SEPTEMBER 2005

virus BULLETIN

The International Publication on Computer Virus Prevention, Recognition and Removal

CONTENTS

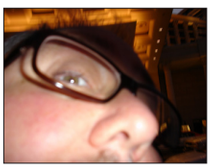
ISSN 0956-9979

IN THIS ISSUE

A NEW BREED

In the last year or two, an increasing number of *Symbian* threats have been reported. While there are not yet many malware writers who are interested in the *Symbian* OS, this may soon change. Robert Wang asks: is the *Symbian* OS in danger of further attacks?
page 6

A RIGHT PAIR



Although one always hears about 'Black Hat *and* DEFCON', they are in fact two very different events. *VB*'s intrepid reporter (aka AV industry miscreant) has a report on each.
page 11

A NEW NAMING INITIATIVE

The Common Malware Enumeration (CME) initiative is a new effort headed by the US-CERT, which aims to match a unique identifier to each threat. Jimmy Kuo and Desiree Beck explain how it is hoped this initiative will help alleviate the 'virus-naming mess'.
page 14

vbSpam supplement

This month: anti-spam news and events, and Sorin Mustaca provides *VB*'s first phishing analysis.

'Support for 16-bit software had been excluded deliberately ... I call this lack of backward compatibility!

Peter Morley, McAfee, UK

WHAT'S COMING? WINDOWS XP 64-BIT

This article was prompted by the arrival of *Windows XP* 64-bit, early in June 2005. I decided to install the new operating system, as normal, and assumed that I would be able to use it in the way in which I was accustomed. How wrong can one be?

I installed *Windows XP* 64-bit on a recent machine, with a new motherboard (ASUS A8N-SL1 with 64-bit *AMD* processor), and a new 30 Gb hard disk. Since the system was already running old-fashioned *Windows XP*, this was achieved merely by changing the HD and proceeding with the installation.

This proved to be the most difficult installation of an operating system I have encountered, and I suspect that many other people would give up and forget it. The OS kept restarting with messages on blue screens, requiring minor changes.

In case you think the operator was the cause of the problem, I wasn't! A search on the Internet demonstrated that the issues I was experiencing had all been experienced by other unfortunates, some of whom *had* indeed given up trying. The cause of the problems may have been the fact that the machine's motherboard was designed after this operating system. Anyway, after a long battle, it did eventually install.

However, worse was to come. I copied onto the system all the programs and files that I use in my day-to-day work.

I could not process viruses, because four of the five tools I normally use would not run (*Q-EDIT*, *Hiew*, *Volkov Commander*, and *DV8*). The failure was designed to happen. Each of the programs produced a dialog box suggesting I that contact the supplier, and get an up-to-date version of the tool. I made a few enquiries, and established that support for 16-bit software had been excluded deliberately from *Windows XP* 64-bit. I call this lack of backward compatibility!

Microsoft has obviously been caned. The August 2005 issue of *Windows XP Magazine* (which arrived in early July) failed to mention *XP* 64-bit, the newest and latest version of the operating system.

The arrival of *Windows Media Centre* 64-bit is imminent. I believe this will be very similar to the version I had, but that it probably won't matter, for two reasons:

i) The operating system will already be installed on arrival.

ii) Most users will not be adding ancient 16-bit software.

What are the consequences of all this?

First, you can expect the following marketing point to be made quite strongly. Internet usage has been exploding for the last four years, and I believe it will continue to do so for the next 10 years. The use of 64-bit technology in both hardware and software will be essential.

No surprise, then, that an *Acer* advertisement in an early July 2005 edition of the *Financial Times* made exactly this point. All the advertisements you see pushing *Intel Centrino Mobile Technology*, may need to be reviewed.

Second, everyone expects a *Microsoft* anti-virus announcement imminently (Beta before the end of September 2005, and product(s) before the end of the year). You can be certain that *Microsoft*'s AV product will run perfectly well under *Windows XP* 64-bit. There is an implication here, that all other AV companies will need to scramble around to ensure that none of their products fail to run on the 64-bit operating system. There isn't much time!

One final question occurs to me: will *Longhorn* fail to support 16-bit programs? All my instincts suggest that *Microsoft* will include support, but let's wait and see!

NEWS

MORE HASH WOES

For the second year running, research presented at the annual Crypto conference has raised concern over the security of commonly-used hash functions. The encryption field was thrown into a frenzy in August 2004 when the security of hash functions MD5, SHA-0 and SHA-1 was called into question (see *VB*, September 2004, p.3 and October 2004, p.13). Last month, researchers revealed that they have discovered a new, faster attack against the SHA-1 hashing algorithm.

Xiaoyun Wang, one of the team of Chinese researchers that at last year's Crypto conference outlined methods of finding collisions in the MD4, MD5, HAVEL-128 and RIPEMD algorithms, has announced that the time complexity of a new attack her team has achieved against SHA-1 is 2^{63} (the team's previous result was 2^{69}; brute force is 2^{80}). It is also expected that this result will be improved upon over the next couple of months. Wang's paper can be found at http://www.infosec.sdu.edu.cn/paper/sha1-crypto-auth-new-2-yao.pdf.

In reaction to the findings, the National Institute of Standards and Technology (NIST) plans to host a two-day Cryptographic Hash Workshop on 31 October and 1 November 2005 to solicit public input on how best to respond to the current state of research in this area.

THE NAMING GAME

First there was *Kaspersky Lab* and *ICSA Labs*, then came along *MessageLabs* and *SophosLabs*. Now *CA* has become the latest AV firm to join the party with the announcement of its new division 'dedicated to promoting and performing advanced research in systems management and security for the enterprise', *CA Labs*. Presumably the inclusion of the word 'lab' or 'labs' in a company's name is intended to lend weight and signify the company/division's dedication to serious research. Unfortunately from an onlooker's point of view it seems more of a case of copycat tactics or else a serious lack of imagination. One wonders whether there is any better way of adding prestige to your AV company's name. Send your suggestions on a postcard ...

ADDENDUM: NETWARE 6.5 COMPARATIVE REVIEW

Unfortunately, due to a combination of miscommunication and missed communications, *Symantec AntiVirus* was not included in last month's *NetWare 6.5* comparative review. *VB* has since tested the product and is pleased to reveal that *Symantec AntiVirus 10.0.0.1* detected all samples in the wild, with no false positives, and is awarded a VB 100%.

Prevalence Table – July 2005

Virus	Type	Incidents	Reports
Win32/Netsky	File	44,177	43.28%
Win32/Mytob	File	36,438	35.70%
Win32/Mydoom	File	6,345	6.22%
Win32/Zafi	File	4,135	4.05%
Win32/Bagle	File	2,760	2.70%
Win32/Bagz	File	2,557	2.50%
Win32/Lovgate	File	1,949	1.91%
Win32/Sdbot	File	554	0.54%
Win32/Funlove	File	551	0.54%
Win32/Mabutu	File	332	0.33%
Win32/Klez	File	303	0.30%
Win32/Bugbear	File	248	0.24%
Win32/Dumaru	File	170	0.17%
Win32/Pate	File	144	0.14%
Win32/Mimail	File	140	0.14%
Win32/Swen	File	104	0.10%
Win32/Valla	File	104	0.10%
Win32/MyWife	File	96	0.09%
Win32/Fizzer	File	77	0.08%
Win32/Mota	File	70	0.07%
Redlof	Script	66	0.06%
Win32/Gibe	File	65	0.06%
Win95/Spaces	File	61	0.06%
Win32/Yaha	File	59	0.06%
Win32/Sober	File	57	0.06%
Win32/Reatle	File	38	0.04%
Win32/Wurmark	File	37	0.04%
Win32/Agobot	File	32	0.03%
Win32/SirCam	File	26	0.03%
Win32/Hybris	File	23	0.02%
Win32/Maslan	File	21	0.02%
Win32/Eyeveg	File	19	0.02%
Others[1]		339	0.33%
Total		102,078	100%

[1]The Prevalence Table includes a total of 339 reports across 61 further viruses. Readers are reminded that a complete listing is posted at http://www.virusbtn.com/Prevalence/.

FEATURE 1

THE TROUBLE WITH ROOTKITS

Patrick Runald
F-Secure, UK

Rootkits are a fast-emerging security threat which can hide malware from conventional security tools. So how do they do this, and what can you do about them?

WHAT IS A ROOTKIT?

Powerful *Windows* rootkits are a potential problem for PC users in the future. Rootkits can hide files, processes and services belonging to malicious files such as backdoors and keyloggers which can later be used to gain access to everything on the system. Typically, rootkits penetrate personal computers and servers via viruses or vulnerabilities. After the rootkit is installed, conventional security products including anti-virus and spyware programs are unable to detect them or the files they are hiding.

Rootkits are an increasingly common 'stealth' technique used by malware authors to conceal their dark handiwork and intentions. Put simply, they are specialised toolkits that can hide malicious programs – whether they be viruses, Trojans, spyware, keyloggers and so on – from detection by conventional anti-virus and anti-spyware tools. Think of a rootkit as a cloaking device for malware, the kind that allows a hacker to move around your computer with complete impunity, undetected and unchallenged, doing as he pleases.

It is believed that this invisible form of malicious code will become a growing problem in the future. At the 2005 *RSA* security conference in San Francisco, *Microsoft Corporation* and security industry experts all expressed their concerns about the rising problem related to rootkits. To give one example, the *Windows XP* operating system is unable to show files or processes deployed by many rootkit programs. This leaves the user or administrator unaware of their presence. These types of stealth spyware program are believed to have been involved in some high-profile industrial espionage cases.

Since a rootkit can hide its presence on your system for a longer time than conventional malware, it is almost certain that ultimately it will be able to take your most confidential data. There are a number of different rootkits available on the market: some are feature-rich and include such functionality as the ability to log keystrokes, create secret backdoors and alter system log files, as well as offering administrative tools to prevent detection. Others are just tools to hide third-party files.

So, as is the case with many modern malware exploits, a PC or network could be fully protected against *conventional* malware with the latest in AV software, yet still unwittingly become infected with a rootkit – and therefore, completely vulnerable to attack. What's more, you may not even realise the attack is happening until it is too late and you have suffered loss of valuable data and money.

ROOTS OF ROOTKITS

So where do rootkits come from? Rootkits originally came from the *NIX world where the purpose of an attack was to give the attacker the control level of an administrator or 'root' – hence the name – and keep that access for as long as possible.

In the beginning, rootkits were mainly replacements for system tools. For example, the login program would be replaced by a modified version that stored the username and password combinations or the 'ls' tool that is used to list directory contents would be replaced by a rootkit version that would not print out certain file names.

Naturally enough, the malware community quickly found a window for exploits from rootkits, which led to the creation of integrity checking tools such as *TripWire* [1]. Such programs were designed to detect these first-generation rootkits by alerting the user to the modification of any system file.

Later generations of rootkits are, however, far more advanced in their range and functionality and have the ability to load themselves as kernel-loadable modules, thus avoiding detection by integrity checks.

Following the evolution of the PC market since Unix days, the latest generation of rootkits targets *Windows*-based machines. Nowadays there are a number of malware programs that use rootkits to hide from conventional detection, including the CoolWebSearch, Win-Spy, PC Spy, ActMon, ProBot SE, Invisible Keylogger and Powered Keylogger spyware programs. Some viruses themselves use rootkits to avoid detection and happily deliver their payloads, including Maslan and Padodor.

In addition to viruses and direct hacking via rootkits, there are several variants of backdoor Trojans, like SDBot and RBot, which incorporate the computer into a botnet that can be used by malicious people to send spam, perform denial

of service attacks and all the other types of exploit for which we typically see botnets being used.

The sophistication and speed with which rootkit techniques are now being applied to spyware and viruses may highlight the growing influence of organised online criminal groups in their bid to develop stealthy, invasive software, as opposed to the typical '15 minutes of fame' exploits performed by geeks and script kiddies. Whatever the ultimate reason, the intention is clear – to circulate malware into the online community which does not register on the users' security radar.

Rootkits have many entry paths to their intended host: they can be planted on a system by a hacker through an unpatched vulnerability, arrive as an attachment or as a download URL in an email. Once activated, the rootkit can be used to hide backdoors and tools that help the hacker maintain access to the hacked computer. This computer can later be used to attack other computers in the same network. Most crucial, however, is the fact that the rootkit will hide the hacker's tracks from current security software.

Having gained access to a computer hacked with a rootkit, the intruder is free to interact with network resources, files and systems with either the same or sometimes even higher privileges than the legitimate user. And if, for example, they gain access to an administrator's username and password, then they have all the keys to the kingdom – with the potential to cause widespread damage.

GHOST IN THE MACHINE

How do rootkits enable all this? Well, that depends on the type of rootkit that is being used. There are two types: user-mode rootkits and kernel-mode rootkits. To understand how they hide themselves in a system, let's look at how these two pieces of malware differ.

User-mode rootkits

A user-mode rootkit typically intercepts API calls in the system and modifies their output to hide files, registry keys and processes. A good example of this is a product called 'Golden Hacker Defender' sold openly on the Internet by its author, which also incorporates a Trojan that includes a built-in hidden door.

Kernel-mode rootkits

A kernel-mode rootkit, on the other hand, can be even more powerful than a rootkit running in user-mode. It can still filter the output of system API calls, similar to that of a user-mode rootkit, but it can also do much more. A common technique to hide a malware process is to remove the process from the kernel's list of active processes. As the kernel does not use this list to actually run the process (that is handled through the kernel scheduler) it's a very effective way of concealing the processes run by a hacker in your system.

Whichever way the rootkit operates, the goal is to stay hidden from security scanners. As most rootkits are also able to intercept the queries that are passed to the kernel and filter out the queries generated there, in effect, they are able to clean up any trace of their own activities. The result is that the typical footprints of a program, such as an executable file name, a named process that uses some of the computer's memory, or configuration settings in the OS registry, are invisible both to administrators and to all types of detection tool – even intrusion detection systems (IDS).

This ability of rootkits to clean log files and erase evidence of the actions it performs can make a hacker truly a 'ghost in the machine'. There are also tools for hiding the files and processes that the intruder may place on the system and even to hide port and protocol connections.

Some security pundits say that rootkits do not pose a significant problem, since more and more systems are effectively protected from outside intrusion which means it is difficult for a rootkit to be planted on a machine in the first place through the normal routes of infection. While this is true to some extent, no modern=day company would want to risk having an invisible backdoor into their network that could be accessed without any warning and used for any number of malicious purposes.

ROOTKITS FOR SALE

As the whole malware-writing scene is shifting quite rapidly towards an economic model where virus writers and botnets are available for hire at the right price, it is no surprise that you can buy your own version of a rootkit. Authors such as Holy Father (Hacker Defender) and Aphex (AFX Rootkit) both have custom undetected versions of their rootkits available for sale.

On the Hacker Defender website, a customer can select which rootkit detection programs he/she wants to buy 'undetection' from, where each application and version is bought separately. Or the customer can just simply buy the Gold or Silver version, which comes with undetection for the most common detection systems.

Spy applications such as ProAgent 2.0 even come with a one-year warranty where the buyer will get a new undetected version if any of the security vendors adds detection for your customized version. But as a lot of the rootkits are open source, an attacker doesn't even need to pay for an undetected rootkit: with some basic

programming skills he/she can just recompile it and thereby avoid detection.

UPROOTING ROOTKITS

So if, once they are installed, rootkits can evade conventional security tools, what can you do if you do discover you are harbouring a rootkit infection? Until recently, the prognosis was not good.

Although there have been some techniques for detecting rootkits, they are intended only for very IT-literate users who are conversant with code and all the other tricks of the trade: they certainly are not plug-and-play. What's more, they do not remove or quarantine rootkits. The standard advice for rootkit removal is to 'repave' – an innocent-enough-sounding euphemism which stands for completely scrubbing all data, applications and the operating system from the infected machine, and then reinstalling from scratch.

Repaving is simply not an option for most computer users who have stored all of their most precious material for safe-keeping in one repository. And if it is the case that more than one PC in a company is infected, the prospect of repaving multiple machines is still less attractive with all the attendant loss of business that follows.

TOOLS

However, new tools to help manage and contain the rootkit problem are emerging. Tools like *SysInternals*' *RootkitRevealer* [2] and *F-Secure*'s *BlackLight* [3] technology are able to scan a machine and detect hidden rootkit files. Some of them can even eliminate the files by renaming them, even though some people think that the only solution to remove a rootkit is to reinstall the system completely.

But, while these applications will detect rootkits, it will not be until these detection capabilities are built into existing anti-virus and anti-spyware applications, with centralized management, that users and corporations will be protected fully from the growing rootkit threat.

REFERENCES

[1] *TripWire*: http://www.tripwire.com/.

[2] *SysInternal*'s *RootkitRevealer* can be downloaded from http://www.sysinternals.com/Utilities/RootkitRevealer.html.

[3] A beta version of *F-Secure*'s *BlackLight* Rootkit Elimination Technology is available free of charge from http://www.f-secure.com/blacklight/.

FEATURE 2

SYMBIAN OS – MYSTERIOUS PLAYGROUND FOR NEW MALWARE

Robert X. Wang
Symantec, Ireland

In the last year or two, an increasing number of *Symbian* threats have been reported. Some of these threats have exploited various services such as Bluetooth, MMS (Multimedia Messaging Service), etc. On the other hand, there are still not many professional malware writers who are interested in the *Symbian* OS. However, this might change in the near future. Even worse, adware/spyware companies may also involve themselves with the *Symbian* OS. Imagine if people have a hidden dialer on their handset, a keylogger that traces PINs or other sensitive information. Is the *Symbian* OS in danger of further attacks?

MILESTONE

In June 2004, SymbOS/Cabir, the first *Symbian* worm to propagate via Bluetooth, was discovered (see *VB*, August 2004, p.4). This threat was a proof-of-concept worm without a payload.

In November 2004, another unpleasant name, became well known: SymbOS/Skulls. This overwrites many system files on the device. Although this threat contained nothing new from a technical point of view, it demonstrated that Symbian Installation System (SIS) files are a handy medium for attacks.

In January 2005, the first SIS file infector, SymbOS/Lasco.A, was discovered. This threat searches for SIS files on the device and appends itself to them.

In March 2005, SymbOS/Commwarrior.A, the first *Symbian* worm to propagate via MMS, was discovered (see *VB*, April 2005, p.4). Furthermore, this threat also attempted to disguise itself as a system kernel process, preventing other processes from changing its priority or terminating it.

Fortunately, these threats are not too complicated. Due to platform dependency and interactive requirements, they are also unlikely to become widespread in the real world. However, that doesn't mean we will be lucky forever. The question is what and when will the next *Symbian* threat appear.

There are many approaches that a threat might use to propagate itself, such as Bluetooth, MMS, email, SMS with malicious link, web browser, file infection, and vulnerable exploits. This article will focus on potential file infection and rootkit functionality. The *Symbian* OS is a fully object-oriented system. This might be one of the reasons

why we have been lucky enough not to have seen any live virus so far. Unfortunately, it is still vulnerable to potential file infection that may cause more troubles.

SYMBIAN OS IMAGE FORMAT

There are three major executable formats:

- Symbian E32 executable image: a format used by normal user applications and dynamically loading link libraries.
- Symbian E32 ROM image: a format used by ROM image files only. There are some similarities between E32 executable image and E32 ROM image. The major difference is that E32 ROM uses physical addresses instead of relative virtual addresses. The kernel loads ROM images directly into the specified address.
- Symbian Installation System (SIS): a format used by the installation system. This is much simpler than the above formats.

<table>
<tr><td colspan="2">E32 executable image header (0x7C bytes)</td></tr>
<tr><td rowspan="3">Code section</td><td>Text section (code and constant data)</td></tr>
<tr><td>Import table</td></tr>
<tr><td>Export table</td></tr>
<tr><td colspan="2">Data section (initialised data)</td></tr>
<tr><td colspan="2">Import section</td></tr>
<tr><td colspan="2">Code relocation section</td></tr>
<tr><td colspan="2">Data relocation section</td></tr>
</table>

Figure 1: E32 executable image format.

Symbian E32 is unlike the *Windows* PE file format. There is no official specification available. Fortunately, *Symbian* has released the source code of PETRAN that translates PE-COFF format into E32 format. The following code is copied from PETRAN source code (see http://www.symbian.com/developer/downloads/tools.html#SymbOSCppBt):

```
class E32ImageHeader
{
public:
  TUint32 iUid1;      // system UID
  TUint32 iUid2;      // interface UID
  TUint32 iUid3;      // program UID
  TUint32 iCheck;     // checksum of the above UIDs
  TUint iSignature;   // signature bytes: 'EPOC'
  TCpu iCpu;          // type of CPU
  TUint iCheckSumCode;    // sum of all 32 bit words
                          // in text section
  TUint iCheckSumData;    // sum of all 32 bit words
                          // in data section
  TVersion iVersion;  // version of PETRAN
  TInt64 iTime;       // time and date the file was
                      // created
  TUint iFlags;       // 0 = exe, 1 = dll, +2 = no
                      // call entry points
  TInt iCodeSize;     // size of code, including
                      // import address table,
                      // constant data and export
                      // address table
  TInt iDataSize;     // size of initialised data
  TInt iHeapSizeMin;  // min size of heap
  TInt iHeapSizeMax;  // max size of heap
  TInt iStackSize;    // size of stack
  TInt iBssSize;      // size of un-initialised data
  TUint iEntryPoint;  // offset into code of entry
                      // point
  TUint iCodeBase;    // where the code is linked for
  TUint iDataBase;    // where the data is linked for
  TInt iDllRefTableCount; // number of imported DLLs
  TUint iExportDirOffset; // offset of the export
                          // address table
  TInt iExportDirCount;   // number of exported
                          // functions
  TInt iTextSize;     // size of the text section
  TUint iCodeOffset;  // file offset to code section
  TUint iDataOffset;  // file offset to data section
  TUint iImportOffset;    // file offset to import
                          // section
  TUint iCodeRelocOffset; // relocations for code and
                          // const
  TUint iDataRelocOffset; // relocations for data
  TProcessPriority iPriority;    // priority of this
                                 // process
};
```

E32 format has three checksum fields:

- iCheck is the checksum of the top three UIDs.
- iCheckSumCode is the sum of all 32-bit words in the text section.
- iCheckSumData is the sum of all 32-bit words in the data section.

This check is simple; malicious programmers could easily patch the code and generate a new checksum.

The *Symbian* OS is based on the ARM architecture. It supports two types of instruction set: ARM (32-bit) and THUMB (16-bit). Each of them includes instructions to switch the processor state. All ARM instructions have a fixed length. If bit 0 of iEntryPoint is set, the program will be started in THUMB state, otherwise, it will be started in ARM state. All these features are extremely beneficial to polymorphism.

By default, the value of iEntryPoint is 0, which means that the code always starts from the beginning of the text section. However, it can also be redirected to other locations within the text section. When a program is launched, the kernel checks UIDs, signature bytes, type of CPU and the checksums, but it does not check the range of the sections. Obscured entry point, patched import table and many other traditional tricks may be used in the *Symbian* world.

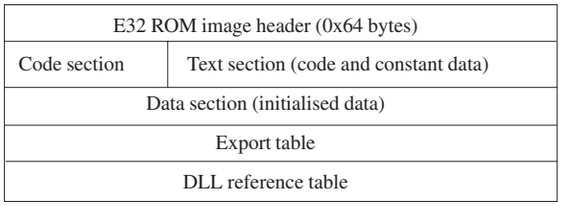


Figure 2: E32 ROM image format.

Compared to the E32 executable image format, the E32 ROM image format is more compact. It uses physical addresses instead of relative virtual addresses. The code section, the export table and the data section will be mapped to specified addresses. The ROM image uses a DLL reference table instead of traditional import table, all API calls are invoked directly to the physical address. All these features make the ROM image more powerful.

Almost all existing *Symbian* threats use SIS files to propagate. Other than Trojans, Commwarrior and Lasco have already demonstrated that SIS files can easily be generated or patched on the fly. Furthermore, there is no check against the first DWORD (system UID) of a SIS file, which makes it more dangerous and may cause potential cross-platform infection.

BOOTSTRAP

The *Symbian* OS bootstrap process is as follows:

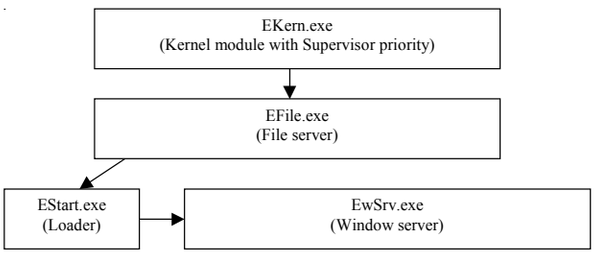


Figure 3: Symbian OS bootstrap process.

Unfortunately, processes after EFile.exe are unprotected. This may leave an opening for malicious code to increase its priority. Furthermore, there are many undocumented kernel APIs. By default, kernel calls are typically made through euser.dll. Euser.dll is responsible for handling direct user API requests and uses software interrupts to enter privileged mode. Kernel calls can also be made directly or indirectly. These hidden exports may also help malicious code to gain unauthorized control.

RUNTIME ENVIRONMENT

On the *Windows* platform, all running programs are locked by the system to prevent files from being modified by other processes. When a program is launched in *Symbian*, the kernel maps the file to memory and then releases it. This means that a virus can infect almost any executable file, regardless of whether it is running or not.

The kernel provides a virtual machine environment for user processes. The data of each process is mapped into the same virtual address range, 0x00400000 to 0x3FFFFFFF.

- The '.data' section (initialised data) is mapped to 0x00400000; the '.bss' section (uninitialised data) is mapped right after the '.data' section.
- ROM code is mapped into the range 0x50000000 to 0x57FFFFFF.
- ROM export table is mapped into the range 0x58000000 to 0x5FFFFFFF.
- RAM device driver is mapped into the range 0x60000000 to 0x7FFFFFFF.
- ROM '.data' and '.bss' sections are mapped into 0x8XXXXXXX.
- RAM code is mapped into 0xFFXXXXXX.

The Memory Management Unit translates only the virtual address of user data. User processes share the same map of ROM code. This may leave another opening for viruses or rootkits to hook into the system.

From version 6.0 of the operating system, several functions have been withdrawn. By default, the kernel no longer allows user programs to create remote threads. User programs must not use the E32 ROM image format, otherwise they will not be loaded and executed.

Does that mean we are safe? No. A file in E32 ROM image format can also be loaded from RAM. The SymbOS/Doomboot.A Trojan has demonstrated that. The kernel may load libraries from RAM instead of ROM. Malicious code may use a combination of exploits to hook into the system and infect or encrypt files.

CONCLUSION

Symbian OS is a very successful and powerful operating system for mobile devices. As the number of mobile devices in use continues to increase significantly, attempts to attack the *Symbian* OS may impact millions of users.

In theory, almost every type of attack might be found in the mobile world: stealing sensitive information, displaying fraudulent information, performing DDoS attacks, opening backdoor shells, starting hidden dialers, lowering security settings, firing dirty payloads, and so on.

Are we, as security professionals, ready to handle more complicated threats and various attacks?

FEATURE 3

NEW MALWARE DISTRIBUTION METHODS THREATEN SIGNATURE-BASED AV

Oren Drori and Nicky Pappo
Commtouch Software, Israel

Dan Yachan
International Data Corporation (IDC)

For some time now, viruses have been designed for rapid distribution during the few hours before anti-virus update signatures are produced (as discussed in a previous article by one of the authors, see [1]). In a recent report *IDC* stated that achieving high propagation rates is one of the main design goals of malware authors today [2]. Modern viruses and worms are not immune to vaccinations – rather, they are designed to infect as many computers as possible before vaccinations become available.

As a result, a timely response has become a key factor in effective protection against malware, and a major challenge for the AV industry. We have argued that all signature-based methods need powerful complements to provide early-hour (preferably zero-hour) protection.

NEW DISTRIBUTION METHODS

In recent months, however, there has been a decided shift in malware distribution patterns. The new breed of malware is distributed in ways that enable attacks to be executed fully before they can be blocked by signatures. Widespread adoption of these new distribution methods could pose a serious threat to signature-based protection methods.

In this article, we identify two new malware distribution methods: short-span attacks and serial variant attacks. We describe their particular distribution patterns, the development of recent attacks, and the potential dangers they present.

MALWARE DISTRIBUTION PATTERNS

Classic malware uses a viral distribution pattern, in which one infected station infects another, and an epidemic develops. Traditionally, an outbreak of this type would grow gradually and peak after several days (see Figure 1a). This distribution pattern allows AV vendors valuable time to produce and distribute signature updates (although some of the viruses penetrate during the first hours). As powerful and dangerous as these attacks may be, signatures are still effective against them, unlike in the case of short-span attacks.

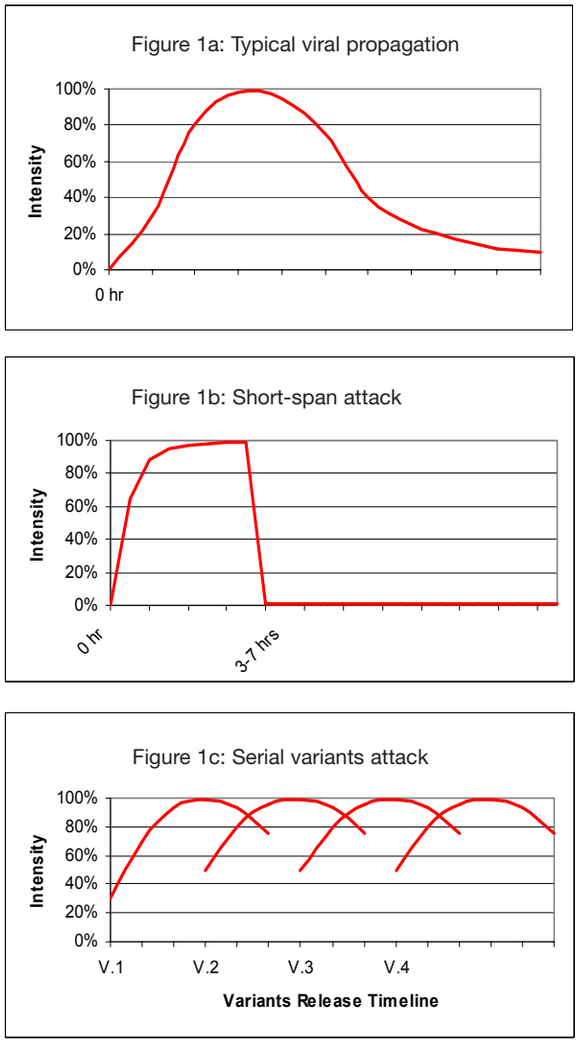


Figure 1: Malware distribution patterns.

SHORT-SPAN ATTACKS

No doubt the increasing spam-virus symbiosis plays a part in malware distribution patterns. The short-span attack combines the distribution methods of spam with the payload of malware: this type of attack is mass-mailed, mostly without any mechanism for self-propagation.

Typically, an entire short-span attack is completed within a few hours, sometimes within as little as 20 minutes. Outbreak-scale attacks, distributed via zombie networks, can infect many millions of users before signature protection is available. As a reference, large zombie-based spam attacks distribute 100–200 million messages, within five to seven hours.

Unlike viral-propagation attacks, which die slowly, short-span attacks have a spam-like distribution pattern: rapid buildup, steady distribution rate throughout the attack, and almost instant dropping off (see Figure 1b). According to *IDC*, this technique is highly effective for Trojan distribution, and is often used in financially-motivated attacks [2].

In many short-span attacks, AV vendors avoid the trouble of developing a signature that will be obsolete by the time it is released.

During the month of June 2005 alone, *Commtouch* identified four short-span malware attacks, which were completed within one to seven hours (see Figure 2).

Short Span Attacks in June 2005				
Attack	**Named by**	**Date**	**Intensity**	**Span**
Goldun.BA	[Commtouch]	03-Jun-05	Medium	1 hour
Goldun.BB	[Commtouch]	17-Jun-05	Medium	45 minutes
Flooder.Agent-1	[ClamAV]	19-Jun-05	Low	1 hour
Flooder.Agent-1, variant	[ClamAV]	20-Jun-05	Low	1 hour
Beagle.BQ	[Symantec]	26-Jun-05	Very high	7 hours

Figure 2: Short-span malware attacks in June 2005 (measured by Commtouch Labs).

The most severe of these attacks was Beagle.BQ, which started and finished within seven hours. Of 20 major AV engines tested independently by *VirusTotal*, 10 did not manage to produce a signature before the end of the outbreak. 24 hours later, seven AV engines still had no signature for it at all (see Figure 3).

Beagle.BQ was one of the most intense attacks seen so far in 2005, perhaps the single most forceful one. Faced with it,

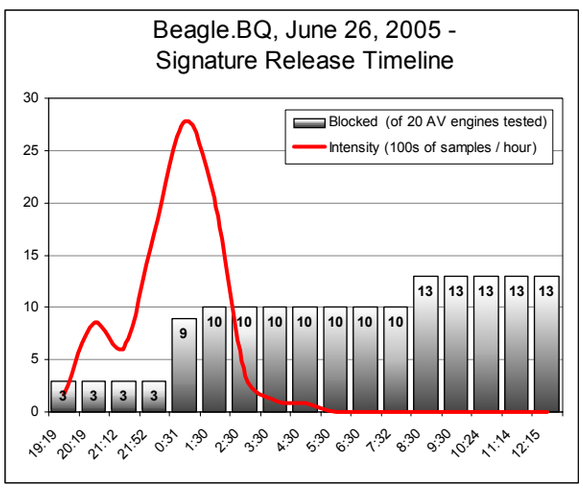


Figure 3: Beagle.BQ short-span attack. Sources: attack intensity based on data from Commtouch Software [3], signature updates based on VirusTotal [4].

35% of commercial AV users obtained adequate protection only halfway through the attack, and 50% of products failed to provide adequate protection throughout the entire attack.

SERIAL VARIANT ATTACKS

Serial variant attacks not only make use of the early-hour vulnerability window in traditional AV methods, but extend it by a cumulative factor.

A series of variants, prepared in advance, are launched at timed intervals. Each of the variants requires a new signature; each outbreak therefore enjoys its own window of opportunity, its own open distribution time, unimpeded by signatures. The overall window of vulnerability of the attack is the cumulative vulnerable time span of the individual variants (see Figure 1c).

To maximize the vulnerability period, the malware distributor uses a larger number of variants. Theoretically, if an unlimited number of variants could be added to the series, it would mean extending the window of vulnerability indefinitely.

In order to maximize distribution intensity – the number of infections or penetrations per hour – the malware distributor would aim to release the variants at very closely-spaced intervals.

Example: MyTob

One example of a low-volume, long-term serial variant attack is MyTob, releasing, on average, one new variant every day over the course of six months (see Figure 4 for the list of variants in July 2005).

Even though the functionality of the different MyTob variants is identical, a new signature must be produced for each one. Considering an average production cycle of 10 hours (see [5]), and a new variant every day, this means that the average paying AV user is unprotected from MyTob for 10 out of 24 hours, or 42% of the time.

MyTob Variants, July 2005	
27-Jul	W32/Mytob-HU
26-Jul	W32/Mytob-DX
25-Jul	W32/Mytob-BV
25-Jul	W32/Mytob-DW
23-Jul	W32/Mytob-HM
23-Jul	W32/Mytob-HN
21-Jul	W32/Mytob-IN
21-Jul	W32/Mytob-DV
21-Jul	W32/Mytob-DU
20-Jul	W32/Mytob-CX
20-Jul	W32/Mytob-DT
18-Jul	W32/Mytob-DS
18-Jul	W32/Mytob-DR
18-Jul	W32/Mytob-DQ
13-Jul	W32/Mytob-DP
13-Jul	W32/Mytob-DN
12-Jul	W32/Mytob-DM
12-Jul	W32/Mytob-DL
12-Jul	W32/Mytob-DK
11-Jul	W32/Mytob-DJ
10-Jul	W32/Mytob-DI
9-Jul	W32/Mytob-DH
8-Jul	W32/Mytob-AS
7-Jul	W32/Mytob-IU
7-Jul	W32/Mytob-DG
7-Jul	W32/Mytob-DE
7-Jul	W32/Mytob-DF
7-Jul	W32/Mytob-DD
5-Jul	W32/Mytob-DC
5-Jul	W32/Mytob-DB
5-Jul	W32/Mytob-CY
1-Jul	W32/Mytob-CW

Figure 4: Serial variants MyTob attack.

Example: Beagle

At the other end of the spectrum are attacks that maximize distribution density, by releasing multiple variants within a short time span. One good example is the Beagle attack of 1 March 2005 (Beagle.BB-BF) – an aggressive, high-volume attack that included no fewer than 15 different new variants in a single day, or almost one new variant per hour.

At the end of the day, *Kaspersky*'s team recounted the news [6]: 'Today we have already intercepted 15 new pieces of malware produced by the author of Beagle. The newest variants follow hard on the heels of our updates and we suspect that the author is creating new variants every time we release updates to block previous versions.'

CONCLUSION

In the past two to three years, malware developers have zeroed in on the early-hour vulnerability gap of traditional AV protection methods. Focusing on this 'sweet spot', they have developed new ways of distributing malware, which not only use, but also extend the early-hour gap in AV protection dramatically.

So far, these particularly pernicious types of attack are a minority on the landscape of malware. Nevertheless, these aggressive short-span attacks and serial variants have the potential of becoming the norm. If such a thing were to happen, it would represent a game-changing event in the AV industry. We believe it is crucial for the AV industry to prepare immediately the technologies to protect users from emerging early-hour distribution attacks.

REFERENCES

[1] 'Virus outbreak protection: network-based detection', Oren Drori, *Virus Bulletin*, March 2005.

[2] 'Zero hour virus protection: defending against the unknown', Dan Yachin, *IDC*, August 2005.

[3] Commtouch Software, http://www.commtouch.com/.

[4] VirusTotal, http://www.virustotal.com/, *VirusTotal* is an independent service that uses multiple anti-virus engines to analyse suspicious files. It facilitates the quick detection of viruses, worms, Trojans, and other kinds of malware detected by each of the anti-virus engines. Data documented by *Commtouch*, during the outbreak time.

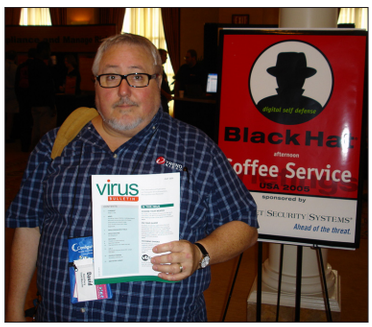
[5] Andreas Marx, AV-Test.org, http://www.av-test.org/, *Proceedings of the Virus Bulletin International Conference* 2004.

[6] Kaspersky Lab, *Analyst's Diary*, 1 March 2005, http://www.viruslist.com/.

CONFERENCE REPORT

BLACK HAT AND DEFCON – TOO HOT FOR MANY

David Perry
Trend Micro, USA



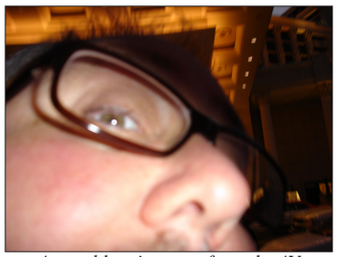
A wise man once told me that the difference between responsibility and blame is that responsibility happens before the fact, and blame happens after the fact. Bear that in mind.

I went to Las Vegas in July to attend both the Black Hat Briefings and DEFCON, at the behest of *Virus Bulletin*, who had asked me to write up a report of the proceedings as I saw them. So without digressing, I will get right to the subject at hand.

Now, you always hear about 'Black Hat *and* DEFCON', so just to set the record straight, the two are very different things. Black Hat is a very serious conference intended to illustrate top issues in the world of network security, and DEFCON is a 'through-the-rabbit-hole' con, where not only is everyone there a poseur, but everyone is proud to admit that everyone there is a poseur.

When registering for Black Hat, you are given a backpack containing the conference proceedings (a paperback volume the size of a very large phonebook) and a number of other useful items. A closer inspection of the proceedings volume showed that the rumours were true – a whole presentation had been torn neatly out of the volume – and the CD versions of the proceedings had been rudely withdrawn to a secret location where each was ceremonially destroyed under the watchful eye of a trained exorcist.

The missing presentation was Michael Linn's *CISCO* disclosure – a subject so controversial that no two people agree on what it really means. You cannot see the slides, you cannot see the video or hear the audio recordings made of the presentation (both were seized by a local court following a cease and desist order), and you can't get a clear story about exactly what happened, but I will tell you this about Michael's presentation: it was *really* crowded! After standing and listening to about 15 minutes (including the famous 'Welcome to the Eighties' line – upon which I will not elaborate here) I did what any other reasonable conference-goer would do – I went to another room, to let everyone else report on the big enchilada.



A notable miscreant from the AV industry (aka David Perry).

I ended up in a panel on certification, listening to a very august panel discuss, among other things, common criteria and other government mandated certs. A notable miscreant from the AV industry made comment that NIAP certification (US common criteria testing and certification) was overpriced and functionally useless – typically taking much longer than the product's shelf life to achieve. This led to a long series of hallway discussions with both government and private industry types.

After lunch was a presentation entitled 'Owning anti-virus: weaknesses in a critical security component', by Alex Wheeler and Neel Mehta. These very earnest guys had some great tips for breaking applications, like 'take lots of notes in notepad' and 'print everything out and write in the margins'. They detailed three vulnerabilities, one each in *Trend Micro*, *Symantec* and *McAfee* anti-virus programs. Apparently none of these items will work on a current version. To their great credit, Wheeler and Mehta were very generous with their praise for the anti-virus industry, noting that AV products were 'much safer than anything else in the industry' and that 'patches and fixes are regularly updated'. (Patches and fixes? Isn't that pretty much all we do?)

Another miscreant from the AV industry (aka Andrew Lee).

Of course, another miscreant from the AV industry stood up in the question and comment section and asked why they did not discuss the vulnerability that led to the WITTY worm. Since both speakers were employed at *ISS*, this was largely a rhetorical question.

The rest of the day was taken up with a lot of very amazing discoveries (long range RFID detection, more pod cast interviews, and a very interesting reception).

Day two of the Black Hat Briefings contained the same four tracks as day one: Applications Security, Forensics, Privacy and Zero Day Defense. There were way too many presentations for any one person to see, so it is a good thing that all of the presentation slides are given to every attendee in a large book (it builds character to carry this albatross for two days, not to mention muscles). There was a new product announcement from Phil Zimmerman (secure VoIP) and a corresponding hack discussion about something called 'SIP fuzzing' (you could never make this stuff up) which is ostensibly either a way to break VOIP security, or a tasty beverage.

My final session at Black Hat was on the US national ID card. Now, as a US citizen, I have a passport, a driver's licence, a Social Security number, a medical insurance card, several credit cards, a COSTCO membership ID, an auto club card, a library card, a dental plan card, an ID card for my office, an ATM card and an annual passport to Disneyland. My wallet is at least an inch thick, and is already causing spinal displacement in an adult American male. So now there is some need for a national ID card?

Apparently this is an issue of great significance and controversial impact. Each of the panel members had something unrelated and, well, totally ambiguous to say about the national ID card program. Like any great panel discussion, this muddied the waters beyond all possibility of repair. And each panellist took great pains to discuss the national ID card issue in light of the WTC bombings of 9/11/01. (I know this is an English publication, but we Americans insist on listing dates out of order.)

Good thing a noted miscreant from the AV industry (noticing a pattern here?) stood up and asked a question that brought thunderous applause from the crowd.

Black Hat is a full week long between the seminars and briefings, and is a very high content security seminar. If you are a professional in this industry you would be well served to attend the Black Hat Briefings at the very least. It is quite expensive, but very informative. One industry professional (who shall remain nameless) is quoted saying, 'At Black Hat you hear first about what you spend the next year fighting'. Sure, his syntax is tested to its theoretical limits, but the lesson is clear: this conference is the real thing. To be responsible, find out ahead of time.

DEFCON – THE OTHER SIDE OF TOWN

DEFCON is an old Paiute Indian word meaning 'many black T-shirts for sale' and is billed as 'the largest hacker gathering on planet earth'. DEFCON (which is actually named for the old pre-colour-code national defence alert status) is everything that Black Hat is not. DEFCON is immensely silly, yet frequently takes itself too seriously. DEFCON is only the weekend long, incorporates games like 'Spot the Fed', 'Capture the Flag', and competitions like 'WarDriving' and 'Lock Picking'. Where BH is aimed at the system administrator learning to protect his network, DC is aimed at the '1337' kid trying to break into something. There is a lot of hair dye at DEFCON.

DEFCON has the most amazing check-in of any computer conference or show in the whole world. Those familiar with conferences know that registering or checking in often means filling out a long form where people try to find out how to sell your soul to a marketing organization.

At DEFCON, you stand in a line at the Alexis Park hotel to pay 80 dollars in cash, in exchange for which you are handed a badge, a CD and a conference schedule the size of a pamphlet, then hustled out the door by a person officially designated as 'goon'. Not only do they not care whether you buy networking equipment for your company, they don't even want to know your name! For 80 dollars (cash) at a hacker convention, everyone is who they say they are (or nobody is).

The conference badge itself is a wonder to behold. This year's was a rectangle of day-glo Perspex with die cut punches to spell out DEFCON 13, the bearer's designation (HUMAN, in my case) and the charming happy face skull and crossbones logo of DEFCON. Each year's badge is a different and unique design.

The Alexis Park has no casino, and the lobby is the main hangout for attendees. There were a pair of ATM machines in the lobby, each proudly displaying a DEFCON 13 logo on the screen. Very few people actually used the ATM ... it is suspect of being owned, or worse.

During the entire weekend, hackers sit in a ballroom trying to capture the flag. Any system broken is displayed (with the user's photo) on the 'WALL OF SHEEP'. These young men (and a small but growing number of women) are too focused to speak, eat or sleep – they are there simply to hack.

Presentations covered a large number of topics, none quite as memorable as 2004's presentation by a drunken self-professed virus writer in a kilt, but I shan't detail the DEFCON presentations (see http://www.defcon.org/ for a complete listing).

DEFCON also featured a dealer's room, where one can find antennae (for WarDriving), t-shirts books, stickers, lock picks, and surplus electronic gear.

DEFCON might consist mainly of poseurs, as many say, but it is the mythic heart of the hacker world. I don't imagine I will miss it ever again. It's worth at least a look.

If you fly over Las Vegas at night you can see the brightest lit mile of any street in the world. Dazzling neon and flashing lights display each casino in what looks, from the air, like a toy, or a brilliant trap set on the pitch black desert night, a trap to catch the superstitious, the illogical, the cocky. Las Vegas hates computer geeks – you know, we don't tend to gamble much per person. There is one good reason for that: we can do the math.

SPOTLIGHT

THE COMMON MALWARE ENUMERATION (CME) INITIATIVE

Jimmy Kuo
McAfee AVERT, USA

Desiree Beck
MITRE, USA

The Common Malware Enumeration (CME) initiative is an effort headed by the United States Computer Emergency Readiness Team (US-CERT, www.uscert.gov). Established in 2003 to protect the USA's Internet infrastructure, US-CERT coordinates defence against and responses to cyber attacks across the nation. US-CERT interacts with federal agencies, industry, the research community, state and local governments, and others to disseminate reasoned and actionable cyber security information to the public.

Through the adoption of a neutral, shared identification method, the CME initiative seeks to:

- Reduce the public's confusion in referencing threats during malware incidents.
- Enhance communication between anti-virus vendors.
- Improve communication and information sharing between anti-virus vendors and the rest of the information security community.

CME is fashioned similarly to the Common Vulnerabilities and Exposures (CVE) initiative (http://cve.mitre.org/), which is also operated by *MITRE* in support of US-CERT. As experience with CVE shows, once all parties have adopted a neutral, shared identification method, effective information sharing can happen faster and with more accuracy.

A CME Preliminary Editorial Board (CME-PEB) has been brought together to work with US-CERT to help bring the CME concept to maturity and expand CME's reach to other members of the anti-malware community. At the time of writing, the members of the CME-PEB represent:

- *McAfee*
- *Symantec*
- *Trend Micro*
- *Microsoft*
- *Sophos*
- *Computer Associates*
- *Norman*
- *Kaspersky Lab*
- *MessageLabs*
- *F-Secure*
- *ICSA Labs*

Oversight of the board is provided by US-CERT and *MITRE*.

The CME Initial Operating Capability (CME-IOC) was stood up at the end of the first quarter of 2005 to provide a limited operational capability for CME identifier acquisition. A CME website will be available in the fourth quarter to introduce the initiative to the public (http://cme.mitre.org/).

REDUCING PUBLIC CONFUSION DURING MALWARE OUTBREAKS

It is apparent that anti-virus companies are having an increasingly difficult time staying coordinated with virus names during computer virus outbreaks. As a result, products report a variety of names and variant designations for the same outbreak. This results in widespread confusion, with members of the public having to determine whether there is a single outbreak underway, whether there are multiple outbreaks underway, or whether they are seeing a new and different outbreak altogether.

Having to determine whether the protection they have in place is effective against the current outbreak increases the public's burden further. For example, the spring of 2004 was an extremely difficult period. Three or more Netsky variants appeared along with new variants of Mydoom, Bagle and Beagle, all within a couple of days. Network administrators were pulling their hair out as they tried to determine whether or not they had the protection they needed.

The CME initiative does not offer to coordinate all the anti-virus companies so that they use one and the same name (although we hope that name coordination will improve eventually, as a side-effect). Rather, the CME initiative will match a CME identifier to a particular threat, with the hope that most anti-virus entities, as well as other security-related entities, will adopt its use. This will allow the public to cross-reference the disparate names through a common identifier.

Note the word 'threat'. This is different from the normal course of anti-virus procedure in detecting and naming singular virus-related files. A 'threat' is a single entity encompassing any number of files that may be involved in a single outbreak. For example, all the components of Nimda – the *IIS* buffer overflow byte stream, the file that is passed through TFTP, the mass-mailed email it creates that attacks via the audio/x-wav vulnerability, the appended html pages or any of its other forms – will be referenced by one CME identifier.

THE CME IDENTIFIER

Initially, CME identifiers will be in the format 'CME-N' where N is an integer between 1 and 999. Digits will be added when the remaining unused identifier space becomes too small.

To accommodate space-deprived anti-virus products, CME IDs can be abbreviated (e.g. M123 or M-123), but the

official format (e.g. CME-123) should be used in places such as web pages, encyclopedias, etc.

For the sake of successful text-based comparisons, leading zeros will always be omitted in an identifier. For example, CME-00123 will always be written as CME-123. Identifiers will be generated randomly within each size range (e.g. CME-439 might be issued before CME-28). This way, it will not be possible for someone to assign their own identifier by guessing the next in sequence.

By minimizing the number of characters used initially, it is hoped that many anti-virus products will be able to add the CME identifier directly to the names their products display for the user. For example, a virus named 'NewOutbreak.A' will be shown to the user as 'NewOutbreak.A!M-555', while at the same time, another anti-virus product might report the same outbreak as 'OldFamily.CC!M-555'. In this way, a user will be able to ascertain quickly whether or not two viruses are the same, and as a result user confusion will be reduced.

THE PROCEDURE

The public needs the most guidance during virus outbreaks. For that reason, the CME-IOC will begin by addressing only the situations that satisfy outbreak conditions. Since most of the initial member organizations on the CME-PEB have representatives who also participate on the Anti-Virus Emergency Discussion list (AVED, http://www.aved.net/), the CME-PEB will follow a similar approach during IOC to identify high visibility threats warranting CME identifiers.

When a qualifying threat occurs, a CME participant will request a CME identifier. The participant will provide a sample and as much supporting information as possible. In response, an automated system will generate a CME identifier and will redistribute the submitted information to the other participants.

A CME identifier will then have been attached to the sample and its corresponding threat. Each CME participant will then disseminate the CME identifier as quickly as possible to those entities with which it regularly communicates in the industry and will reference the CME identifier on their web pages, in their product, or when speaking to the press, as best as can be achieved.

Use of the CME identifier is completely voluntary. However, we hope that anti-virus product users will encourage their preferred vendors to adopt CME identifiers. Widespread use of the CME identifier will help us all communicate more effectively about threats. Using CME identifiers, we will know when two threats are equivalent and when they are not.

DECONFLICTION

Deconfliction is a term that originated in the military. Here, we use it to refer to the activity required to avoid issuing more than one CME identifier to equivalent threats.

The first step in deconfliction is when a CME identifier is issued automatically. At this point, automated issuance of CME identifiers is turned off for the next two hours. Two hours was chosen as a reasonable amount of time for the CME-IOC. It may be adjusted as needed. This two-hour moratorium prevents messages that may have passed in the ether to cause two CME identifiers to be issued for the same event.

During the two hours following the issuance of a CME identifier, additional CME identifier requests will be deferred until the participants can decide whether the submitted samples constitute a new threat or are equivalent to the previous threat. If the participants agree that a submitted sample is a new threat, then an additional CME identifier can be 'forced'.

ADOPTION

Samples distributed. Matching CME identifier produced. The next step is the most critical. We must garner adoption of the CME initiative among anti-virus product producers.

Long ago, one of the authors argued that the 'virus-naming mess' is not a technical problem. The problem and solution lie in the willingness of the product producers to *want* to help resolve this mess. Supporting and, as applicable, participating in the CME initiative is a bold first step in announcing to your users that you want to help alleviate their confusion. There are certainly technical challenges to coming out of the gate with all products all using the same name. But these challenges cannot be said to hold true 48 hours, one week, or many weeks after an outbreak.

First, coordinated CME identifiers. Then, maybe we can solve this naming mess!

Editor's note:
The CME-PEB will be holding a Birds of a Feather session at the Virus Bulletin conference in Dublin, as an opportunity for VB2005 delegates to learn more about CME and its current status, as well as provide feedback and express interest in future involvement. The BoF session will take place after the close of the first day of the conference: 6pm–7pm Wednesday 5 October 2005 at the conference venue. Also at VB2005, Vesselin Bontchev will be presenting his take on the 'virus-naming mess' with a paper on the current status of the CARO malware-naming scheme. Register for the conference now at http://www.virusbtn.com/conference/vb2005/.

PRODUCT REVIEW

MCAFEE VIRUSSCAN ONLINE

Matt Ham

It has been some time since a *McAfee* product has been the subject of a standalone review in *VB*, hence the return to its study. The look and feel of *McAfee*'s corporate products have changed relatively little during that period and, while there have been additions to the feature set, and the particular flavour-of-the-month requirements of the market have stressed different aspects of the product, a user from several years ago would be able to launch and use today's version of *VirusScan Corporate Edition* without much confusion. This is good news for corporate users, but not so good for a reviewer looking for impressive changes to describe. Therefore, it is the home-user version of the software that is under review here, the '*Online*' appellation arising from the ability to install *VirusScan* directly from the Internet.

In addition to the *VirusScan* product for home users, *McAfee* also offers a personal firewall, anti-spam product and an anti-spyware product. These can be purchased as a bundle or as more of a mix-and-match set of products. These products will be mentioned in the text, but were not tested.

INSTALLATION AND UPDATES

As the product name suggests, the method of installation for this test was directly from the Internet. In theory, the user heads to the *McAfee* website, enters their login details and the installation process proceeds in a matter of a few mouse clicks. Unfortunately my experience of installation was not ideal, although it is unlikely that many home users will encounter the same number of problems, since they are likely to have different default settings on their machines.

The first problem occurred when I used my default browser, *Mozilla Firefox*, to access the *McAfee* website. Although the login process was reached with no problems, an error page came directly after this, stating that *Internet Explorer* was required for operation. Grumbling somewhat, I launched *IE* and stumbled again – this time after the licence agreement had been accepted. In this case my faux pas was that I had chosen to leave ActiveX disabled.

Although not totally unexpected, the fussiness of the installation process was ironic on two fronts. First, the use of *Firefox* and disabling of ActiveX are both recommended as easy ways to lessen the inherent security risks which beset anyone surfing the Internet. Relaxing security in order to install security products has always caused me wry amusement. The second, rather more interesting irony of the situation came as I explored other parts of the *McAfee* site.

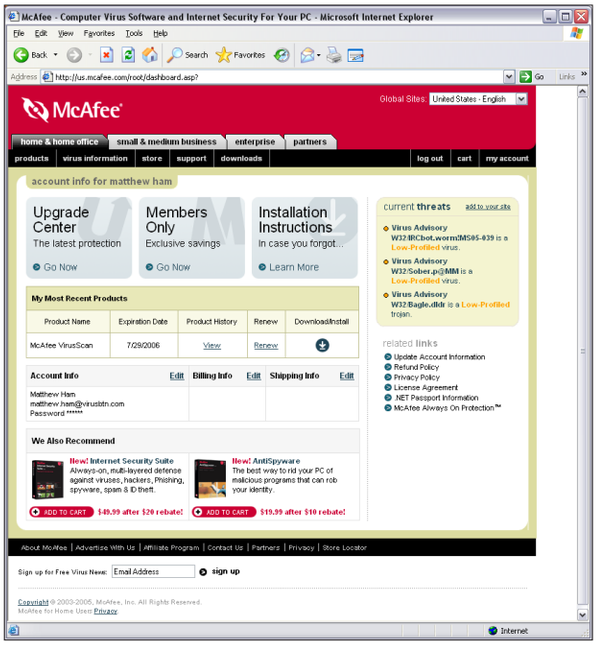
An affiliate of *McAfee* offers an Internet security program, *SecureIE*, which is advertised on several pages of the main website. The advertisements lead to external pages where a routine 'security check' of one's computer is offered. Imagine my surprise when, with no settings changed within *IE*, the *SecureIE* page declared that I was running the risk of ActiveX affecting my machine maliciously. The security on my machine was both too great to install *McAfee* software and too little to suffice according to *McAfee*'s affiliate – an interesting paradox.

Having instructed *IE* to accept the ActiveX control in question, the installation continued momentarily before announcing that pop-up blocking (which is standard as part of *XP SP2*) was rendering the installation process impossible. It was no surprise, considering the way things had progressed thus far, that the suggested solution was to disable this security feature until installation had been completed. I selected the option to allow pop-ups temporarily through the information bar which *IE* creates as part of the blocking process, and at this point the *VirusScan* Installation Wizard arrived on the scene.

The installation process was described as taking around 30 minutes on a 56Kbit dial-up connection, which seemed rather an optimistic estimate considering that the download time on a 1Mbit connection was close to five minutes. Nevertheless, this is remarkably speedy considering the size *McAfee* products have been in the past. Clearly some paring down has been done for the sake of efficiency.

I was a little surprised that, until this point, there had been no warning about other security products that may be installed on the machine. The warning comes on completion of the download phase. Since I am naturally paranoid, and I was using a test machine which is not totally expendable, my machine was already protected by a competing product. *McAfee*'s installation routine detected this and announced that the product should be removed, stating rather sternly that *VirusScan* would not function correctly if the competing product were not disposed of. This is a little worrying, since if *Microsoft*'s anti-virus APIs are implemented and interfaced with properly, two or more on-access scanners should be able to exist side by side. Either *McAfee* has no great confidence in some part of the process, or the appearance of this message is dependent upon the product detected, with a known issue having been discovered at some point.

The uninstallation of the other product triggered *XP SP2*'s various warnings about the subsequent lack of protection. The lack of protection was something of a concern for me, although I suspect that changing from one anti-virus developer to another is not a situation encountered very often by the majority of home users.



In any case, the machine reboots speedily after the uninstallation, so that the installation of *VirusScan* can proceed. Several installation selections are made after reboot and before installation, however, which delays the process of restoring on-access functionality to the machine. It would be more sensible to relocate these choices to a less time-sensitive part of the installation process – after all, the creation of a desktop icon and participation in the Virus Map scheme are not particularly good reasons for keeping the machine unprotected.

The next stage consists of the installation of *McAfee SecurityCenter*. While the *McAfee SecurityCenter* may be installed in addition to the *Windows SP2 Security Center,* the installation dialog recommends that the *Windows* version is disabled so as to avoid duplicate security status messages. (Although the disabling of the *Windows Security Center* is at least automatically reversed if *SecurityCenter* is removed at a later stage.)

After this choice is made, a rather belated informational dialog arrives, detailing the capabilities of the *McAfee SecurityCenter*. It would have been much more useful to have been presented with this dialog before having to decide whether it should be the default area for such operations.

After this rather long series of events, *VirusScan* is installed fully on the target machine. A final dialog box is all that remains, offering a full scan for viruses and 'What's new?' information. Considering the unprotected state of the machine during parts of the installation procedure, it seemed wise to accept the scan.

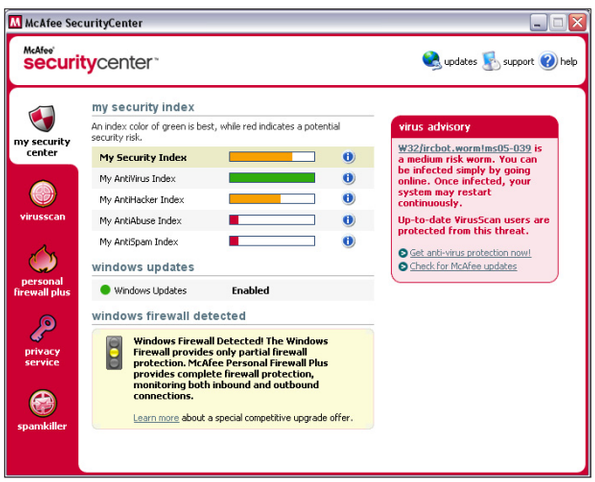
Updates are checked each day by default, with pop-up warnings in the case of outbreaks being declared. Somewhat oddly, the pop-up for obtaining protection from these new threats does not link to a download site, but instead to the area on the *McAfee* site where the product may be purchased. Presumably the pop-ups are produced by the *Security Center* without reference to the presence or otherwise of *VirusScan*.

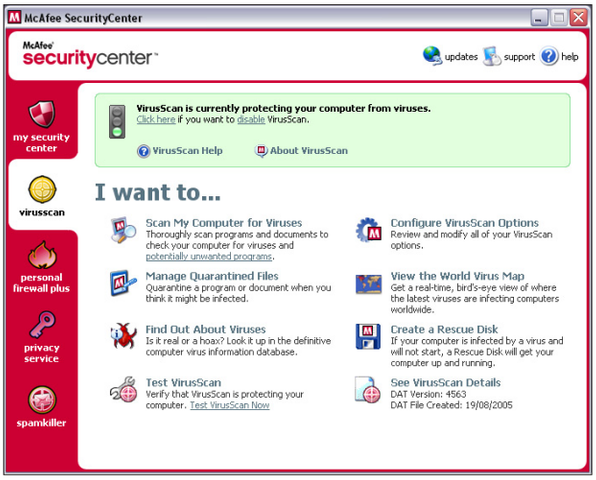
DOCUMENTATION AND WEB PRESENCE

As indicated, this version of *VirusScan* assumes that there is an Internet connection. Indeed, the operation of *VirusScan* and much of the additional content linked from within alerts and the *SecurityCenter* require the presence of an always-on Internet connection in order to be of any value. For example, the option exists to test whether *VirusScan* is installed correctly. This relies on downloading a file from the *McAfee* website, having been directed automatically to the correct page by the *SecurityCenter*. Clearly this can operate only if the machine is connected to the Internet.

One notable feature of the *McAfee* web presence is the distribution of information over different sites – when installing and updating the corporate product, for example, it is necessary to visit, at the very minimum, the sites secure.nai.com and www.mcafeesecurity.com. This splitting of operations across sites also occurs in the informational aspect of the sites. If, when searching for virus information, one starts at www.mcafee.com, the journey will take one through www.avertlabs.com before arriving at the information on vil.nai.com.

This maze-like aspect of the websites is a shame, since the content of the informational parts is among the best available. The vil.nai.com site contains the main virus information library, together with numerous tools and





associated information sources. Also disappointing is the lack of true *Firefox* compatibility – several pages looked rather odd when viewed through *Firefox* rather than *IE*.

FEATURES

The main interfaces for the operation of *VirusScan* come through *McAfee SecurityCenter* and the default scan. Both are available from desktop links and from the start menu. The *SecurityCenter* is also available as a tray icon. The Start menu contains the options to create a rescue disk, manage quarantined files and the on-access scanner *ActiveShield*. In a departure from *McAfee* tradition, the on-access scanner does not have its own shield tray icon. Perhaps this is just as well, since these days red and yellow shields in the tray are signs that something untoward is afoot on the security front – imagine the panic that might be engendered by a red, white and silver shield.

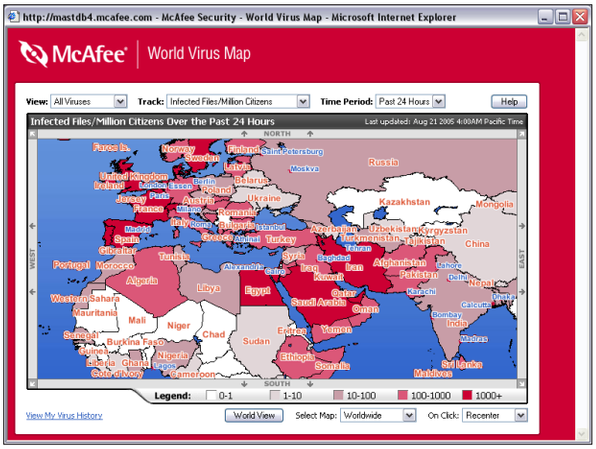
The *SecurityCenter* default view is 'my security center', which is dominated by the combination of five status bars, an advert for *McAfee Personal Firewall Plus* and the latest virus advisory. The activation of *Windows Update* here warrants an unobtrusive green blob on this page.

With *VirusScan* installed, only one of the status bars showed green, this being the AntiVirus Index. *Windows Firewall* on its own rated a medium level of protection in the AntiHacker Index, while the AntiAbuse and AntiSpam indices rated no protection. This came as little surprise, since the programs that had been offering protection in these areas had been removed, since they were part of a competing anti-virus product. Clicking on these indices produces information about any related problems and how they might be rectified. However, rather than providing unbiased or useful information, this feature seems to have been hijacked by *McAfee*'s marketing department as a method of displaying advertising copy for other *McAfee* products. In particular it surprised me to discover quite how much I needed to purchase *McAfee Spamkiller* when no email client was in use on the machine in question.

In addition to the 'my security center' tab, there are views for VirusScan, personal firewall plus, privacy service and spamkiller. Cynical readers will note that these are all the names of *McAfee* products rather than generic descriptions. *McAfee*'s reason for suggesting that *SecurityCenter* is installed and *Microsoft Security Center* is disposed of becomes altogether obvious at this point. Having resolved to slay another few marketing executives before breakfast I moved quickly on to the *VirusScan* view.

Unusually for an anti-virus product, the status of the product is given pride of place in this view, the same traffic-light symbol being used here as elsewhere in the interface. Below this are the actions available, each with a short description of its functionality. The most obvious option for inclusion here is scanning for viruses. The scanning also includes detection of what are termed 'PUPS' when detected, this derived from 'potentially unwanted programs'. One hopes that this designation is broad enough while not too damning of objects detected by *VirusScan*. In the current climate it is a short step from an overly harsh description of an application to the courtroom.

Within the scanning section the options are more clearly set out than in the corporate product, though they are fewer in number. Recursion of scanning, scanning of all files, compressed file scanning and heuristic activation are all supported. Together with the selection of scanning for PUPs, however, this is the full range of detection options on



offer. Location can be selected, of course, though there are no options here as to the action to take upon detecting malicious files.

For a greater variety of options a separate view is available from the VirusScan tab. This is not an area where significant changes may be instituted, however. Areas where on-access scanning will be performed are adjustable, as are the parameters for scheduled scans. No scheduled scan is activated by default. There is a preset scheduled scan which may be activated, however. This is timed to occur at 8pm on a Friday. This seems strange timing for a product aimed at home users, since one might expect home users to be more likely to be using their machines then than at many other times or on other days of the week – and while an on-demand scan is in operation other applications certainly feel the strain. Virus Map reporting may also be deactivated here, or set to a different reporting location.

The Virus Map is viewed through another of the parts of the VirusScan tab. The information contained here is very interesting, though it would be more so if there were finer granularity in the data reported. The location data as entered in the registration process distinguishes between US states, for example. However, the Virus Map indicates that the whole of the USA has the same level of virus detections – it would be useful to know whether this is because the data applies only to the US en masse, or whether each state does indeed have over 1,000 infected computers per million citizens.

The VirusScan tab provides a link to virus information. Strangely, this does not take the user to vil.nai.com but to a rather more rounded, pastel-coloured interface at us.mcafee.com. It seems that vil.nai.com is reserved for enterprise users rather than home users or small businesses, although the virus information on both sites looks, reassuringly, as if it has been pulled from the same database.

The remaining three areas which can be reached through the VirusScan tab are the Test VirusScan page, version details of *VirusScan* and the area where a rescue disk may be created. Incidentally, the file used to test the detection capability at this point is not the EICAR test file but an in-house *McAfee* creation, which is 24KB in size. The machine I was using did not possess a floppy drive, so the rescue disk functionality could not be tested. Since machines are increasingly likely to have a CD or DVD writer instead of a floppy drive, some method of burning a rescue CD would be a wise addition.

CONCLUSION

Although in the past *McAfee*'s various product lines have been increasingly inclusive of all possible security functions, the version on offer here seemed very much limited to anti-virus. This is not to say that the integration of extra products is not supported – quite the opposite in fact. It is very strange, however, to see the parts of the *McAfee Security Center* where these extras can be bolted on.

The provision of options for firewalls etc. in the *SecurityCenter* set me pondering. What happens if, for example, I wish to retain *McAfee*'s anti-virus product, but use a different vendor for each of anti-spam software, firewall and privacy protection? The *Microsoft Security Center* supports only two of these functions but currently does allow for a software firewall and anti-virus product from different vendors. Matters may become more complicated when *Microsoft*'s anti-spyware product finally arrives. Logic would, however, mark this as a perfect time for *Microsoft* to expand the *Security Center*'s coverage to include anti-spyware. This would be one more reason to reject the *McAfee* version and opt for *Microsoft* 's offering.

While reading through the comments I have made in this review, I notice that many of them point out foibles in the product that caused me to feel uneasy. Usually I am quite happy for the reader to share my disappointment with a product, but in this case my overall impression of *VirusScan* was less negative than my comments might suggest. The heart of *VirusScan* is well constructed and designed, and I certainly find it more user-friendly than the corporate versions.

Technical Details

Test environment: Athlon 64 3800+ with 1 MB RAM, 80 GB hard drive, CD/DVD ROM drive, 1Mbit ADSL connection, running *Windows XP Professional, Service Pack 2*.

Product: *McAfee VirusScan 10.0.25* Engine 4400 DAT 4563.

Developer: *McAfee Inc.*, 3965 Freedom Circle, Santa Clara, CA 95054, USA. Tel: +1 888 8478766; email: sales@mcafee.com, web: http://www.mcafee.com/.

END NOTES & NEWS

The IDC Security Conference takes place on 14 September 2005 in London, UK. Delegates will hear how other organisations have ensured the security of their business through the use of technology and security strategies. See http://www.idc.com/uk/security05/.

The Gartner IT Security Summit takes place 14–15 September 2005 in London, UK. The summit will look at how current technology provides guidance on which old and new product categories are most useful in controlling information security risk. For more information see http://www.gartnerinfo.com/.

T2'05, the second annual T2 conference, will be held 15–16 September 2005 in Helsinki, Finland. The conference focuses on newly emerging information security research. All presentations are technically oriented, practical and include demonstrations. See http://www.t2.fi/english/.

COSAC 2005, the 12th International Computer Security Symposium, takes place 18–22 September 2005 near Dublin, Ireland. A choice of more than 40 sessions and six full-day master classes and forums is available. The full programme and details of how to register are available at http://www.cosac.net/.

The Network Security Conference takes place 19–21 September 2005 in Las Vegas, NV, USA. The conference is designed to meet the education and training needs of the seasoned IS professional as well as the newcomer. For details see http://www.isaca.org/.

The 5th Annual FinSec Conference takes place 20–23 September 2005 in London, UK. This year's conference will focus on the unique set of challenges afflicting information security professionals in the financial community. See http://www.mistieurope.com/.

HITBSecConf2005 Malaysia will take place in Kuala Lumpur from 26–29 September 2005. The event will see six hands-on training classes and over 30 speakers who will present their research and findings. See http://conference.hackinthebox.org/.

The 4th annual SecurIT Summit will be held 28–30 September 2005 in Montreux, Switzerland. SecurIT 2005 will integrate a busy conference programme, one-to-one business meetings and informal networking with leisure activities. For more information see http://www.securit-summit.com/.

e-Secure Malaysia 2005 takes place 28 September to 1 October 2005 in Kuala Lumpur, Malaysia. The exhibition and conference will cover issues such as computer emergency response, spam and viruses, hacking, cyber laws and terrorism, security management, access control and network security. For full details see http://www.protemp.com.my/.

The SophosLabs Malware Analysis Workshop will be held 4 October 2005. The course is aimed at IT security professionals who are responsible for implementing and maintaining IT security solutions, or who are involved in computer security research. For details see http://www.sophos.com/.

The 15th Virus Bulletin International Conference, VB2005, will take place 5–7 October 2005 in Dublin, Ireland. The programme for the three-day conference can be found on the *VB* website. For more information or to register online see http://www.virusbtn.com/.

Black Hat Japan (Briefings only) will be held 17–18 October 2005. See http://www.blackhat.com/.

RSA Europe 2005 will be held 17–19 October 2005 in Vienna, Austria. For more details see http://www.rsaconference.com/.

WORM 2005 (the 3rd Workshop on Rapid Malcode) will take place 11 November 2005 in Fairfax, VA, USA. The workshop will provide a forum to bring together ideas, understanding and experiences bearing on the worm problem from a wide range of communities, including academia, industry and the government. For more details see http://www1.cs.columbia.edu/~angelos/worm05/.

The eighth Association of Anti-Virus Asia Researchers International Conference (AVAR 2005), takes place in Tianjin, China 17–18 November 2005. The theme of this year's conference will be 'Wired to Wireless, Hacker to Cybercriminal'. For details email avar2005@antivirus-china.org.cn or see http://aavar.org/.

SUBSCRIPTION RATES

Subscription price for 1 year (12 issues) including first-class/airmail delivery: £195 (US$358)

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

TRIAL AND RETRIBUTION

Former *AOL* employee Jason Smathers has been sentenced to 15 months imprisonment for selling customers' email details to spammers. The 25-year-old former software engineer pleaded guilty in February to stealing at least 92 million screen names from *AOL*'s database and selling the information to an associate (whose criminal charges are pending). Smathers was also ordered to pay $83,000 in compensation, but escaped the maximum 15-year prison sentence and $500,000 fine thanks to a plea bargaining agreement made earlier this year.

Meanwhile, *AOL* has been organising a sweepstake for its members with some very special prizes: assets it has seized from a 21-year-old New Hampshire spammer. The swag includes a luxury Hummer H2 vehicle, $75,000 in cash and $20,000 in gold bullion (see http://corp.aol.com/press/media_spammersloot.shtml for photographs). Last year the company made a similar very public demonstration of its tough stance against spammers when it raffled a Porsche Boxster which it had acquired as part of a settlement against yet another spammer. The company's message to would-be spammers reads: '*AOL* will find you and sue you. And *AOL* will do everything it can to make sure its members end up with any money you made as a spammer.'

EVENTS

TREC 2005, the Text Retrieval Conference, takes place 15–18 November 2005 at NIST in Gaithersburg, MD, USA. For more details see http://trec.nist.gov/.

The third Conference on Email and Anti-Spam, CEAS 2006, will be held in July 2006 in Silicon Valley, USA. Interested parties should subscribe to a low-volume mailing list for details of the event, see http://www.ceas.cc/mailinglist.htm.

PHISHING ANALYSIS

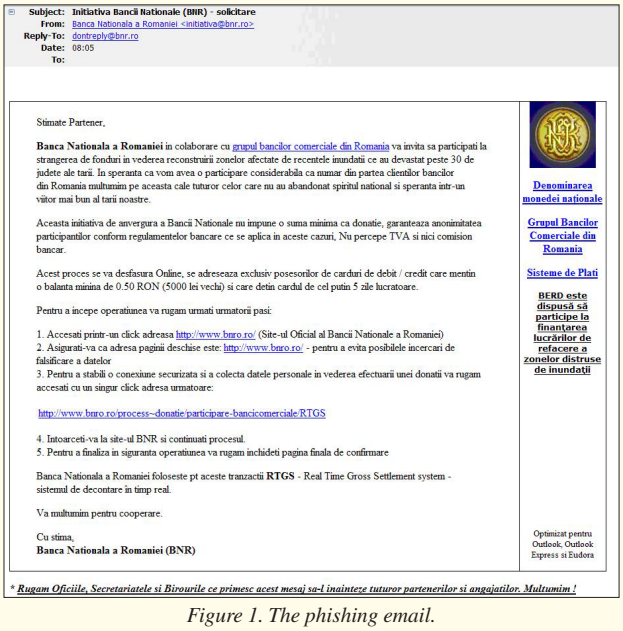
PHISHING COMES TO ROMANIA

Sorin Mustaca
AVIRA GmbH, Germany

Romania's first major phishing fraud was detected on the morning of 26 July 2005. Emails were sent from two sources; the first mail was detected at 07:25am and the second at 08:05am. The emails were designed to look like part of a fund-raising campaign initiated by the National Bank of Romania (BNR – Banca Nationala a Romaniei) to provide aid for the reconstruction of areas affected by severe flooding this summer (see Figure 1). Clearly, the authors of the message were planning to take advantage of the compassion shown by the Romanian public after the devastating floods, in order to gain credit card details and other personal information for identity theft.

The fake messages were made to look as if they had been sent by the National Bank of Romania, from the address initiativa@bnr.ro. The National Bank of Romania owns the domain www.bnr.ro (as well as www.bnro.ro) [1].

The authors of this phishing attack requested a sum of money that would be considered insignificant, even for



Subject: Initiativa Bancii Nationale (BNR) - solicitare
From: Banca Nationala a Romaniei <initiativa@bnr.ro>
Reply-To: dontreply@bnr.ro
Date: 08:05
To:

Stimate Partener,

Banca Nationala a Romaniei in colaborare cu grupul bancilor comerciale din Romania va invita sa participati la strangerea de fonduri in vederea reconstruirii zonelor afectate de recentele inundatii ce au devastat peste 30 de judete ale tarii. In speranta ca vom avea o participare considerabila ca numar din partea clientilor bancilor din Romania multumim pe aceasta cale tuturor celor care nu au abandonat spiritul national si speranta intr-un viitor mai bun al tarii noastre.

Aceasta initiativa de anvergura a Bancii Nationale nu impune o suma minima ca donatie, garanteaza anonimatatea participantilor conform regulamentelor bancare ce se aplica in aceste cazuri, Nu percepe TVA si nici comision bancar.

Acest proces se va desfasura Online, se adreseaza exclusiv posesorilor de carduri de debit / credit care mentin o balanta minima de 0.50 RON (5000 lei vechi) si care detin cardul de cel putin 5 zile lucratoare.

Pentru a incepe operatiunea va rugam urmati urmatorii pasi:

1. Accesati printr-un click adreasa http://www.bnro.ro/ (Site-ul Oficial al Bancii Nationale a Romaniei)
2. Asigurati-va ca adresa paginii deschise este: http://www.bnro.ro/ - pentru a evita posibilele incercari de falsificare a datelor
3. Pentru a stabili o conexiune securizata si a colecta datele personale in vederea efectuarii unei donatii va rugam accesati cu un singur click adresa urmatoare:

http://www.bnro.ro/process~donatie/participare-bancicomerciale/RTGS

4. Intoarceti-va la site-ul BNR si continuati procesul.
5. Pentru a finaliza in siguranta operatiunea va rugam inchideti pagina finala de confirmare

Banca Nationala a Romaniei foloseste pt aceste tranzactii **RTGS** - Real Time Gross Settlement system - sistemul de decontare in timp real.

Va multumim pentru cooperare.

Cu stima,
Banca Nationala a Romaniei (BNR)

Denominarea monedei nationale

Grupul Bancilor Comerciale din Romania

Sisteme de Plati

BERD este dispusă să participe la finantarea lucrărilor de refacere a zonelor distruse de inundatii

Optimizat pentru Outlook, Outlook Express si Eudora

** Rugam Oficiile, Secretariatele si Birourile ce primesc acest mesaj sa-l inainteze tuturor partenerilor si angajatilor. Multumim !*

Figure 1. The phishing email.

someone on a medium-to-low Romanian salary. Thus, the individual losses are relatively small, but the greater the number of victims, the greater the total amount. Of course, once the credit/debit card number has been obtained, there is nothing to prevent the perpetrators from taking as much money as they want (or as much as allowed by the limit set by the bank).

THE EMAILS

The same message (identical content) was sent from two different sources. The differences between them are as follows:

1. Subject. The first email had the subject: 'Initiativa Bancii Nationale a Romaniei (BNR) - colaborare', and was received at 07:25am. The second email had the subject 'Initiativa Bancii Nationale (BNR) - solicitare', and was received at 08:05am. As can easily be observed even by non-Romanian speakers, the only difference is the last word of the subject. The first one translates to 'collaboration' and the second to 'solicitation'.
2. Method of distribution. The first email was sent to a distribution list hosted by bcentral.com, a site owned by *Microsoft* (http://www.microsoft.com/smallbusiness/online/email-marketing/list-builder/detail.mspx). The interesting thing about this list is that in order to create it, you need to register and pay with a credit card. When the police investigate who created and paid for this list, they will probably find out that the card used for the payment was stolen. The second email was sent to individual email addresses by some web-based generator.

DETAILED ANALYSIS

As mentioned, the emails come from the address initiativa@bnr.ro, but if we look at the headers, the sender domain of the first one is listbuilder.com and the sender of the second is hostbigger.com.

Unfortunately, *SpamAssassin* does not detect anything strange about the mail:

```
X-Spam-Checker-Version: SpamAssassin 3.0.4-gr0 (2005-06-05)
[…]
X-Spam-Level: *
X-Spam-Status: No, score=1.5 required=3.0
tests=FORGED_RCVD_HELO,HTML_40_50,
HTML_MESSAGE,HTML_TAG_EXIST_TBODY,MIME_HTML_ONLY
autolearn=disabled
      version=3.0.4-gr0
```

How do we know that this is a phishing email? There are a number of reasons. The first indication that this is a phishing email is the forged link in the message. When looking inside the source of the mail, you see the following fake link, among others that are perfectly legitimate:

```
<a href="http://www.rnb.ro/process~donatie/
participare-bancicomerciale/RTGS">
<table>
<tr>
<td>
<a href="http://www.rnb.ro/process~donatie/
participare-bancicomerciale/RTGS" target="CONTINUT">
http://www.bnro.ro/process~donatie/participare-
bancicomerciale/RTGS
</td>
</tr>
</table>
</a>
```

This is what makes it a real phishing email – it displays a link to a legitimate website (a link which never existed):

```
http://www.bnro.ro/process~donatie/
participare-bancicomerciale/RTGS
```

and it goes to the illegitimate link:

```
http://www.rnb.ro/process~donatie/
participare-bancicomerciale/RTGS
```

where it asks for credit/debit card information.

The name of the domain RNB.RO was not chosen randomly. It is an anagram of the Romanian National Bank's true domain.

The technique used here is not new. However, you don't often see a table put inside a link and then an identical link inside that table. The intention was to make the link's active area as long as the width of the page, even if the link was half of it. However, there is a mistake. The author forgot to close the inner link with a ''. So the desired effect is not obtained. This feature (or bug?) is possible only in *Mozilla*-based html parsers. The same feature does not work, for example, in *IE*-based html parsers (with or without the closing '').

The second indication that this is a phishing email is that the illegitimate domain's registration is incomplete. If we look at the legitimate BNR.RO domain by querying the whois information database we see the following:

```
domain-name:          bnr.ro
description:          Banca Nationala a Romaniei
admin-contact:        TP1003-ROTLD
technical-contact:    TP1003-ROTLD
zone-contact:         TP1003-ROTLD
nameserver:           ns.bnr.ro 194.102.208.6
info: object maintained by ro.rnc local registry
info: Register your .ro domain names at www.rotld.ro
notify:               domain-admin@listserv.rnc.ro
object-maintained-by: ROTLD-MNT
mnt-lower:            ROTLD-MNT
updated:              hostmaster-danacorb@rotld.ro
19981214
source:               ROTLD
person:               Tiberiu Parvulescu
address:              Banca Nationala a Romaniei
```

```
address:              Str. Lipscani nr 25, sector 3
address:              Bucuresti
address:              Romania
phone:                +40-21-311 14 62
fax-no:               +40-21-311 14 62
e-mail:               tiberiup@nbr.ro
```

As you can see, this is a fully registered domain. All the identification data are present, and they are valid.

However, when you look at RNB.RO by querying the whois information database we see the following:

```
domain-name:          rnb.ro
description:          MobiFon S.A.
description:          Piata Charles de Gaulle, nr.15
description:          Sector 1
description:          Bucharest, Romania
description:          Phone: +40-21-302 4156
description:          Fax: +40-21-302 1475
admin-contact:        IOS1-ROTLD
technical-contact:    IOS1-ROTLD
zone-contact:         IOS1-ROTLD
nameserver:           ns7.dr.myx.net
nameserver:           dnsbck.dr.myx.net
info:                 Mugur Isopescu
info:                 Lipscani 25
info:
info:             cod fiscal / cod numeric personal:
info:                 Registered via xnet
info:    The NIC for Romania is http://www.rotld.ro/
notify:               domain-admin@listserv.rnc.ro
object-maintained-by: ROTLD-MNT
updated:        domain-admin@listserv.rnc.ro 20050722
source:               ROTLD
application-date:     20050722
domain-status:        active
registration-date:    20050722
expire-date:          20060722
```

You can see that some things are not quite right here. The identification information is present, but incomplete. The information about the owner is not only incomplete, but also fake:

```
info: Mugur Isopescu
info: Lipscani 25
```

The name of the owner seems to be some kind of joke. This is a combination of the name of the BNR Governor (Mugur Isarescu) and the name of a TV presenter (Emanuel Isopescu). The combination that results is a name that looks familiar to a lot of people: Mugur Isopescu. The address '*Lipscani 25*', even though incomplete, is the address of the BNR.

Apparently, the Romanian Internet authority (RNC) didn't even notice that something was wrong. However, to be fair, the RNC didn't register the domain directly; this was done by *Mobifon*. *Mobifon* owns MYX.NET, XNET and *Connex*, one of the biggest mobile phone networks in Romania (acquired recently by *Vodafone*).

The next reason for suspecting that this is a phishing email is the date of registration of the domain. The date of the creation of the domain was Friday 22 July 2005. It is well known in Romania that people browse the Internet more at the weekend than during the week. This is because in Romania many people use dialup connections and the telephone rates are cheaper during the weekend. Moreover, the registration of the domain took place just a couple of days before the attack began.

Finally, the target page gave a strong indication that this was part of a phishing scam. The website where the page was hosted was encoded with escape characters and the decoding, of course, took place only locally, in the browser, using some Java Script code. This way no web filter could detect anything strange like special keywords (see Figure 2).

```
<BODY><SCRIPT LANGUAGE="JavaScript"><!--
hp_d01(unescape(">`mf{%22ancqq? n3 %3C□□>vc`ng%22ancqq? n3 %22ukfvj
vmrq %22jgkejv? 14 %22tcnkel? vmr %22ukfvj? 5;6 %22qv{ng? `mpfgp/`mv
%3C□□>r%22qv{ng? vgzv/cnkel8%22aglvgp %3C@CLAC%22LCVKMLCN$cvknfg9%2
%22qv{ng? `mpfgp/`mvvmo/qv{ng8%22lmlg9%22`mpfgp/`mvvmo/ukfvj8%22ogfl
012 %22ukfvj? 514 %22qv{ng? `mpfgp/vmr/qv{ng8%22lmlg9%22`mpfgp/vmr/
```

Figure 2. Encoded content.

Even if the content was encrypted, the Java Script code was pretty well written. It validated the input fields such as: email, telephone, card information, name, etc. (see Figure 3).

```
<script Language="JavaScript">
function isEmailAddr(email)
{
  var result = false;
  var theStr = new String(email);
  var index = theStr.indexOf("@");
  if (index > 0)
  {
    var pindex = theStr.indexOf(".",index);
    if ((pindex > index+1) && (theStr.length > pindex+1))
        result = true;
  }
  return result;
}

function validRequired(formField,fieldLabel)
{
        var result = true;

        if (formField.value == "")
        {
                alert('Please enter a value for the "' + fieldLabel +'" field.');
                formField.focus();
                result = false;
        }

        return result;
}

function allDigits(str)
{
        return invalidCharSet(str,"0123456789");
}

function invalidCharSet(str,charset)
{
        var result = true;

        // Note: doesn't use regular expressions to avoid early Mac browser bugs
        for (var i=0;i<str.length;i++)
                if (charset.indexOf(str.substr(i,1))<0)
                {
                        result = false;
                        break;
                }
```

Figure 3. A lot of JS validation.

THE WEBSITE

By announcing an online fund-raising campaign for the flood victims, the attackers were able to target exclusively card owners who might have wanted to donate money, setting as the only condition a minimum amount in the respective bank account: 0.50 Romanian Lei (0.14 Euros). They provided a link that led to a forged page, which looked like a www.bnr.ro website page, where the potential donors could input their personal details and credit/debit card

numbers, as well as the donated amount, that supposedly would have been charged automatically from those accounts.

In order to make everything look exactly the same as the BNR webpages, the email used all links, except the target one, from the legitimate website (nothing special here, all phishing emails have the same structure).

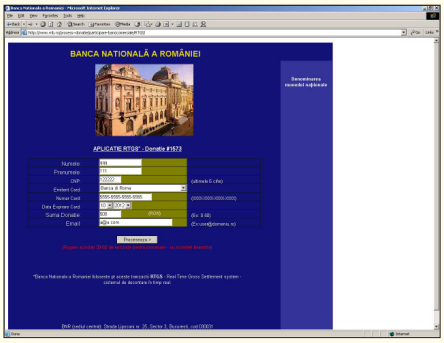


Figure 4. First page of the site.

Figure 4 shows the first page. Here, the victim is asked to provide their name, personal identification number taken from their identity card, the card issuer, the number of the card, expiry date of the card, the amount of money to be donated and the email address. There is nothing unusual here, except that debit cards are allowed as well as credit cards. This is a special adaptation for Romania, where a lot of people have debit cards, but not many have credit cards. Of course, for the debit card you need more information, so let's go to the next page.

On the second page, the user is asked for their PIN. Remember that nobody should know the PIN of your card except yourself. Finally, the victim is thanked for their donation and is even given an ID of their transaction. We only hope that the ID of our fake transaction (1572) was generated randomly, because if it relates to the number of transactions carried out, then more than 1,500 people will have a nasty surprise at the end of the month (apart from any who were playing with the website as we did).

THE EMAIL HEADERS

Here are other elements that prove that this mail is part of a phishing scam:

```
Return-Path: McCandle@mccandless.mozcal.org
```

This address exists and is hosted by hostbigger.com.

```
Return-Path: 55167-return-1-116905988@lb.bcentral.com
```

This address exists and is hosted by *Microsoft*.

The first email had the following text appended to the body:

```
'Powered by List Builder
Click <here> to change or remove your subscription'
```

where <here> is a link that goes to lb.lbcentral.com.

Following the link takes us to a page where we are asked:

```
'What would you like to do?
Your email address:
<email>
To unsubscribe from the mailing list, click the
Unsubscribe button.
If you wish to remain on the mailing list, but would
like to update your personal information
click the Change Preferences button.'
```

Clicking again on 'Unsubscribe' we get:

```
'Your email address and preferences have been removed
from the Banca Nationala a Romaniei mailing list as
you requested.'
```

They have actually created a list with the name of the Romanian National Bank.

CONCLUSIONS

After noting these messages, *AVIRA* proceeded to put an end to the fraud [2]. The director of *AVIRA Soft*, Mihai Anghel, contacted *Mobifon*, the Internet provider that had registered the rnb.ro domain (the destination of the forged link). Shortly after, through cooperation with *Mobifon* staff, the link was disabled and the respective domain suspended. He also advised the National Bank of Romania of the scam and they officially asked for a police investigation.

Unfortunately, and probably following an automatic procedure, *Mobifon* put the domain up for sale the very next day. *AVIRA* recognized that the risk of the scam starting all over again was still pretty high and decided to rent the domain for a couple of months, until the waters calm down (check here: http://www.rnb.ro/).

ACKNOWLEDGEMENTS

Many thanks to all the people who helped me in stopping the fraud. People from the National Bank of Romania, *Mobifon* and, most important of all, from the *AVIRA* team. Without their quick actions, the fraud would have continued longer. Also, thanks again to the *AVIRA* team members and to Costin Raiu who gave me precious comments when writing this article.

REFERENCES

[1] Romanian National Bank: http://www.bnro.ro/def_en.htm.

[2] *AVIRA*'s press releases about this incident (in English): http://avira.com/en/news/phishing_attack_in_romania.html and http://www.avira.ro/en/news/suppressing_major_phishing_attack.html.