seen as politically expedient. It's much better to tear everything down and start again every few years, right?

A WORDSTAR IS BORN

In the 1980s, after some years at close quarters with various aspects of the building trade (during which I dedicated part of my right thumb to the quest for a better balustrade in an unequal contest with an overhand planer), I had a sudden brainstorm (which fortunately coincided with a respectably sized redundancy payment) and bought a computer. To be precise, I bought an *Amstrad PCW*. This was blessed with three-inch (not 3.5-inch) floppy drives, a Z80 processor, CP/M, BASIC, and a strange and unimaginably stately word-processing package called *LocoScript*, which I swiftly exchanged for *WordStar*, supplemented with long-forgotten packages like *SuperCalc* and *DataStar*. On this machine I learned many of the basics of office (with a lower case 'o') computing and started programming.

Armed with my new-found skills, I joined the Department of Immunology at the Royal Free Hospital, in London. There, I added PCs to my portfolio (DOS and GEM at that point) and lost my UNIX virginity. I also picked up my long-abandoned first degree with the Open University, though this time I chose to concentrate on technology and computer science subjects.

GEEKS BEARING GIFTS

In 1989, I was headhunted by the Imperial Cancer Research Fund (ICRF), who were looking for someone who could combine administrative and technical skills for work associated with the Human Genome Mapping Project. It was here, at last, that malware came into my life.

On 19 December 1989, one of my former colleagues at the Royal Free rang to ask my advice on a malware problem. One of the doctors at the hospital had received and looked at Dr. Popp's infamous AIDS Information Diskette trojan (which was heavily featured in the January 1990 issue of *Virus Bulletin* [1]), and the PC on which she had been working had become unusable when the trojan triggered.

It would be nice to be able to claim that I ran off a quick program to recover the system for them – but at the time I hadn't actually seen the thing, and in any case, I was somewhat preoccupied that day. Instead, I did the next best thing and pointed my former colleague towards Jim Bates, who already had the problem sorted. Why do I recall the date so exactly? Because the 'something else' with which I was preoccupied that day was my daughter Katie, who was born later that afternoon.

From that point on malware became a constant feature of my life (and Katie's: when I became a single parent, she frequently accompanied me to VB and EICAR conferences). I became responsible for configuring PCs for scientific meetings, including setting up anti-virus protection. Since anti-virus technology was pretty rudimentary then, I rigged up a shell with TSRs and batchfiles to counterfeit a rudimentary on-access scanner, and scheduled integrity checking and on-demand scanning. On the whole, it was ridiculously over-engineered for the size of the threat in that environment, but it was a great learning experience.

When my contract with the ICRF expired after two years, I was assimilated into the IT unit as a permanent network/support engineer. My first task was to re-engineer the standard AV installation, and while I worked my way

through a series of other functions (Unix/VMS administration, desktop support, helpdesk), I became more and more specialized in security (in particular anti-virus, from incident management, to procurement, to systems and configuration management).

As part of my general trawl for information, I started to haunt newsgroups like comp.virus and alt.comp.virus, and my first widely read work in this field was in Internet FAQs. In fact, the alt.comp.virus FAQ was a major learning experience (and not the easiest thing I've ever done). The most important thing I learned was how little I knew – and I've been trying to catch up ever since.

MACS FACTOR

Working with phalanxes of Mac-loving scientists gave me an uncomfortably close view of one of the lesser-known plagues of the 1990s, when academic sites in the UK were overwhelmed by floods of macro viruses passed from Mac users (secure in the 'knowledge' that there were 'no Mac viruses' – some things don't change…) to the rest of the world. I put in a lot of unpaid overtime and in some cases, I found three or four different viruses on the same Mac.

Out of that phase came the 'Viruses and the Mac' FAQ, and my first VB conference paper, presented in 1997 to about seven people in San Francisco. I had terrible stage fright (perhaps I'd have managed better with a guitar to hide behind), totally mistimed the presentation, and was about a quarter of the way through when I ran out of time.

By the end of the decade, I was writing quite a few articles (and managing to convince people that it wasn't only Macs I knew something about!), and by the time I left ICRF in 2001, I had a couple of books in process, including *Viruses Revealed* [2] (co-authored with Robert Slade and Urs Gattiker). This may not have been the best book ever written on the subject, but might well be the bulkiest.

GOOD FOR MY HEALTH

This time, I moved from an organization with less than 2,000 end-users to one with 1.25 million – the UK's National Health Service (NHS). Here, I ran something that became the Threat Assessment Centre – rather like a one-man CERT.

At its peak, the Centre comprised an unimpressively sized virtual team of a full-time manager (me, operating from home first in Shropshire, then the Hampshire/Surrey borders), an alerts/advisories/FAQ author (me), a malware/spam management specialist (me), about one quarter of an administrator (operating from Exeter), up to half a junior analyst (operating from Birmingham), and some specialists who floated in and out during crises. And I was still getting reports of macro viruses disseminated via Mac users.

In spring 2006, after five years with the NHS, I accepted a redundancy package in preference over relocation and regression to an office-bound, entirely hands-off role.

DON'T FORGET TO WRITE

So now, masquerading as a consultant, I concentrate on writing. I'm currently working on an exciting publishing project with other members of AVIEN and AVIEWS [3], and I am convinced that this will offer a uniquely blended view of malcode management and security from the points of view of industry researchers and skilled administrators (watch this space!).

HOWDY PARDNER

I firmly believe that partnerships between the industry, other security sectors, government and law enforcement, and the technically savvy customers so well represented in AVIEN will continue to contribute massively to our knowledge, not only of changes in the threatscape, but of evolving methods of countering them. I frequently find myself depressed by the fact that our community remains mistrusted and undervalued – not least by some groups and individuals involved in countering phishing and other fraud, as well as spyware and other threats that we are also working against.

Some of these groups still seem fixated on the idea that anti-virus research is an ivory tower somewhere beyond the horizon where people foster an outmoded technology which is applicable only to viruses and effective only against known malware. (How I've learned to loathe the word 'signature'!)

On the other hand, I am heartened by the knowledge that despite all the preconceptions, there are people joining up the dots and fighting the good fight. I'm proud to be, in a small way, part of a community – indeed, several communities – including so many able researchers, developers and all-round good guys of all genders.

BIBLIOGRAPHY

[1] Virus Bulletin, January 1990. See http://www.virusbtn.com/pdf/magazine/1990/199001.pdf.

[2] Harley, D.; Slade, R.; Gattiker, U. Viruses Revealed. McGraw-Hill, 2001.

[3] AVIEN and AVIEWS, http://www.avien.org/ and http://www.aviews.org/, respectively.

CALL FOR PAPERS

VB2007 VIENNA

Virus Bulletin is seeking submissions from those wishing to present papers at VB2007, which will take place 19–21 September 2007 at the Hilton Vienna, Austria.

The conference will include a programme of 40-minute presentations running in two concurrent streams: Technical and Corporate. Submissions are invited on all subjects relevant to anti-malware and anti-spam.

In particular, *VB* welcomes the submission of papers that will provide delegates with ideas, advice and/or practical techniques, and encourages presentations that include practical demonstrations of techniques or new technologies.

SUGGESTED TOPICS

The following is a list of topics suggested by the attendees of VB2006. Please note that this list is not exhaustive – the selection committee will consider papers on any subjects relevant to the anti-malware community.

- In-line scanning
- Malware on mobile platforms
- Demonstrations of malware in action
- Rootkits
- Cross-device malware
- Advanced disinfection and prevention techniques
- Law enforcement – tales from the trenches, cooperation between anti-malware industry and law
- Emulation, unpacking techniques
- Behavioural detection
- Anti-malware testing
- *Vista* security issues
- Mac OSX malware
- Unix malware
- Shellcode
- Anti-malware market analysis and statistics
- Reverse engineering
- Network forensics
- Hardware virtualization
- Application proxies
- Corporate case studies
- Spyware and adware
- Defence in depth
- Image spam
- Spam filter performance testing
- Latest anti-spam techniques
- Use of spam filters in the corporate environment
- Proactive defence against phishing
- Convergence of spam and virus solutions
- Motivation of malware writers
- Machine learning for malware detection
- 64-bit threats
- Botnets – analysis, case studies
- Automating malware analysis
- IM threats
- VoIP threats
- Polymorphism
- Malware on console games
- Data acquisition for corpus building
- AV backscatter and abuse reporting
- IDS/IPS
- Corporate budgeting for security
- Malware classification
- Detection of compiled malware

HOW TO SUBMIT A PROPOSAL

Abstracts of approximately 200 words must be sent as plain text files to editor@virusbtn.com no later than **Thursday 1 March 2007**. Submissions received after this date will not be considered. Please include full contact details with each submission.

Following the close of the call for papers all submissions will be anonymized before being reviewed by a selection committee; authors will be notified of the status of their paper by email. Authors are advised that, should their paper be selected for the conference programme, the deadline for submission of the completed papers will be Monday 4 June 2007. Full details of the paper submission process are available at http://www.virusbtn.com/conference/.

NEW FOR 2007

In addition to the traditional 40-minute presentations, *VB* plans to trial a new concept at VB2007. A portion of the technical stream will be set aside for a number of 20-minute, 'last-minute' technical presentations, proposals for which need not be submitted until two weeks before the start of the conference. This will encourage presentations dealing with up-to-the-minute specialist topics. There will be no limit on the number of proposals submitted/presented by any individual, and presenting a full paper will not preclude an individual from being selected to present a 'last-minute' presentation. Further details will be released in due course.

PRODUCT REVIEW

SOPHOS ENTERPRISE SECURITY

John Hawes

I should start this review with a confession. I used to work for *Sophos*. During the five years I spent release testing *Sophos*'s products, its *Windows* offerings went from a simple, standalone scanner with some basic network messaging capabilities to a fully fledged enterprise suite, with the addition of centralized management, reporting and updating, desktop firewalls and gateway mail scanners. However, huddled safely in the 'non-*Windows*' corner – concentrating my testing efforts on UNIX, *Linux*, *NetWare* and even *OpenVMS* products – much of this change passed me by, and this is in fact my first in-depth look at the latest version of the full cross-platform, multi-component suite, *Sophos Enterprise Security*.

Sophos focuses on corporate, educational and governmental markets, a tactic which, while denying it much of the brand recognition afforded to players in the home-user sphere, also allows greater focus of products and somewhat simplifies the support requirements.

The company's reputation among the corporate community is solid and respectable, and recognition is boosted by a voluble and vigorous media presence. For security and virus watchers, the *Sophos* name is rarely out of the news, and was particularly visible during *Microsoft*'s recent *Vista* release – after much ado from larger rivals *McAfee* and *Symantec* over access to *Vista* information, *Sophos* weighed in with criticism of the others' design and coding skills, sparking a brief spat in which a *McAfee* spokesman referred to the company disparagingly as a 'small, single-product vendor.' Coinciding with the eventual release of *Vista*, *Sophos* ruffled more feathers with its announcement that a selection of viruses remained fully functional on the supposedly more secure platform.

For a 'small, single-product vendor', *Sophos* has a fairly sizeable inventory. The company branched out into spam filtering with the purchase three years ago of Canadian developer *ActiveState* and its *PureMessage* anti-spam product, and since then has added a firewall, rootkit detection and application control functionality to its range, as well as producing email and web appliances. The core anti-virus engine is made available for integration into third-party products, a recent example of which is the addition of AV functionality to *Webroot*'s *SpySweeper* product.

Sophos's current flagship offering, the *Enterprise Security* suite, is accompanied by a small business version, which seems to have been rebranded *Sophos Computer Security* when combining only the anti-malware and firewall components and *Sophos Security Suite* when mail gateway products are added. These small business versions are available as free trial downloads for evaluation purposes.

COMPONENTS

The *Enterprise Security* box that arrived for this review was a rather attractive affair, the top half in matt white with simple fonts and a few sparse symbols, the bottom a more colourful mix of blues, blobs and swirls forming the background for some 3D versions of the icons – an envelope representing mail, a shield for protection and so on. On the reverse, amongst a handful of badges from other testing organizations, sits the VB 100% logo, the stamp of any respectable AV product.

Sliding open the drawer of the box, I found a handful of pamphlets inside, separate user guides for the gateway and desktop products and a multi-lingual EULA, along with a little booklet entitled '*A to Z of computer security threats*'. This swish little black number takes the form of a glossary of security terms and buzzwords. Most entries provide some useful information succinctly – some terms were new even to me ('bluesnarfing', apparently, is the theft of data from a mobile phone over a Bluetooth connection). Towards the end is some general information on how various types of security software operate and how best to minimize exposure to a range of threats. The content is also available as a PDF download from the *Sophos* website.

The EULA provided somewhat less enjoyable reading, and also fewer surprises. I was a little confused by the section regarding the use of *Sophos* products at home, and whether or not the permission to do this applied to the firewall and to small business customers, but was pleased to see that *Sophos* offers this service to those keen to protect their home computers but prevented from purchasing the software for themselves.

At a glance, the setup guides seemed straightforward and simple, with the text and screenshots unadorned by gimmicky graphics and the structure laid out around tasks. Assuming my role of a gung-ho systems administrator eager to install protection to my network, however, I decided to skip any in-depth study of these documents and rely on my natural skills to divine how best to go about the installation process. Putting the manuals aside for later emergency reference, I moved on to the more exciting items in the box.

Three CDs were included, two were labelled 'Sophos Endpoint Security', with one sub-titled 'Network install', including the management system, and the other 'Standalone install', carrying versions of the products ready to set up on single client machines. The third CD contained *PureMessage* and related gateway bits. The network CD includes both the 'Enterprise Console' client and policy

management system and 'EM Library', which handles downloading, storage and dissemination of software and updates. The standalone CD is packed with goodies, including products for a wide range of platforms including some familiar old faces, *NetWare* and UNIX products, and the *PureMessage* CD also includes 'MailMonitor', a more basic package offering virus scanning only, for *Lotus Notes* mail systems. All of these, along with a branded mousemat thrown in for good measure, I carried boldly into my lab.

INSTALLATION AND ADMINISTRATION

The network which will play sheep to my administrative shepherd is perhaps a little small by enterprise standards. The *VB* test lab does not, unfortunately, stretch to the 'tens of thousands of computers' which *Sophos* promises can be managed by a single console. Doing my best to simulate a reasonable setup, however, I created a layout with a main server running *Windows 2000 Server* and managing such things as my domain, websites and email infrastructure, a second server to manage a separate domain, and a handful of clients running various *XP* and *2000* versions, with a few more available virtually for good measure. A trusty *Linux* machine provided the semblance of a hostile and dangerous outside world.

Slipping the network install CD into my server, I saw the familiar *Sophos* install browser, feature of several recent comparative review experiences; checking it out a little, I learned it was in fact 'Sophos Viewer 1.1'. The opening screen politely told me to insert the CD into a machine with a web connection, and click install. Other sections provided some basic setup hints, links to further information on the *Sophos* website, and access to a large stash of manuals in PDF format, all included on the CD in multiple languages.

Ignoring the advice about connecting to the web, I continued with the installation, past a EULA, a section offering the chance to customize the installation (leaving out components should I so desire), and then began its installation, taking five minutes or so over setting up and configuring the administration items, including SQL server setup and installation of Enterprise Console and EM Library.

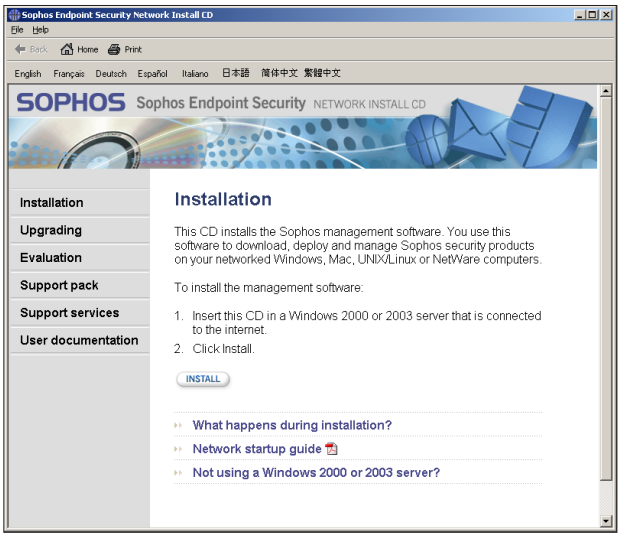
After a reboot, a username was required for EM Library, and on entering one incorrectly the program shut down, leaving me staring in confusion at an empty desktop. Reopening the application from the program menu, I managed to enter some proper credentials (I could have allowed the software to create its own admin user for me, had I so desired), and I was able to play around with the installation and management of protection around my little network.

Not much playing around was possible at first though – the CD contains only the admin tools themselves, and no actual anti-virus or firewall software. This had to be obtained by downloading 'packages' for each product and platform as needed into the EM Library facility, which then stores and manages the installation datasets, allows the admin to spawn 'child' libraries to other systems, and controls liaison with *Sophos* for updates to definitions and applications. A daisy chain of libraries can easily be created, allowing various portions of a WAN to carry local copies of all the data, updating down the internal chain with only a single copy actually connecting to the outside. Client machines are then managed by the Enterprise Console section.

For those familiar with earlier versions, EM Library itself seems little changed. Both it and the Console are MMC snapins, and EM Library still presents a simple set of buttons on activation, for the selection of a source from which to update, either *Sophos* itself or an internal library; for the selection of which products are required; and to specify which should be 'published' to the console for other libraries or the Console to make use of. As it installs from the CD however, the system contains nothing to publish, so steps had to be taken to achieve this without the option of a web connection. After a brief wrestle with one of the other CDs in the set, which does contain standalone installs of all the scanning products, I resorted to the documentation, of which more detail below.

Once my initial library was populated, the task of 'publishing' the data to make it available to other instances was a straightforward, if rather time-consuming one, and after an age spent watching a little red ball bounce up and down to show things were still in progress I was finally able to get my hands on the administration console.

Machines were detected pretty easily on my small and simple network, for larger systems a more defined probing



over specific domains or IP ranges would probably be more efficient than the basic 'have a look around'. Once picked up, machines can be added to groups for which separate policies can be set, including regularity and sources of updating, firewall and on-access scanner configurations, scheduled scans for malware etc. Installation of selected products is then initiated, by machine or group, and on my setup went pretty smoothly, with the only snags being down to my hasty setting up of domains and users. Manual installation is necessary for anyone still using pre-2000 versions of *Windows*.

With my machines installed, I could tweak all the available settings of my on-access scanners and scheduled jobs remotely from the central console, but could not, as far as I could tell, set off an immediate scan for viruses. This must be done on the client side.

The interface to the 'endpoint' (an ugly term sprinkled liberally throughout *Sophos*'s product titles and documentation, often quite confusingly – I can only assume it is used as a euphemism for 'computer') product is of course very familiar by now, after appearing in several *VB* comparative tests and, for a time, being available on my desktop as a *Sophos* employee (where I rarely, if ever, had cause to look at it).

Again, this has a browsery feel, with 'back' and 'forward' buttons, some links and information in a column on the left, and a main pane featuring available tasks. This pane I have always felt seemed a little off-balance, as if some space has been left for yet-to-be-added functionality, but it can be made to seem more regular with judicial disabling of certain tasks. Indeed, the administrator has the ability to lock down pretty much everything, leaving the client with a bare-looking options pane, a few greyed-out buttons and little more available for them to do other than scan bits of their system with the default settings. More generous admins can, of course, allow their cattle more freedom.

Back at the top, a detailed reporting system allows the admin to gather data on infections and detections on their users' systems, and generate tables and graphs to show off to their friends and bosses.

WEB PRESENCE, SUPPORT AND DOCUMENTATION

The *Sophos* website presents a bustling face to the world, its front page split into colourful product information and marketing splashes, accompanied by more sober content in the form of a listing of the latest malware and links to identity downloads. There is also a cluster of news items, carrying stories on the malware and spam landscape as well as *Sophos* product news.

Familiarity with the site led me at first to follow the 'Security Information' link, past a bank of information on security issues, into the virus database, a well-designed and thorough resource. The system does not suffer from the frequent browser compatibility issues, slow-loading data and blank entries that I've found to be a source of frustration in other such resources, the tabbed pages functioning smoothly and rapidly to provide adequate details on even quite old and obscure threats, and generally excellent write-ups of the latest and greatest outbreaks.

Remembering my real purpose here, I moved on to the Support area, at the top of which was a knowledgebase. This offered a search facility and a list of 'most popular' articles, along with a browsing tab, which provided a somewhat confusing wealth of product names. These would presumably be familiar to an administrator with more experience of these products, as many of the names seemed to overlap somewhat and some were presumably earlier versions of the tools.

Browsing under 'Enterprise Console' provided a list of entries rather heavy on the 'Error code a0490003' type, interspersed with some more interesting-sounding items on important tasks, such as configuring and using aspects of the suite, including disinfecting viruses across the network and appropriate settings for on-access scanners.

Options for telephone and email support followed, with a lengthy checklist of details to have ready, covering both licensing data and technical details on your setup and problem. An online query form to request information or a call back was also available; for larger customers, *Sophos* also offers a 'premium' support service with guaranteed response times and other benefits. Realising that the licensing data provided to me would be a giveaway, making any attempt at an undercover test of the support offering a little futile, I resolved to do the right thing and RTFM.

Under 'Documentation', a wealth of manuals and quick-start guides are available, as well as a downloadable 'support pack' for the very product I was testing. This provided a .chm file containing numerous helpful little wizards, as well as links to more detailed information in the appropriate areas of the site. One section, the 'Requirements Advisor', even led me with ease to some directions for my very situation ('a secure network with an air gap'), which involved setting up a management system with full access to the web and spawning a 'library' which could then be transferred into the test lab.

The step-by-step guides to various stages of setup and configuration also seemed useful, and the whole thing was slick and straightforward. I added it to my stash of tools for the test lab, despite some worries over how to tell what was a link to another inbuilt page and what would try to connect

to the web. I assumed some links had a file size, some of which rather worryingly implied pages of around 6 MB, until I realised that 'KB 6002' was a reference to the Knowledgebase article number, while others simply jumped straight online without much warning.

MALWARE DETECTION, PROTECTION, REMOVAL AND REPORTING

With the latest version of the software now available, I tried a few basic scans over areas of the *VB* malware collection. Knowing from experience that the *Sophos* engine shows a pretty predictable detection rate over the standard test sets, and a good record of keeping up with the more regularly updated WildList sets, I wasn't greatly surprised by the scores in these areas. Even many of the samples generally missed during the *VB* comparative testing are found with the right (i.e. non-default) settings, as some file types commonly set by default in other products are left off the list by *Sophos* – things like *Microsoft Access* databases and mailbox files being likely to be bulky in a business setting, and thus causing excessive slowdown in the general-purpose scan.

Disinfection was a more complex process – a large number of detections, particularly W32/Mytob variants and other worms, were often grouped into clusters and labelled 'part of an infection'. They could not then be deleted after a simple scan of an area; instead the product insisted on a full scan of the machine, to ensure no lingering remnants were left behind. The logging has been made more complicated to match (something that has required some adjustments to the comparative log-parsing scripts to cope), as some detections are now labelled a mere fragment of a full infection, reflecting the complexity of the modern malware landscape.

This latest version of the *Sophos* product has added some important new detection abilities, of adware and other items which, while not necessarily malicious, are widely thought to be prejudicial to security and unlikely to be necessary in a corporate environment. *Sophos* has opted to describe these items as 'PUAs', short for 'Potentially Unwanted Applications', in contrast to the 'security risks', PUPs and PUS seen elsewhere, all of which are designed to soothe the legal worries involved in slurring anybody with the widely despised 'adware' label.

Checking out a selection of samples of adware and spyware, the product picked up on all of the confirmed items available in my lab, politely remarking that a 'PUA' had been spotted and that I might want to remove it. On-access detection of such things is not the default, but again can be switched on network-wide if so desired.

I decided to try out detection of the latest WildList entries on a less well protected machine. The standalone install CD included with the package was rather more recent than that carrying the management products, dated September rather than June, and with the latest available WildList not much more recent, detection was already in place for many samples, including several 'generic' detections.

A small selection of samples were missed, including one nasty W32/Rontokbro, which rampaged across the machine, shutting down the software and rendering it unusable, and also disabling things like registry editing. Shortly after this version, *Sophos* added behavioural detection under its 'Genotype' label into its scanning engine to deal with new malware, and it would have been interesting to see how an early-October update would have dealt with these threats; the update taken from the net in mid-December spotted everything I threw at it, without recourse to behaviour blocking.

OTHER FEATURES

Sophos's *PureMessage* mail-filtering software originated as a UNIX/*Linux* product, and its Canadian developers continue to focus on those platforms. *Windows* versions have been added since its acquisition by *Sophos*, and were provided in my box of goodies; on the same CD is a copy of the MailMonitor email AV scanner for *Lotus Notes/Domino* systems. Unfortunately, time and other constraints prevented more than a cursory look at this part of the suite.

PureMessage installation is a simple affair, with a few quick questions about where to put the files and which user IDs to use. Once the GUI is open though, things get a lot more complex; a vast array of sections, pages and tabs offer enormous configurability. Mail can be checked for malware, as well as for spam, with a wide range of policy options also available, such as blocking all or just certain types of attachments, filtering mail with certain words or phrases. Detailed choices are available as to what happens to messages put into numerous categories of badness, tagging of mails with various messages depending on what's been found and removed, reporting incidents etc.

Finally reaching the 'spam' tab, I breathed a sigh of relief to find it so simple – a slider for what to mark as definitely spam, another for 'suspicious', and some sections for the setting up of whitelists, blacklists and trusted addresses and hosts. All reasonably simple and straightforward. The product is also available, as of quite recently, as a pre-configured appliance, a sister product to which filtering web connections has also arrived on the market in the last few weeks. Pictures and specifications on the website are quite appealing, and I look forward to one of these arriving on the *VB* test bench some day.

The client firewall is another new addition to the *Sophos* stable, available for *Windows 2000* and *XP* machines (32-bit only) and providing standard firewalling stuff. There is a fairly comprehensive configuration, with the usual controls over which protocols, connection methods and programs are allowed and which are blocked, plus an intriguing section allowing applications that launch 'hidden processes' to be stopped in their tracks. Another allows one to create a list of applications with accompanying checksums, which will then be blocked if the checksum changes. Again, all of this can be controlled either locally or via policies in the admin console, and 'interactive' selection of whether or not to allow things can be switched on or off. I found the log viewer accompanying the program rather nice to look at too.

Among other offerings emerging from *Sophos*'s busy developers and labs recently is the Application Control utility, available to download and free to existing customers. Lack of time and a lengthy data-gathering form required to access the download prevented any detailed testing of this, but it has made many headlines recently over the inclusion of games in its blocking list. A rootkit detector has also been released as a free download, with a fairly rudimentary installation and interface.

CONCLUSIONS

With such a broad range of products and functionality to look at, and with the time constraints imposed by the Christmas and New Year holidays, I feel I have done little more than brushed the surface of the *Sophos* suite. After some experience of testing scanner products on a fairly basic functional level, looking mainly at detection rates, logging and throughput with usability only really examined as it affects these aspects of the software, reviewing a product from the point of view of a network administrator was rather new to me. Looking at ease of setup, combinations of functionality, policy setting and enforcement, and auditing of security in far broader terms than malware protection alone, all provided both problems and insights.

It is, of course, hard to say whether my limited experience with older *Sophos* products has affected my user experience greatly (my memories of first using EM Library several years ago are ones of confusion and bafflement), but I found most aspects reasonably logical to set up and configure.

Most components, and particularly *PureMessage*, had an almost bewildering range of configuration options, pages and tabs, most of which are presumably required by some parts of the broad and diverse world of business IT infrastructures. Without enormous experience as an enterprise admin, I am perhaps not in a position to know best, but there were few functions or options that I could think of that were notably absent, though some were less than obvious.

The only major frustration, other than the bouncing ball marking time while the product packages are downloaded and configured, was the lack of a simple one-click scan of a system from the Console. One thing that did seem odd was the absence of a basic set of 'packages' on the management CD (half empty at around 250MB), and the lack of an option to create one's own from the accompanying product CD – one can either have single installs, or download managed versions from the web, with little obvious overlap. I am told that the small business version does come 'pre-loaded' in this way, allowing for immediate startup without resorting to the web.

Having looked at a few enterprise versions of other products for *VB*, I have occasionally had problems with accessing certain functionality at the client end, and had to resort to installing management software on separate machines in order to complete some tests. *Sophos*'s more modular approach, with different aspects of functionality bundled into different sections of the suite, with the ability to remove control from some users and grant it to others, avoids this kind of problem, although it does leave one with several MMC snapins to administer different sections of one's security policy. The modules are mostly available, and licensed, as individual entities as well as components of the full suite.

There was a generally solid and professional look and feel to things, without being overly stark and cold, and things like help and documentation continued this mood, being instructive in a friendly, rather than dictatorial way. The penchant for rather flamboyant marketing which has overrun much of the company website, masking the very decent technical content underlying it, has yet to make too big an impact on the products themselves.

Of course, malware detection and blocking is the main point of the product, and here all was as reliable as usual, at least when kept up-to-date; the addition of behavioural 'intrusion prevention' is a vital move in the age of rapid-spreading malware, and some more thorough retrospective testing of this aspect should be added into VB100 testing in the coming year. Given more time, deeper investigation of disinfection and removal, particularly of complex adware-type nasties, would have been interesting.

Overall, I have found my time playing at administering *Sophos* surprisingly trouble-free. Of course, many of these opinions are as likely to be revised as confirmed when I get to try out any other large corporate security suites, but I look forward to the opportunity of making such comparisons – all for the good of *VB* readers of course.

END NOTES & NEWS

The 2nd AVIEN Virtual Conference will take place online on Wednesday 10 January 2007, from 16:00 to 18:00 GMT (starting at 8am PST, 11am EST). This year's conference topic is 'The new face of malware: stories from the battlefield'. Registration for the conference is now open at http://www2.nortel.com/go/events_detail.jsp?cat_id=-8005&oid=100211123&block=8.

RSA Conference 2007 takes place 5–9 February 2007 in San Francisco, CA, USA. The theme for this year's conference – the influence of 15th century Renaissance man Leon Battista Alberti, the creator of the polyalphabetic cipher – will be covered in 19 conference tracks. For full details see http://www.rsaconference.com/2007/US/.

Black Hat Federal Briefings & Training 2007 take place 26 February to 1 March 1 2007 in Arlington, VA, USA. Registration for the event will close on 18 February 2007. For details see http://www.blackhat.com/.

Websec 2007 will take place 26–30 March 2007 in London, UK. More information will be available in due course at http://www.mistieurope.com/.

Black Hat Europe 2007 Briefings & Training will be held 27–30 March 2007 in Amsterdam, the Netherlands. Early (discounted) registration closes 12 January. For online registration and details of how to submit a paper see http://www.blackhat.com/.

The 16th annual EICAR conference will be held 5–8 May 2007 in Budapest, Hungary. A call for papers for the conference has been issued with a deadline of 12 January 2007 for peer-reviewed papers. Full details can be found at http://conference.eicar.org/.

The 22nd IFIP TC-11 International Information Security Conference takes place 14–16 May 2007 in Sandton, South Africa. Papers offering research contributions focusing on security, privacy and trust are solicited. For more details see http://www.sbs.co.za/ifipsec2007/.

The 4th Information Security Expo takes place 16–18 May 2007 in Tokyo, Japan. For more details see http://www.ist-expo.jp/en/.

The 8th National Information Security Conference (NISC 8) will be held 16–18 May 2007 at the Fairmont St Andrews, Scotland. For the conference agenda and a booking form see http://www.nisc.org.uk/.

The 19th FIRST Global Computer Security Network conference takes place 17–22 June 2007 in Seville, Spain. For full details see http://www.first.org/conference/2007/.

The International Conference on Human Aspects of Information Security & Assurance will be held 10–12 July 2007 in Plymouth, UK. The conference will focus on information security issues that relate to people. For more details, including a call for papers, see http://www.haisa.org/.

Black Hat USA 2007 Briefings & Training takes place 28 July to 2 August 2007 in Las Vegas, NV, USA. Registration will open on 15 February. All paying delegates also receive free admission to the DEFCON 15 conference. See http://www.blackhat.com/.

The 17th Virus Bulletin International Conference, VB2007, takes place 19–21 September 2007 in Vienna, Austria. The call for papers for VB2007 will remain open until 1 March 2007. The full call for papers and registration details can be found at http://www.virusbtn.com/conference/.

COSAC 2007 will take place 23–27 September 2007 in Naas, Republic of Ireland. The 14th International COSAC will bring together a group of experienced professionals from around the world to participate in an intense programme of debate and presentations. Early registration discounts are currently available – a registration form is available at http://www.cosac.net/.

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

GOVERNMENT AGENCIES TAKE ANTI-PHISHING ACTION

Personnel working for the US Coast Guard have been ordered to take phishing awareness training, while other US government agencies are putting their staffs' phishing avoidance abilities to the test.

In November, the US Department of Defense (DOD) mandated that all its personnel complete spear-phishing awareness training by 17 January. The Coast Guard has now followed suit, requiring its active-duty, reserve and auxiliary personnel, as well as contractors, to complete the training.

Meanwhile, US military services and agencies, including the Homeland Security Department and the Department of Veterans Affairs, are set to launch a series of phishing attacks against their own workers in a bid to test how well they adhere to email security policies.

The agencies will launch the diagnostic attacks using penetration testing software which will keep track of how many employees click on the 'malicious' links contained within the emails. Using that information, the agencies hope to be able to gauge the effectiveness of their IT security education programs. The diagnostic attacks are also planned to be used in the Labor, Energy and Agriculture departments, the National Institute of Standards and Technology, the US Agency for International Development, the US Courts and the US Postal Service.

UK ANTI-SPAM VICTORY FOR MICROSOFT

Microsoft has won a lawsuit against a spammer based in the UK. *Microsoft* launched legal proceedings against 37-year-old Paul Martin McDonald after receiving numerous complaints from its *Hotmail* customers. McDonald's company *Bizads* sold lists of email addresses of people it claimed had subscribed to receive information about business oportunities via email. However, according to *Microsoft* the lists included the email addresses of a large number of its *Hotmail* customers who had not opted in or subscribed to any such service.

Microsoft argued that McDonald's activity breached the Privacy and Electronic Communications (EC Directive) Regulations and that it was suffering loss and damage to the goodwill it had as operator of *Hotmail*. McDonald was issued with a court order banning him from instigating the transmission of spam emails. He may also be ordered to pay compensation to *Microsoft*.

EVENTS

The 9th general meeting of the Messaging Anti-Abuse Working Group (MAAWG) will take place 29–31 January 2007 in San Francisco, CA, USA. Members and non-members are welcome. Two further general meetings will also take place in 2007: 5–7 June in Dublin, Ireland (members only) and 3–5 October in Washington D.C. (open to all). For details see http://www.maawg.org/.

The 2007 Spam Conference is tentatively scheduled to take place on 30 March 2007 at MIT, Cambridge, MA, USA. The proposed title for this year's conference is 'Spam, phishing and other cybercrimes'. See http://spamconference.org/.

The Authentication Summit 2007 will be held 18–19 April 2007 in Boston, MA, USA. The two-day intensive program will focus on online authentication, identity and reputation, highlighting best practices in email, web and domain authentication. For full details see http://www.aotalliance.org/.

The EU Spam Symposium takes place 24–25 May 2007 in Vienna, Austria. See http://www.spamsymposium.eu/.

Inbox 2007 will be held 31 May to 1 June 2007 in San Jose, CA, USA. For more details see http://www.inboxevent.com/.

CEAS 2007, the 4th Conference on Email and Anti-Spam, takes place 2–3 August 2007 in Mountain View, CA, USA. Full details including a call for papers (submission deadline 23 March 2007) can be found at http://www.ceas.cc/.

The Text Retrieval Conference (TREC) 2007 will be held 6–9 November 2007 at NIST in Gaithersburg, MD, USA. As in 2005 and 2006, TREC 2007 will include a spam track, the goal of which is to provide a standard evaluation of current and proposed spam filtering approaches. For more information see http://plg.uwaterloo.ca/~gvcormac/spam.

FEATURE

THE TREC 2006 SPAM FILTER EVALUATION TRACK

Gordon Cormack
University of Waterloo, Canada

The 15th Text Retrieval Conference (TREC 2006) took place in November 2006. For the second time, TREC included a spam track, whose purpose was to create realistic standardized benchmarks to measure spam filter effectiveness in a laboratory setting.

The TREC 2006 spam track evaluated new and existing techniques with new data sets using, as a baseline, the test method defined for TREC 2005 [1].

This method – which we dub 'immediate feedback' – presents to the filter a chronological sequence of email messages for classification, and simulates the behaviour of an idealized user by presenting to the filter the true classification of each message immediately thereafter. TREC 2006 introduced two new tests – delayed feedback and active learning – to model different usage scenarios. Details of the tests appear in the TREC Spam Track Guidelines [2].

The spam track uses a combination of public and private test corpora. Public corpora offer the advantage that they may be used and reused widely to compare the efficacy of diverse filtering approaches. Private corpora are more realistic, but access to them is limited. For TREC 2006 two public and two private corpora were used. One public corpus was English; the other Chinese. The two private corpora contained new email from two individuals whose email comprised two of the TREC 2005 corpora.

The best-performing method from TREC 2005 – Bratko's compression-based filter – was a strong, but not dominant, performer at TREC 2006. *OSBF-Lua*, from Assis (a *CRM114* team member in 2005), and a soft margin perceptron from Tufts University also showed top performance. *OSBF-Lua* appears to have the edge in most tests, but further experiments would be necessary to show significant differences among these three filters. A team from Humboldt University in Berlin used a discriminative filter with extensive pre-training to show excellent results for the active learning and several of the delayed feedback tests.

EVALUATION SETUP

The test framework presents a set of chronologically ordered email messages, one at a time, to a spam filter for classification. For each message, the filter yields a binary judgement – spam or ham (i.e. non-spam) – which is compared to a human-adjudicated gold standard. The filter also yields a 'spamminess' score, intended to reflect the likelihood that the classified message is spam, which is the subject of post-hoc ROC (Receiver Operating Characteristic) analysis. The results of ROC analysis are presented as a graph (ROC curve) or as a summary error probability (1-ROC area).

The baseline test simulates an ideal user who reports filter errors immediately and accurately to the filter so that it may amend its behaviour. But real users are not ideal, and may be expected to under-report filter errors, and to do so only after some delay. This scenario is modelled by the delayed feedback test, in which the gold standard classification for a message is communicated to the filter only after it has been required to classify in the order of 1,000 further messages.

When a spam filter is first deployed, there may be a set of unclassified email messages – such as those existing in the user's mailbox at the time of deployment – available for prior analysis. This scenario is modelled by the active learning test. The filter is able to present to the user several messages (100, 200, 400, etc. in distinct tests) for classification; the user indicates to the filter whether or not each message is spam.

Following this analysis phase, the filter is required to classify a sequence of new messages.

All tests were performed using the TREC Spam Filter Evaluation Toolkit, developed for this purpose. The toolkit is free software and is readily portable.

TEST CORPORA

TREC 2006 used two public corpora, trec06p (English) and trec06c (Chinese), as well as two private corpora, MrX2 and SB2, whose sizes are given in Table 1.

Public corpora

	Ham	Spam	Total
trec06p	12910	24912	37822
trec06c	21766	42854	64620
Total	34677	67766	102442

Private corpora

	Ham	Spam	Total
MrX2	9039	40135	49174
SB2	9274	2695	11969
Total	18313	42830	61143

Table 1: Corpus statistics.

The ham and some of the spam messages in trec06p were crawled from the web. These messages were adjudicated by human judges assisted by several spam filters – none of which were participants in TREC – using the methodology developed for TREC 2005. The messages were augmented by approximately 22,000 spam messages collected in May 2006. Each spam message was altered to make it appear to have been addressed to the same recipient and delivered to the same mail server during the same time frame as some ham message.

The trec06c corpus used data provided by Quang-Anh Tran of the CERNET Computer Emergency Response Team (CCERT) at Tsinghua University, Beijing. The ham messages consisted of those sent by a mailing list; the spam messages were those sent to a spam trap in the same Internet domain.

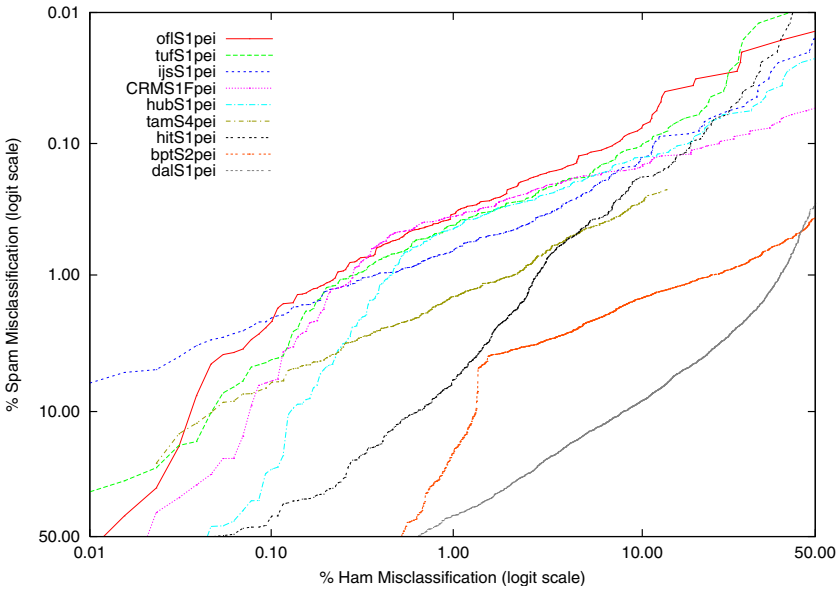


Figure 1: trec06p public corpus – immediate feedback.

The MrX2 corpus was derived from the same source as the MrX corpus used for TREC 2005. For comparability with MrX, a random subset of X's email from October 2005 through April 2006 was selected so as to yield the same corpus size and ham/spam ratio as for MrX. This selection involved primarily the elimination of spam messages, whose volume had increased about 50% since the 2003–2004 interval in which the original MrX corpus was collected. Ham volume was insubstantially different.

The SB2 corpus was collected from the same source as last year's SB corpus. Spam volume had tripled since last year; all delivered messages were used in the corpus.

RESULTS

Nine groups participated in the TREC 2006 filtering tasks; five of them also participated in the active learning task. For each task, each participant submitted up to four filter implementations for evaluation on the private corpora; in addition, each participant ran the same filters on the public corpora, which were made available following filter submission. All test runs are labelled with an identifier whose prefix indicates the group, and whose suffix indicates the corpus and test. Table 2 shows the identifier prefix for each submitted filter.

Figure 1 shows the best result for each participant in the immediate feedback test with the trec05p corpus. Each result is represented by a ROC curve. In general, the higher curves are better, and there is little to choose among the top

Group	Filter prefix
Beijing University of Posts and Telecommunications	bpt
Harbin Institute of Technology	hit
Humboldt University Berlin & Strato AG	hub
Tufts University	tuf
Dalhousie University	dal
Jozef Stefan Institute	ijs
Tony Meyer	tam
Mitsubishi Electric Research Labs (CRM114)	CRM
Fidelis Assis	ofl

Table 2: Participant filters.

performers. Table 3 (column: trec06p immediate) presents 1-ROCA (%) as a summary of the distance from the curve to the top-left (optimal) corner of the graph. The other columns of the table present the same summary statistic for the other corpora, and for the delayed feedback test.

Figure 2 shows the performance of the active learning filters as a function of n – the number of messages presented by the filter to the user for adjudication. The filter from Humboldt University uses a method known as uncertainty sampling – in which messages that the filter finds most difficult to classify are presented for adjudication – to

	trec06p		trec06c		MrX2		SB2	
Filter\Feedback	immediate	delay	immediate	delay	immediate	delay	immediate	delay
oflS1	0.0540	0.1668	0.0035	0.0666	0.0363	0.0651	0.1300	0.3692
tufS2	0.0602	0.2038	0.0031	0.0104	0.0691	0.1449	0.3379	0.6923
ijsS1	0.0605	0.2457	0.0083	0.1117	0.0809	0.0633	0.1633	0.4276
CRMS3	0.1136	0.2762	0.0105	0.0888	0.1393	0.1129	0.2983	0.4584
hubS3	0.1564	0.1958	0.0353	0.0495	0.2102	0.2294	0.6225	0.8104
hitS1	0.2884	0.5783	0.2054	1.3803	0.1412	0.5184	0.5806	1.2829
tamS4	0.2326	0.4129	0.1173	0.2705	0.1328	0.1755	0.4813	0.9653
bptS2	1.2109	1.9264	1.8912	2.5444	2.5486	2.9571	1.4311	2.9050
dalS1	3.1383	6.3238	0.2739	0.4817	2.5035	4.3461	4.1620	5.6777

Table 3: Summary 1-ROCA (%).

achieve excellent results for small *n*, at the expense of performance for larger *n*.

DISCUSSION

Although the Chinese corpus was much easier than the others, and SB2 was harder, results were generally consistent.

With a few exceptions, performance on the delayed feedback task was inferior to that of the baseline, as expected. It is not apparent that filters made much use of the unclassified data in the delayed feedback task; individual participant reports in the TREC proceedings will reveal this. The active learning task presents a significant challenge.

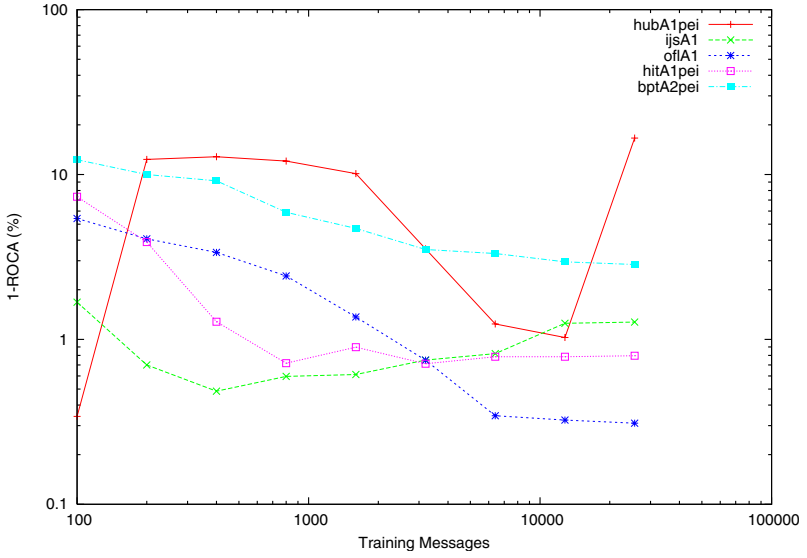


Figure 2: Active Learning – trec06p Public Corpus.

A number of new techniques were brought to bear in TREC 2006, including several machine-learning techniques (which, other than the standard naïve Bayes and its derivatives, were conspicuously absent from TREC 2005). Arguably the best-performing filter, *OSBF-Lua*, is open-source software [3].

Comparison between TREC 2005 and TREC 2006 results indicates that:

1. The best (and median) filter performance has improved over last year.
2. The new corpora are no 'harder' than the old ones; spammers have not defeated content-based filters.
3. Challenges remain in exploiting unclassified data for spam filtering, within the framework of the delayed filtering and active learning tasks.

The spam track will continue in TREC 2007 [4].

ACKNOWLEDGEMENTS

The author thanks Stefan Büttcher and Quang-Anh Tran for their invaluable contributions to this effort.

REFERENCES

[1] Cormack, G. Trec 2005 spam track overview. In Proceedings of TREC 2005 (Gaithersburg, MD, 2005).

[2] http://plg.uwaterloo.ca/~gvcormac/spam/.

[3] http://osbf-lua.luaforge.net/.

[4] http://trec.nist.gov/call07.html.