

Malware Forensics: Detecting the Unknown

Martin Overton, IBM ISS, UK

Email: *overtonm@uk.ibm.com*

WWW: *http://www.ibm.com/uk*

Tel: *+44 (0) 2392 563442*

Abstract:

The increasing speed of new malware strains being written and released means that security professionals are more likely than ever before to see new malware.

This means new malware which is not detected by the anti-malware solutions they have deployed in their infrastructure, be it workstation, server, PDA or at the gateway.

Imagine this scenario: An end-user calls the helpdesk and reports that their system is running very sluggishly when it wasn't a week ago and that they can't access the Windows 'Task Manager' or open a command prompt any more.

Is this caused by malware or is it a 'user' problem? The virus scanner is right up to date and active, and it says the system is clean; the personal firewall is active too. Where do you go from here? Investigate or rebuild the box?

How can you tell if the machine is clean or infected by a new malware, with a reasonable level of confidence for your conclusion?

This paper will look at what tricks, tools and techniques you can use to help establish the true state of the 'suspect' system. It will focus on a step by step approach of what tools to use, what to look for and what to do with any suspicious files. It will also discuss the use of forensic tools in such a scenario, as a last port of call.

The paper will draw on real scenarios where new [undetected] malware has been responsible for 'odd' system or network behaviour.

Disclaimer:

Products or services mentioned in this paper are included for information only. Products and/or services listed, mentioned or referenced in any way do not constitute any form of recommendation or endorsement by IBM or the papers author.

This paper was presented at the 2008 Virus Bulletin conference at the Westin Hotel, Ottawa, Canada between October 1st – 3rd 2008.

Please note: This is an updated version of the paper written for EICAR 2008.

I would welcome any constructive feedback on this paper and its content.

1 Introduction

This paper will look at what tricks, tools and techniques you can use to help establish the true state of the 'suspect' system. It will focus on a step by step approach, including suggestions on what tools to use, what to look for and what to do with any suspicious files. It will also discuss the use of forensic tools in such a scenario, as a last port of call.

The paper will draw on real scenarios where new [undetected] malware has been responsible for 'odd' system or network behaviour.

Before we start let us cover a few definitions so that we all know what I mean by the relevant terms used in this paper.

I would strongly suggest that unless you have in-depth knowledge of malcode and related security threats that you try and obtain copies of the books/papers/articles listed in Appendix A.

1.1 What is Malware?

I will use the following definition which originally appeared in my Virus Bulletin 2005 paper: *Bots and Botnets: Risks, Issues and Prevention*.

“Malware is the generic name [or short name] used to describe Malicious Software. This includes viruses, worms, Trojans, bots and related threats.

In the ‘old-days’ [1980s and early 1990s] malware took a long time to spread widely, typically months. However, once the internet and networks became ubiquitous they started to spread wide and far more quickly, typically weeks. Malware that spread via e-mail took the next step, spreading widely in days or less than a day. Then along came the likes of CodeRed, Blaster and Slammer which could be widespread in hours. In Slammer’s case 90 percent of vulnerable systems were infected in under 10 minutes [mainly because it used UDP instead of TCP and could in theory have fired off 30,000 scans per second on a 100Mbps network. In reality however Slammer averaged around 4,000 scans per second per infected system].

The almost instantaneous appearance of new mass-mailing worms in all geographic areas of the World has been blamed on the use of botnets as launch points. Imagine a botnet of 10,000 plus systems that are ordered to spam a new mass-mailer [or Trojan] out to the world, or even to infect themselves to effectively kick-start the infection.

For example, the Witty worm was reported to have been launched from a small bot net of around 4,200 zombies. This allowed it to virtually appear almost instantaneously all over the world at the same time and to start searching for new victims to infect/attack.

It has been widely suspected that many of the recent most successful mass-mailing worms have used botnets to enable faster initial world-wide distribution, effectively giving the worm a head start. These include: MyDoom, Netsky and Bagle amongst others.”

2 Discussion

This section of the paper will discuss ways to try and decide whether a system is infected or not by a new [or currently unknown] malware which your current anti-malware defences do not detect.

This can not be done with complete accuracy [although you can get pretty close] due to the complexity of computer operating systems and also a fair proportion of modern malware itself.

To give ourselves the best chance of achieving the goal of proving [beyond reasonable doubt] that a suspected system is simply faulty [hardware/software fault] or actually infected by one or more malwares, we will offer advice on what evidence to gather from existing tools on the system and the network it is attached to. Finally we will then discuss other tools, techniques and tricks you can use to help you find and eliminate any malwares found on the suspected system being analysed.

Firstly, we will briefly look at the changes in malware itself over the years, so that you can understand what you are up against.

2.1 The problem

To save time and having to effectively repeat myself, I will use the following part of the conclusions from my Virus Bulletin 2007 paper: *The Journey, So Far: Trends, Graphs and Statistics*.

“It has been an interesting journey, since the start of the problem with malware on the IBM PC and compatibles in 1986 with Brain.

We can see the following trends since those first tentative steps:

1986 until early nineties they were the almost exclusive domain of the DOS COM, EXE file infectors and boot viruses. They became more complex and stealthy as the years passed. We also saw viruses that would attack or disable anti-virus defences. Mostly the motivations for these creations were, in the early days, curiosity and research; later it became the electronic equivalent of graffiti, vandalism or bullying. Occasionally it would be used to get a message across, be it personal or political.

From 1995-2000 Macro viruses were King, slowly spreading at first, as people exchanged infected .doc/.xls files via floppy, CD or e-mail. Later examples would be able to propagate via e-mail by reading the Outlook or Windows address book, but only after a recipient had opened the infected attachment. Mostly the motivations for these later creations were the electronic equivalent of graffiti, vandalism or bullying. Occasionally it would be used to get a message across, be it personal or political. There were less likely to be motivated by research.

2000-2003 saw Script viruses steal the crown from Macro viruses, and we also started to see 32 bit PE files becoming dominant; multi-component malware started to appear. A large proportion of malware started to use vulnerabilities in both the OS and applications. The motivations for this period were almost the same as for those between 1995 and 2000.

2004 to the start of 2005, the mass-mailing worms were the Kings; resulting in many overloaded mail servers and worn-out anti-virus researchers and corporate security staff. However, in most cases the motivations were the same as before, although the shift towards seeing malware as a business tool had already started. Social-engineering was becoming more widely used.

2005-2007 and the new Kings, were BOTs, Trojans and Spyware. Phishing grew from almost nowhere to one of the biggest security risks, aside from malware. The motivations for writing malware changed dramatically from the start of 2005. Money was the main motivational driver, and this would grow as organised crime got into the act, and slowly took over. Many malware authors were regularly trying to disguise their creations using packers and compressor, such as UPX, ACE, PEX, etc. The use of social-engineering was very noticeable and by 2007 it had become almost the most common method used by malware authors to get their creations onto a computer, aside from using vulnerabilities.”

So, in summary what we have seen is not only the birth of malware on the IBM PC in 1986 but also the birth of Stealth malware too. Since then we have seen increasingly complex malware [as well as a lot of very mundane and simple ones]. The techniques used over the years include:

- *File infection [and not just .COM and .EXE].*
- *Boot Sector [MBR and DBR] infection.*
- *Stealth and it's rebirth as Windows Rootkits.*
- *Polymorphism and it relatives [including server-side].*
- *Macro and Script malware.*
- *Entry Point Obfuscation [EPO]*
- *Cavity, Link, Prepending, Appending, Companion, Sparse infectors.*
- *Trojans, Worms and Bots.*
- *Spyware and Adware.*
- *Packers and Compressors.*
- *BHOs, LSPs, Fake Codecs and Plugins*
- *Packet and Keyword Filters.*
- *Data-diddling and Encryption.*
- *File, Directory or Operating System damage or removal [including formatting drives]*
- *Resident or Direct infection.*
- *Exploit code and vulnerabilities.*
- *Anti-malware detection and removal.*
- *Personal Firewall detection and removal.*
- *Other malware detection and removal.*
- *Virtual Machines [running inside and detection of].*
- *Social Engineering.*

The above list is not complete, it is there to give you a flavour of the many techniques; infection, damage, protection and hiding, that have been used since the dawn of IBM PC malware.

At the time of writing this paper there were over 383,000 known malware strains¹

2.1.1 Speed of infection/infestation?

How long can an unprotected PC last on the Internet before it gets infected/infested?

Well, according to the latest data from SOPHOS, just 720 seconds!

Here's a quote from them which was used in an article on The Register² in 2005:

"More computer viruses and worms mean an unprotected Windows PC (without either firewall or antivirus protection) stands a 50 per cent chance of infection by a worm after just 12 minutes online. Graham Cluley, senior technology consultant at Sophos, conceded"

Are you surprised just how quickly your PC(s) can get infected? Well, you shouldn't be!
Am I surprised at this?

No, not at all, in fact I've seen systems infected even faster than this by more than one malware strain.

So, as the average Windows PC [once unboxed and connected to the internet] has a life expectancy of around 10 minutes before getting a digital dose of the Pox, and sometimes more than one strain to boot!

¹ Source: McAfee

² Source: http://www.theregister.co.uk/2005/07/01/sophos_1h05_malware_report

I know of systems that have had 6 different doses of different digital Pox [Malware] in less than an hour and that's on a slow day!

The above assumes that the PC does not have a firewall installed and/or enabled, no anti-malware tools installed, or it isn't up-to-date and/or enabled for on-access scanning.

That was back in 2005, the situation in 2008 is significantly worse due to the commercialisation of malware, which is mainly due to cyber-criminals and their ilk.

2.2 Solutions

Right hopefully by the time you have reached this point of the paper, you now understand the threats, infection vectors used, techniques employed and the speed of infection? If not, then if you haven't already I'd strongly suggest that you go and read my papers from *EICAR 2005* and *2006* and *Virus Bulletin 2005-2007* as otherwise you might not get the most out of this section of the paper.

So, if you are ready, let me begin by covering the first steps of the process to try and determine if a system is infected or just faulty. I will mention tools as we proceed, but I will not cover them in any detail at that point. For details on a specific tool please see the *Tools* section of the paper which included links to the homepages, or the download page for that tool or product.

2.2.1 Step 1: Identifying Suspect Systems

The first thing to do is to understand that you have a problem; the next thing to do is to try and identify possible systems that may be infected.

This information can come from help-desk tickets [personal firewall or anti-malware alerts, strange system behaviour, etc], Log files from your routers, proxies, firewalls, IDS/IPS systems, DNS and so on, or maybe even just a passing comment from a colleague or even a customer or other third party [maybe to your abuse@yourdomain.com e-mail address].

Once you have a potential suspect, gather all the data you can from it and network traffic to and from it, including all ports and protocols used as this may help to narrow down your search. At this point you should consider removing the suspected system from your network until your investigation is completed [this helps to minimise the chance of further infections, data loss, and so on]. Also check for hidden files using ADS [Alternate Data Streams] as this technique is increasingly being used by malware authors to hide their malcode on the infected system.

Once the machine has been removed from the main network, you can either investigate it in isolation or move it to a test [secure] network used for analysing suspected infected systems.

To analyse suspected traffic on your test network you could use tools such as SNORT, WireShark or WinDump [you may also need to install WinPCap first, unless you are using *NIX or a Mac] or one of the many other IDS/IPS or packet/protocol analysers that exist.

You may also decide to carry out some vulnerability assessment of the suspected system; this can be done via tools such as Nmap, Superscan, Nessus or the Microsoft Baseline Security Analyzer.

2.2.2 Step 2: Analyse the Data (Part 1)

At this point you may already be able to state with some level of confidence that the system is infected by a malcode which *phones-home*. Examples of these include bot clients, or a Trojan or multi-component malcode [such as a dropper] that has contacted one or more websites to download other malcode or adware to install. This act, in many cases effectively starts a chain reaction leading to a heavily infected system with tens or hundreds of malcode files [or components] installed.

In either case, you could, visit the websites, FTP sites or IRC channels used to gather more information or even a *fresh* sample [or samples, scripts, etc.] of what you are fighting. This will help in your remediation, as well as allowing you to supply your anti-malware vendor with something to analyse, which in turn could end up making remediation [or at least detection] easier.

2.2.3 Step 3a: Scan the System

Scan with up-to-date anti-malware tools [anti-virus, anti-spyware, anti-rootkit, etc.] and see if anything is identified, ensure that heuristics and generic detection features are enabled. Preferably you should use at least two different products from each category, after all the anti-malware solution you have deployed didn't detect it, did it?

Try clean-booting if performing a *live* system scan fails [or if a Windows system try booting into *Safe Mode* first] to find anything. Clean booting will ensure that any active malware or related processes are not active. You can use BartPE or a Live Linux CD/DVD to do this and either include your scanning tools on the disc or a USB flash drive instead.

Any files identified as malware or flagged as suspicious [assuming you have remembered to enable heuristics and/or other generic/behavioural features of the scanners], should be copied to a USB flash drive or other removable media and labelled as potential malware to minimise the chances of anyone accidentally executing the files on another system.

As with *Step 2*, if you now have some suspected files, send them to your anti-malware vendor for analysis, however, this does not stop you analysing the files yourself [assuming you have the relevant skills and tools and have been given permission from your security manager/director to do so]. Place suspect files into a password protected zip file [use the password of *infected*] and send them to your preferred anti-malware company.

You could also send any samples to scanning services, such as VirusTotal and Jotti, and also to sandboxes such as the one run by Norman, or the CWSandbox [also available via Sunbelt]. Some of these services will analyse the files in great depth and supply you with copious amounts of useful data. This can help you to understand what the files are doing, and therefore how to remediate any affected systems, even before your anti-malware vendor has detection.

You can see the amount of data that some of these tools and services produce in 'Real World Example 2' later in this paper.

2.2.4 Step 3b: D-I-Y Sample Analysis

Assuming you have the relevant skills and tools and have been given permission from your security manager/director to do so, you could analyse the files yourself.

I would recommend that this is done on a system that is not connected to the network, and ideally this is a system that you will either use VMWare [or some other Virtual Machine software] on, so that it can be re-imaged, or reset back to a clean image [snapshot] after running the suspected files on the test system.

If you are using a Virtual Machine such as VMWare then you need to be aware that the malware may be able to detect that it is in a virtual machine and either change its behaviour accordingly or turn destructive and kill your virtual machine.

The malware covered in 'Real World Example 1' appeared to be able to detect it was being executed inside at least one of the most commonly used Sandboxes.

Once this has been setup, you can use whatever tools you prefer to carry out the analysis, such as, using static analysis tools, like PEiD, Strings, File Alyzer and so on, you could also examine the file in a hex editor and/or a debugger. This is only advised if you are able to understand assembler code and you are sure that the file to be debugged does not contain anti-debugging code which may be triggered during examination.

You could then move onto running the file and seeing what it does, using tools such as InCtrl5, Windiff, PSTools or you may prefer to disassemble it using tools such as IDAPro, WinDbg or OllyDbg. This is only advised if you are able to understand assembler code and you are sure that the file to be debugged does not contain anti-debugging code which may be triggered during examination.

This is also a good time to try out any remediation scripts or tools you have created as a quick-n-dirty solution to the problem [obviously only on a test system].

2.2.5 Step 4: Analyse the Data (Part 2)

By now you should have a good idea what is going on, and what any malware is doing to the affected systems and what network traffic is being generated by it [or them].

If you haven't then you should now take time to go over all the data you have acquired during the first three steps. You could use a flow diagram to plot the malware's features and activities, or you may prefer to brainstorm on a whiteboard with suitable colleagues. From here you should emerge with a clear [or fairly clear] understanding of what needs to be done to protect the rest of the network [it could be as simple as putting in a new, or changing an existing router ACL, firewall rule, or IDS/IPS signature/rule in place] which may also allow you to identify other infected systems that need to be removed from the network and remediated.

2.2.6 Step 5: Remediation

Hopefully by now, you can either create or at least plan out the steps that you need to take to remediate all the infected systems identified. You may decide that you can create your own clean-up scripts [paper and/or code] rather than wait for your anti-malware vendors to get detection and cleanup definitions [signatures] to you. Otherwise you will have to be patient until your anti-malware vendor delivers the goods.

The other alternative, especially if a system is heavily infected, or you can't find any sign of malware [even when using all the tools/tricks and techniques listed in this paper], is to restore the system from the last known clean backup, or re-image it to your organisation's standard desktop/server build image.

2.2.7 Step 6: Post Mortem

This is where you take stock of what has happened and decide what [if any] changes are required to improve protection of your infrastructure, your security policy and procedures and, last but not least, user education.

The whole point of this is to help minimise the risk of another similar outbreak. The ideas that come out from this session should be wide-ranging and generic as these will generally offer the best improvements in your organisation's security posture; both from the aspects of prevention and incident management.

This is not the time for a witch-hunt to take place so that blame can be attributed to individuals and/or teams, you should focus on what went wrong [or failed] and put together solutions to minimise the chances of a similar attack being successful next time. It may also be useful to revisit your overall approach to threats and infection vectors, as they may have changed since the last time you looked.

***A final note:** If it is a criminal case then you need to follow computer forensic principals, such as the chain of custody, and follow the prevailing laws [including all guidance from law enforcement agencies that might get involved] for your country, state, or other geographical divide. Failure to do so may mean that a successful prosecution is unlikely; the case may not even get to court. If in doubt seek legal guidance first, before proceeding.*

3 Tools

A common problem when you think you have a rogue [malware/spyware/adware] program running on your system is trying to find it.

This section of the paper will cover a number of tools which can be useful in checking a system out for odd behaviour and for testing/analysing suspected files. I will not cover all of these in depth as that is beyond the scope and purpose of this paper, in the cases where I do not cover a tool in depth I will use some of the description text found on the website of that specific tool or application instead.

However, I will cover some of the most useful diagnostic tools in more than passing where I can. These tools are suggested to be used where you have already carried out some investigation, such as you have already scanned the suspect system with at least one 'up-to-date' anti-virus product, at least one 'up-to-date' anti-spyware product and at least one 'up-to-date' anti-rootkit product.

The final option is to use forensic tools, such as: Encase or F.I.R.E.

3.1 Remote Access

These are very useful tools for when you can't physically get to a suspected system as it is in another building, country or a secure facility.

3.1.1 VNC

“VNC stands for Virtual Network Computing. It is remote control software which allows you to view and fully interact with one computer desktop (the "VNC server") using a simple program (the "VNC viewer") on another computer desktop anywhere on the Internet. The two computers don't even have to be the same type, so for example you can use VNC to view a Windows Vista desktop at the office on a Linux or Mac computer at home. For ultimate simplicity, there is even a Java viewer, so that any desktop can be controlled remotely from within a browser without having to install software.”

Website: <http://www.realvnc.com/>

3.1.2 PSEXEC

“Utilities like Telnet and remote control programs like Symantec's PC Anywhere let you execute programs on remote systems, but they can be a pain to set up and require that you install client software on the remote systems that you wish to access. PsExec is a light-weight telnet-replacement that lets you execute processes on other systems, complete with full interactivity for console applications, without having to manually install client software. PsExec's most powerful uses include launching interactive command-prompts on remote systems and remote-enabling tools like IpConfig that otherwise do not have the ability to show information about remote systems.”

Website: <http://www.microsoft.com/technet/sysinternals/security/psexec.mspx>

3.2 File Information

These tools are very useful in analysing a file, its structure and may often tell you if the file is packed, compressed, what resources it requires, exports as well as often showing the internal file format [hex viewer] and often even text strings found in the code. Others covered in this section will show you network connections [including which file or process is responsible for it].I will also include debuggers and dissemblers in this section.

3.2.1 PEiD

“PEiD detects most common packers, cryptors and compilers for PE files. It can currently detect more than 600 different signatures in PE files.”

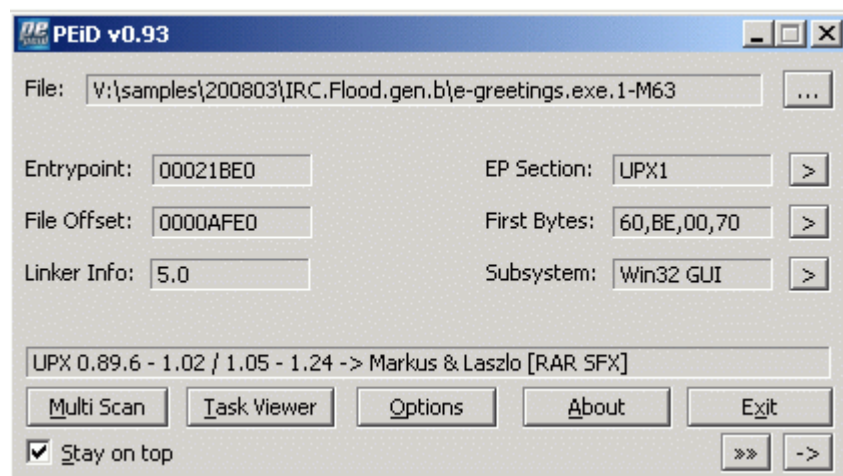


Figure 1: PEiD screenshot

Website: <http://www.peid.info/>

3.2.2 FileAlyzer

“FileAlyzer is a tool to analyze files - the name itself was initially just a typo of FileAnalyzer, but after a few days I decided to keep it. FileAlyzer allows a basic analysis of files (showing file properties and file contents in hex dump form) and is able to interpret common file contents like resources structures (like text, graphics, HTML, media and PE).”

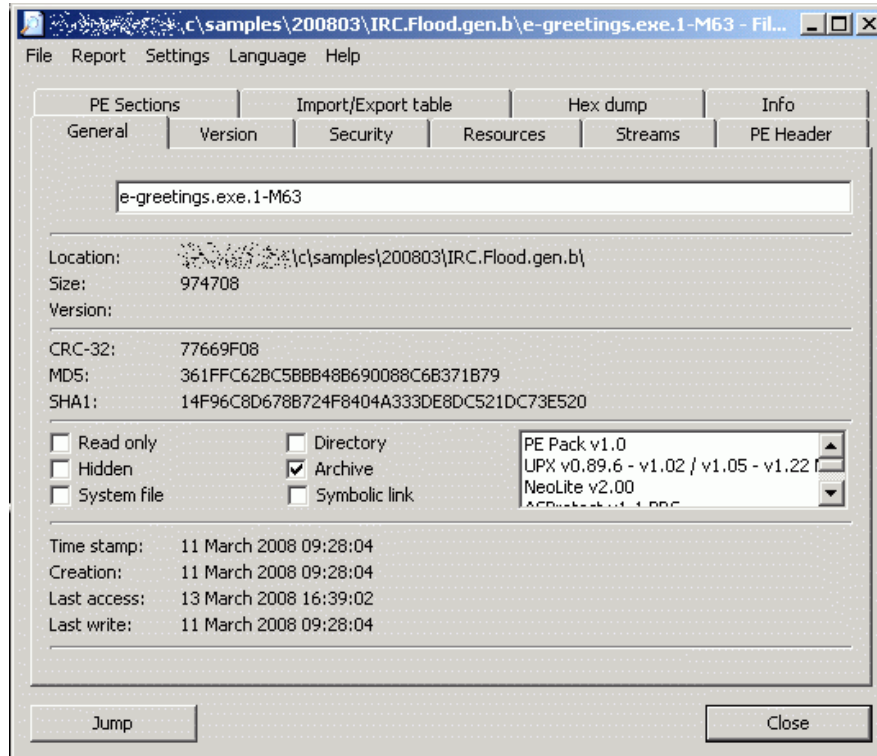


Figure 2: FileAlyzer screenshot

Website: <http://www.safer-networking.org/en/filealyzer/index.html>

3.2.3 Stud_Pe

“Stud_Pe The Portable Executables Viewer/Editor, view/edit PE basic Header information (DOS also):
 -header structures to hexeditor; view/edit Section Table: - add new section; view/edit Directory Table: -
 Import/Export Table viewer; -Import adder; -Resource viewer/editor save/replace ico/cur/bmp); Pe
 Scanner (PEid sig database): -400 packers/protectors/compiler; Task viewer/dumper/killer;
 PEHeader/Binary file compare; RVA to RAW to RVA; Drag'nDrop shell menu integration; Basic
 HexEditor;”

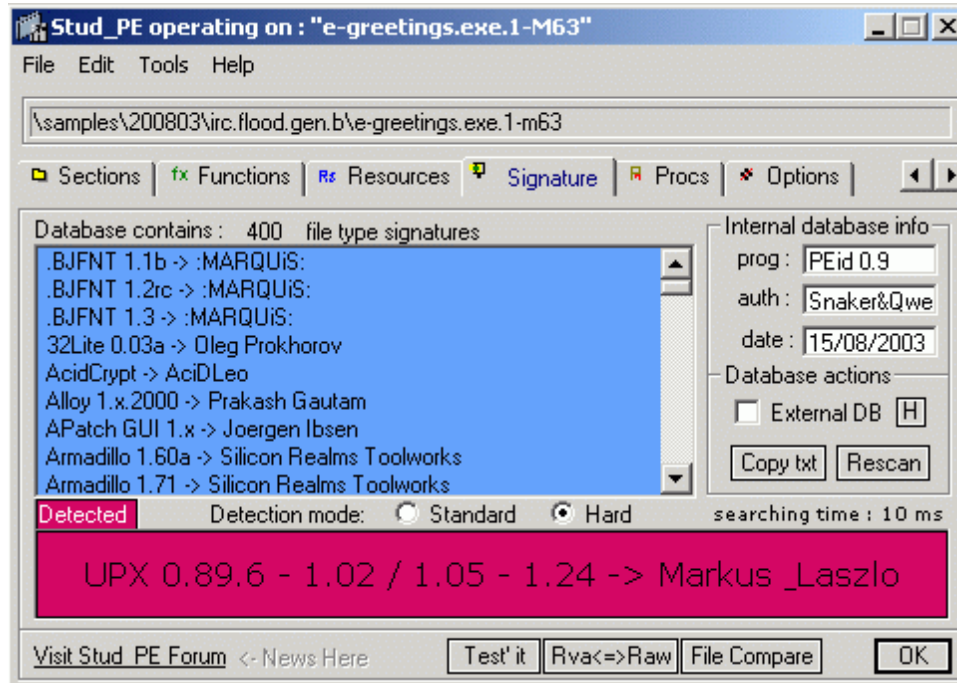


Figure 3: Stud_PE screenshot

Website: <http://www.cgsoftlabs.ro/studpe.html>

3.2.4 Strings

“Working on NT and Win2K means that executables and object files will many times have embedded UNICODE strings that you cannot easily see with a standard ASCII strings or grep programs. So we decided to roll our own. Strings just scans the file you pass it for UNICODE (or ASCII) strings of a default length of 3 or more UNICODE (or ASCII) characters. Note that it works under Windows 95 as well.”

Website: <http://www.microsoft.com/technet/sysinternals/Miscellaneous/Strings.msp>

3.2.5 WinDbg

“You can use Debugging Tools for Windows to debug drivers, applications, and services on systems running Windows NT 4.0, Windows 2000, Windows XP, Windows Server 2003, Windows Vista, and Windows Server 2008 as well as for debugging the operating system itself. Versions of the Debugging Tools for Windows package are available for 32-bit x86, native Intel Itanium, and native x64 platforms.”

Website: <http://www.microsoft.com/whdc/DevTools/Debugging/default.msp>

3.2.6 OllyDbg

“OllyDbg is a 32-bit assembler level analysing debugger for Microsoft® Windows®. Emphasis on binary code analysis makes it particularly useful in cases where source is unavailable.”

The screenshot shows the OllyDbg interface for the process e-greetings.exe. The main window displays assembly code for the main thread in module e-greeti. The registers window shows the current state of the CPU registers, with EIP pointing to 00421BE0. The memory dump window shows a hex dump of memory starting at address 00422000, with ASCII characters visible below it. The status bar at the bottom indicates 'Analysing e-greeti: 0 heuristical procedures' and the debugger is in a 'Paused' state.

Figure 4: OllyDbg screenshot

Website: <http://www.ollydbg.de/>

3.2.7 IDA pro

“IDA Pro is a Windows or Linux hosted multi-processor disassembler and debugger that offers so many features it is hard to describe them all.”

Website: <http://www.hex-rays.com/idapro/>

3.2.8 Fport

This is one of the 'tools-of-the-trade' that can be used to identify open and listening ports that are being used by the 'scumware' to talk/listen to the internet.

Most network technicians will normally first suggest that you use the ubiquitous 'Netstat' command found on all Windows and Linux systems.

Netstat when run with the '-a' switch will show all the active and listening ports in use on the TCP/IP stack, which is useful as long as you know what the all port numbers mean!

Here is an excerpt from the output of Netstat -a:

```

D:\Utils>netstat -a

Active Connections

Proto Local Address          Foreign Address        State
TCP   ACER-Sempron:echo      ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:discard  ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:daytime  ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:gotd     ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:chargen  ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:http     ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:pop3     ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:sunrpc   ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:epmap    ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:imap     ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:microsoft-ds ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:990      ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:1030     ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:1044     ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:8080     ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:9999     ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:44334    ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:44501    ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:nethios-ssn ACER-Sempron:0        LISTENING
TCP   ACER-Sempron:microsoft-ds :3187                 ESTABLISHED

```

Figure 5: Netstat -a screenshot

To understand it you really need to understand networking to a reasonable level, this includes the different protocols, all the port numbers used by common applications and also how to get the output for UDP as well as TCP ports. This is a bit of a minefield for non-technical users!

What if you want to find out which application/program/executable is actually using a specific port [or range of ports]? Well, in that case Netstat can't help, however there is a simple little tool that can give you just that information and can be very, very, useful in helping to diagnose the presence of a new piece of network-enabled 'scumware'; this tool is Fport.

Introduction:

Fport is a free tool that will show you what programs on your system are opening which ports (both TCP and UDP). You can look at the output and see if you notice any strange programs that don't belong on the machine. Then you can use a command-line "kill" utility such as PSKill to stop the programs. Typically, trojans and some viruses will open up non-standard ports which can be a great clue to determining if a system is compromised or not. Watch out for open high numbered ports such as 3112, 31337, 12345, and 65000. Fport can be used on Windows NT4, Windows 2000, and Windows XP.

Installation:

Place the Fport.exe file directly on your C drive. Fport works only if you navigate to where it is being stored in the command prompt.

Usage:

Once installed, invoke fport like this:

```

Start --> Run --> cmd
C:\> cd \
C:\> fport -p

```

If you want to pipe the output of fport into a file:

```

C:\> fport -p >> [filename].txt

```

You can download Fport from: <http://www.foundstone.com/us/resources/proddesc/fport.htm>

The beauty of Fport is that it is very useable by even the most non-technical of users; it is small and currently is not being defeated/manipulated by malware, unlike a number of other system diagnostic tools. So, if you think you are infected and have tried all the usual things to track down the rogue application, then give Fport a go.

3.2.9 Handle

“Ever wondered which program has a particular file or directory open? Now you can find out. Handle is a utility that displays information about open handles for any process in the system. You can use it to see the programs that have a file open, or to see the object types and names of all the handles of a program.”

Website: <http://technet.microsoft.com/en-us/sysinternals/bb896655.aspx>

3.2.10 Netstat

See FPort

3.3 System Information

This section will cover tools that are generally considered to be vulnerability analysis tools and tools that can be used to help pinpoint rogue entries in key system areas, such as registry keys, services, browser helper objects and other plugins, DNS, LSP and other networking modifications and settings. Several of these tools could also be included in the previous section, as they are multi-purpose.

3.3.1 Nessus

“Nessus performs sophisticated remote scans and audits of UNIX, Windows, and network infrastructures. Nessus discovers network devices and identifies the operating systems, applications, databases, and services running on those assets.

Any non-compliant hosts, such as systems running P2P, spyware, or malware (worms, Trojans, etc.) are detected and identified. Nessus is capable of scanning all ports on every device and issue remediation strategy suggestions as required.”

Website: <http://www.nessus.org/nessus/>

3.3.2 Microsoft Baseline Security Analyzer

“Microsoft Baseline Security Analyzer (MBSA) is an easy-to-use tool designed for the IT professional that helps small- and medium-sized businesses determine their security state in accordance with Microsoft security recommendations and offers specific remediation guidance.”

Website: <http://www.microsoft.com/technet/security/tools/mbsahome.mspx>

3.3.3 SuperScan

SuperScan is a Windows GUI alternative to using NMap, useful when you can't get hold of NMap or don't know how to use it. The tool is described by the creator, as a:

“Powerful TCP port scanner, pinger, resolver.”

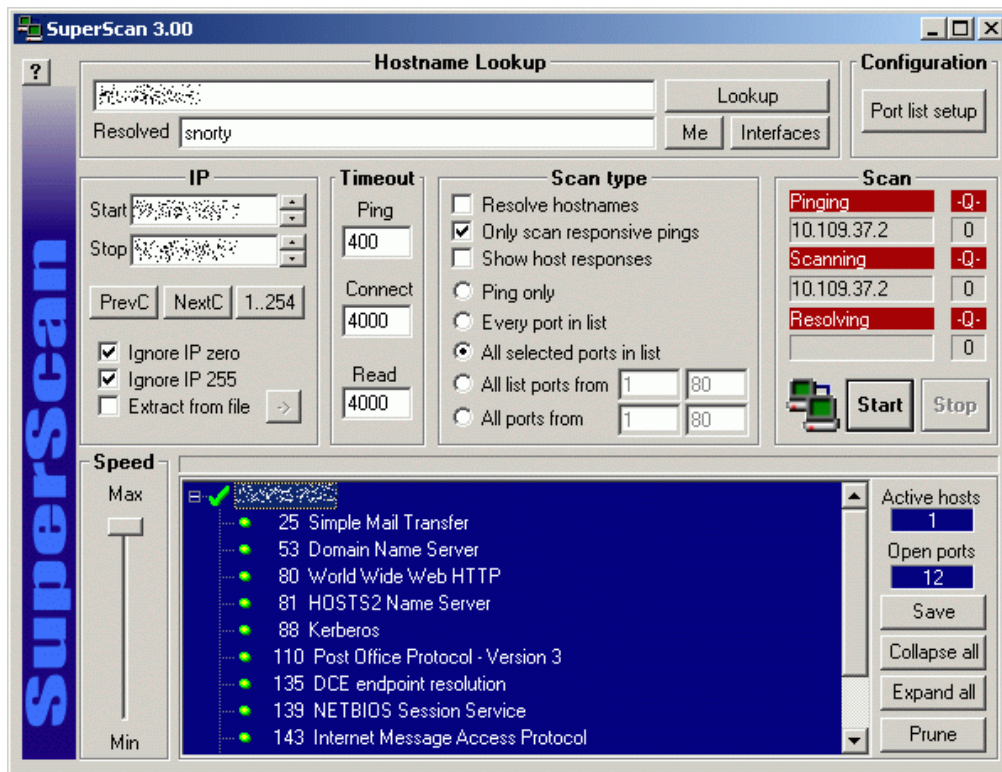


Figure 6: SuperScan screenshot

Website: <http://www.foundstone.com/us/resources/proddesc/superscan4.htm>

3.3.4 Nmap

“Nmap ("Network Mapper") is a free and open source (license) utility for network exploration or security auditing. Many systems and network administrators also find it useful for tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime. Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics. It was designed to rapidly scan large networks, but works fine against single hosts. Nmap runs on all major computer operating systems, and both console and graphical versions are available.”

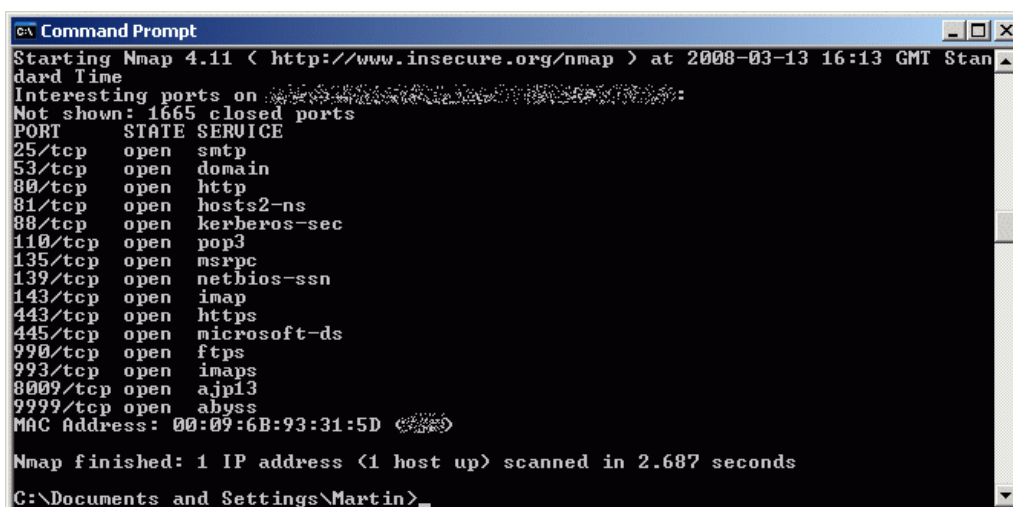


Figure 7: Nmap screenshot

Website: <http://nmap.org/>

3.3.5 HijackThis

This is another useful tool for finding spyware, adware and other malware programs running on your system via one of the registry keys which ensures that the 'scumware' is running whenever it wants to; such as at system startup or when a specific application is launched.

To try and assist in this situation I will cover one of the 'tools-of-the-trade' that can be used to list registry keys and related launch points that are being used by the 'scumware' when it gets on to your system.

Introduction:

HijackThis examines certain key areas of the Registry and Hard Drive and lists their contents and provides the ability to remove any unwanted stuff.. These areas are used by both legitimate applications and hijackers.

This is how the author describes it:

“A general homepage hijackers detector and remover. Initially based on the article Hijacked!, but expanded with almost a dozen other checks against hijacker tricks. It is continually updated to detect and remove new hijacks. It does not target specific programs/URLs, just the methods used by hijackers to force you onto their sites. As a result, false positives are imminent and unless you are sure what you're doing, you should always consult with knowledgeable folks (e.g. the forums) before deleting anything.”

Installation:

Download the HijackThis zip file to your computer and unzip it. I would recommend first creating a folder named 'HijackThis' for it located someplace easy to find like 'My Documents' and place the file into the same folder.

Now to make opening the program simple create a shortcut to the desktop. This is done easiest by right clicking on the HijackThis exe file, scroll down to 'Send To', and scroll across to 'Desktop (create shortcut)' and click it.

Usage:

Now open the program and click 'Scan'. When the scan is done click 'Save log' and save the log file to the same folder HijackThis is in. Please do not check or fix anything.

Open the log file. Double-clicking on the file should open the log file with notepad or similar text editor. If asked to choose a program to open it with select Notepad. Using Notepad click 'Edit', scroll down to 'Select All' to highlight all the text in the file. Click 'Edit', scroll down to 'Copy' and click.

Website: <http://www.spywareinfo.com/~merijn/programs.php>

So, what does it look like? Like this [this list of programs, BHOs, etc. will not in most cases be the same as the ones shown in this screenshot]:

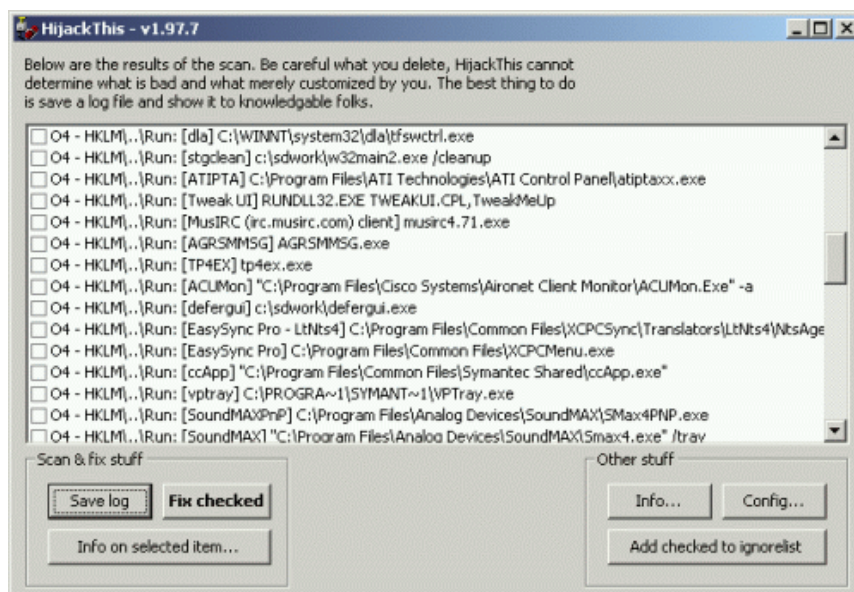


Figure 8: HijackThis screenshot

This tool is not for non-techies; luckily some kind soul has come to the rescue to assist in understanding the raw log files produced by HijackThis. This online tool is known as the ‘HijackThis Log Analyser’³. This is a useful site for turning the output of HijackThis into something that means something to most end-users, not just techies or propeller-heads.

HijackThis can also be used to remove scumware.

The beauty of HijackThis is that it is useable by most non-technical users; it is small and currently is not being defeated/manipulated by malware, unlike a number of other system diagnostic tools. So, if you think you are infected and have tried all the usual things to track down the rogue application, then give HijackThis a go. What have you got to lose, apart from the scumware?

3.3.6 WinPatrol

WinPatrol is an interesting tool described by its author as a “*robust SECURITY MONITOR, WinPatrol will alert you to hijackings, malware attacks and critical changes made to your computer without your permission.*”

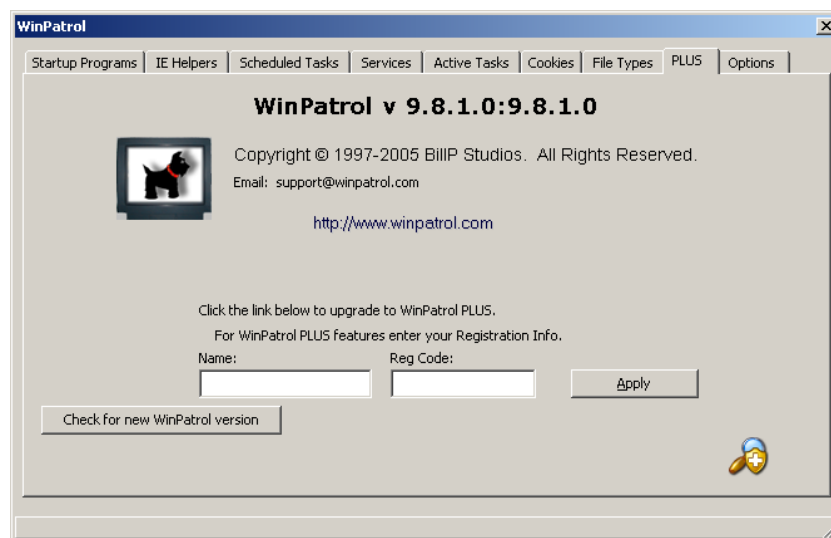


Figure 9: WinPatrol screenshot.

It is a rather useful watchdog tool, as it monitors numerous parts of the operating system and key applications, such as Internet Explorer. WinPatrol regularly checks the system areas monitored and warns you about any changes. You get to decide whether the change is allowed or not.

It has functionality that is found in a number of individual diagnostic tools, such as Sysinternals autoruns⁴ and a number of Windows tasks, such as displaying the current active tasks and services.

Website: <http://www.winpatrol.com/>

3.4 Virtual Analysis of real Malcode

Other than forensics this is the most technical and also most useful section as it allows you to see exactly what a malcode is doing, in real-time. The tools covered here are for advanced users only who are already used to handling live malcode. As mentioned earlier in this paper, a reasonable number of malware now has the ability to detect that it is being run inside a VM or Sandbox and may well either change its behaviour accordingly; this could be as simple as not running any malicious code, or it may turn destructive and delete files, directories, format the drive or simply kill the VM instead.

³ The HijackThis Log Analyser can be found here: <http://www.hijackthis.de/en>

⁴ Which can be downloaded from here: <http://www.sysinternals.com/Utilities/Autoruns.html>

3.4.1 *VMware*

“VMware Workstation lets you use your virtual machines to run Windows, Linux and a host of other operating systems side-by-side on the same computer. You can switch between operating systems instantly with a click of a mouse, share files between virtual machines with drag-and-drop functionality and access all the peripheral devices you rely on.

With Workstation, you can take a “snapshot” that preserves the state of a virtual machine so you can return to it at any time. Snapshots are useful when you need to revert your virtual machine to a prior, stable system state. Workstation displays thumbnails of all your snapshots on a single screen, making it easy for you to track and revert to a previously saved snapshot.”

The screenshot shown in *Figure 10* is of VMWare Workstation showing a running XP Home guest operating system.

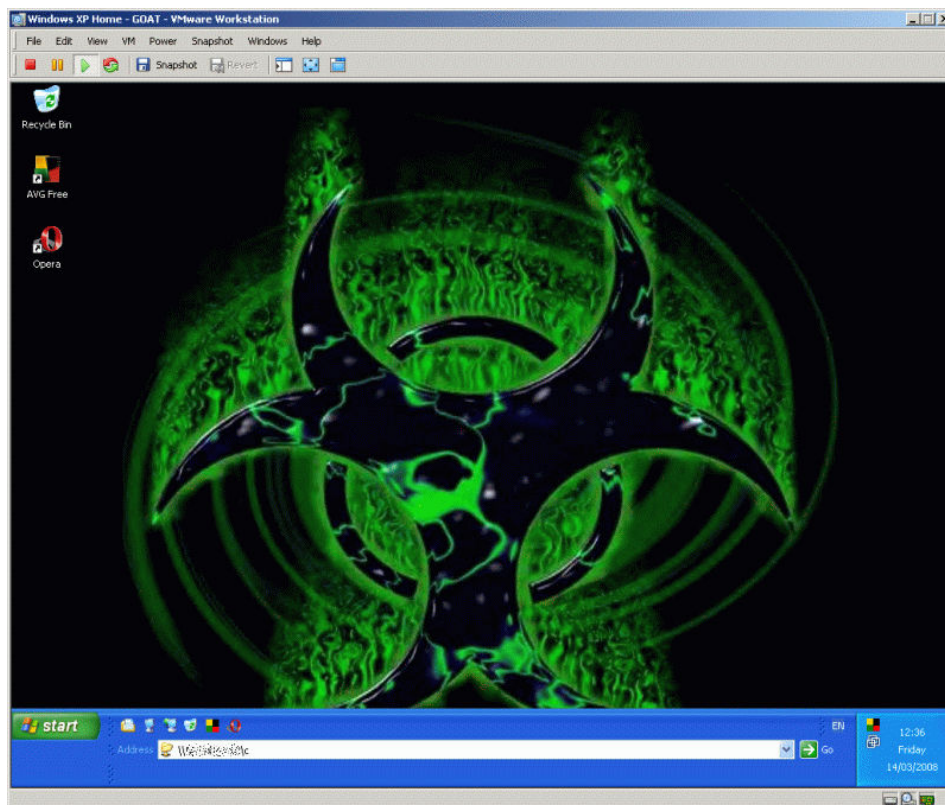


Figure 10: VMWare screenshot

Website: <http://www.vmware.com/>

3.4.2 *InCtrl5*

“InCtrl5 is the fifth incarnation of one of PC Magazine's most popular utilities. By monitoring the changes made to your system when you install new software, it enables you to troubleshoot any problems that come up. Virtually every modern program uses an install utility that installs or updates files; these utilities may also record data in the Registry and update INI files or other essential text files. A companion uninstall utility should precisely reverse the effects of the install utility. When a newly installed program causes existing applications to fail, or when the supplied uninstall utility can't complete its task, you need a record of exactly what the original install utility did in order to restore your system. InCtrl5 can provide this record.”

Website: <http://www.pcmag.com/article2/0,4149,9882,00.asp>

3.4.3 *PSTools*

“The Windows NT and Windows 2000 Resource Kits come with a number of command line tools that help you administer your Windows NT/2K systems. Over time, I've grown a collection of similar tools,

including some not included in the Resource Kits. What sets these tools apart is that they all allow you to manage remote systems as well as the local one. The first tool in the suite was PsList, a tool that lets you view detailed information about processes, and the suite is continually growing. The "Ps" prefix in PsList relates to the fact that the standard UNIX process listing command-line tool is named "ps", so I've adopted this prefix for all the tools in order to tie them together into a suite of tools named PsTools.

Note: some anti-virus scanners report that one or more of the tools are infected with a "remote admin" virus. None of the PsTools contain viruses, but they have been used by viruses, which is why they trigger virus notifications.

The tools included in the PsTools suite, which are downloadable individually or as a package, are:

- PsExec - execute processes remotely
- PsFile - shows files opened remotely
- PsGetSid - display the SID of a computer or a user
- PsInfo - list information about a system
- PsKill - kill processes by name or process ID
- PsList - list detailed information about processes
- PsLoggedOn - see who's logged on locally and via resource sharing (full source is included)
- PsLogList - dump event log records
- PsPasswd - changes account passwords
- PsService - view and control services
- PsShutdown - shuts down and optionally reboots a computer
- PsSuspend - suspends processes
- PsUptime - shows you how long a system has been running since its last reboot (PsUptime's functionality has been incorporated into PsInfo)

All of the utilities in the PsTools suite work on Windows Vista, Windows NT, Windows 2000, Windows XP and Windows Server 2003. The PsTools download package includes an HTML help file with complete usage information for all the tools.”

Website: <http://www.microsoft.com/technet/sysinternals/FileAndDisk/PsTools.mspx>

3.4.4 Norman Sandbox

“Norman Sandbox Information Center (NSIC) is a web site that offers

- * Free uploads of program files that you suspect are malicious or infected by malicious components, and instant analysis by Norman SandBox. The result is also sent you by email.
- * Comprehensive statistics of files that are uploaded to NSIC during the latest day, week and month. You will then be able to see tendencies in the creation of malicious software.
- * In-depth information about the analysis performed by Norman SandBox of each malicious file that is uploaded.
- * Search facility in all analyses after Registry keys, file names, etc.”

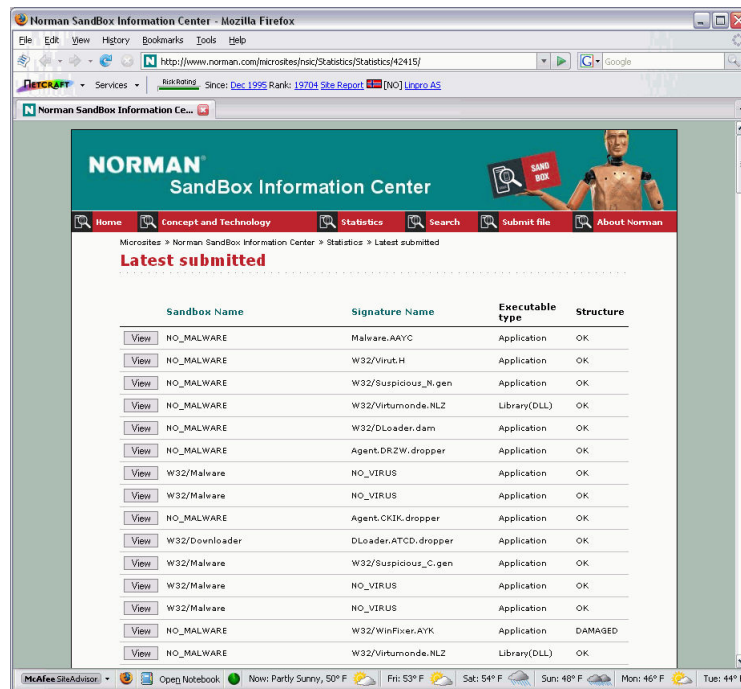


Figure 11: Norman Sandbox website screenshot

Website: <http://www.norman.com/microsites/nsic/Submit/en-uk>

3.4.5 CWSandbox

“CWSandbox is an approach to automatically analyze malware which is based on behavior analysis: malware samples are executed for a finite time in a simulated environment, where all system calls are closely monitored. From these observations, CWSandbox is able to automatically generate a detailed report which greatly simplifies the task of a malware analyst.”



Figure 12: CWSandbox website screenshot

Website: <http://www.cwsandbox.org/> or <http://research.sunbelt-software.com/Submit.aspx>

3.5 Scanners

This section covers the main options you have to get any suspected files scanned by multiple anti-malware scanners, without having to buy, install and then run each product against the suspected files. I have also included the sample submission e-mail addresses for most of the major anti-malware firms, so that you can send samples directly to them instead, if you prefer.

Finally I will briefly cover the various classes of anti-malware tools that you should consider, and hopefully already have in place.

3.5.1 VirusTotal

“VirusTotal is a service that analyzes suspicious files and facilitates the quick detection of viruses, worms, trojans, and all kinds of malware detected by antivirus engines.

Specs:

- * Free, independent service
- * Use of multiple antivirus engines
- * Real-time automatic updates of virus signatures
- * Detailed results from each antivirus engine
- * Real time global statistics”

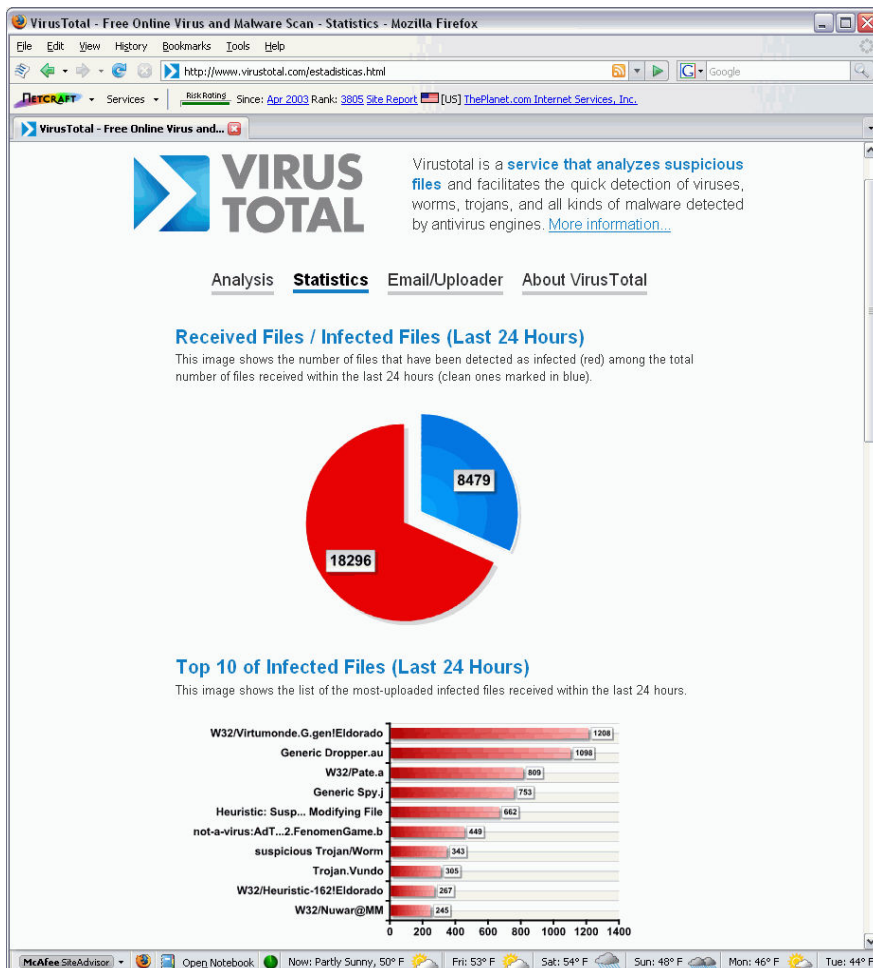


Figure 13: Virus Total website screenshot

Website: <http://www.virustotal.com/>

3.5.2 Jotti

“This service is by no means 100% safe. If this scanner says 'OK', it does not necessarily mean the file is clean. There could be a whole new virus on the loose. NEVER EVER rely on one single product

only, not even this service, even though it utilizes several products. Therefore, we cannot and will not be held responsible for any damage caused by results presented by this non-profit online service.

Also, we are aware of the implications of a setup like this. We are sure this whole thing is by no means scientifically correct, since this is a fully automated service (although manual correction is possible). We are aware, in spite of efforts to proactively counter these, false positives might occur, for example. We do not consider this a very big issue, so please do not e-mail us about it. This is a simple online scan service, not the University of Wichita.

Scanning can take a while, since several scanners are being used, plus the fact some scanners use very high levels of (time consuming) heuristics. Scanners used are Linux versions, differences with Windows scanners may or may not occur. Another note: some scanners will only report one virus when scanning archives with multiple pieces of malware.

Virus definitions are updated every hour. There is a 10Mb limit per file. Please refrain from uploading tons of hex-edited or repacked variants of the same sample.”

Website: <http://virusscan.jotti.org/>

3.5.3 Vendors

Anti-Virus Vendor Submission E-mail Addresses:

- *Authentium (Command Antivirus) virus@authentium.com*
- *Computer Associates (US) - virus@ca.com*
- *Computer Associates (Vet/EZ) - ipevirus@vet.com.au*
- *DialogueScience (Dr. Web) - Antivir@dials.ru*
- *Eset (NOD32) - sample@nod32.com*
- *F-Secure Corp. - samples@f-secure.com*
- *Frisk Software (F-PROT) - viruslab@f-prot.com*
- *Grisoft (AVG) - virus@grisoft.cz*
- *H+BEDV (AntiVir, Vexira engine) - virus@antivir.de*
- *Kaspersky Labs - newvirus@kaspersky.com*
- *McAfee - virus_research@mcafee.com - use a ZIP file with the password 'infected' without the quotes)*
- *Norman (NVC) - analysis@norman.no>*
- *Panda Software - labs@pandasoftware.com*
- *Sophos Plc. - support@sophos.com*
- *Symantec (Norton) - avsubmit@symantec.com*
- *Trend Micro (PC-cillin) - virus_doctor@trendmicro.com*

3.5.4 Anti-Rootkit Tools

Rootkits have been around for *NIX systems for many years; however they are now a growing problem for Windows systems. This is not only true in regard to bots and worms; we are now seeing Spyware authors actively using so-called ‘rootkit’ technology. This really should be called ‘cloaking’ or ‘stealth’ techniques rather than ‘rootkit technology’ as what they are doing is hiding the malware files and processes from the operating system. Malware using stealth techniques is not a new phenomenon; many years ago DOS malware authors used similar techniques.

There are a number of tools available that claim to be able to detect and remove rootkits, these are listed below, along with the OS that they are suitable for:

- *ChkRootkit* [*NIX - <http://chkrootkit.org/>]
- *Rootkit Hunter* [*NIX - http://www.rootkit.nl/projects/rootkit_hunter.html]
- *RootkitRevealer* [Wintel - <http://www.sysinternals.com/ntw2k/freeware/rootkitreveal.shtml>]
- *UnHackme* [Wintel - <http://gratis.com/unhackme/>]
- *Blacklight* [Wintel - <http://www.f-secure.com/blacklight/>]

A number of anti-virus products now include so-called ‘rootkit’ detection functionality which is required to detect many of the more advanced ones that bind in at kernel level.

3.5.5 *Anti-Virus and Anti-Spyware*

On the subject of anti-virus tools, I am not going to list them as any sensible organisation should already have at least one deployed across their infrastructure, and preferably two different vendors [covering different parts of the infrastructure, say one for desktop/laptop and the other for file servers and/or at the perimeter scanning e-mail, http and ftp], so that the window of opportunity for a new malcode is as small as possible

The use of anti-virus technologies as a detection method for systems infected by malicious spyware, rootkits and bots is self-evident, as many bots, key loggers, rootkits, diallers and droppers are now reliably detected by anti-virus products.

Because of this we are seeing the inclusion of techniques in many of the modern bots and some other malicious spyware to allow them to disable as many security and anti-virus products as possible. In some cases this functionality may well be the first to be deployed, as a dropper being spammed out. Once run the dropper lowers or neutralises any local defences and then opens up the backdoor, or just downloads more components as required to complete the infiltration.

The thing to remember with anti-virus tools is that they can only [normally] detect malware they know about. New malware variants may well be detected by heuristics; however they are still far from perfect.

Many anti-virus vendors have bought in spyware detection technology, such as via an acquisition or licensing deals. Others have created their own and seamlessly integrated spyware detection into their existing anti-virus products. Either way it is good news for their customers.

There are also hardware [appliance] solutions that can be used to combat malware at the perimeter of the network, these use a variety of techniques such as URL filtering, active content blocking or filtering many of these appliances are policy driven, so that you can decide what should and shouldn't be allowed in to your network. Examples of these devices include:

- *Bluecoat WebFilter*
- *Finjan Vital Security™ Web Appliance*
- *McAfee Secure Web Gateway*

A number of the largest anti-virus vendors offer products that can be centrally managed and will also offer compliance statistics for coverage and how up-to-date the signatures and products are within your network. Some of the management tools have been updated to manage spyware detection and personal firewall components alongside the traditional anti-virus functionality. This allows complete coverage of not only desktops but also servers and in some cases security appliances and other perimeter/network solutions.

If you want spyware protection for your home computer, bearing in mind that home users' computers are more likely to be infected than those in large businesses, then this is the section of the paper for you.

However, if you are looking for anti-spyware tools that might be suitable for use in a small to medium

business or tools that may be useful for support staff; be they in small, medium or large businesses or even academia then this section should still be useful to you.

One of the anti-spyware tools I suggest that home users should consider is Ad-Aware. The product is easy to use, accurate and signature updates are regular. The free version will do on-demand scans and clean, however if you want on-access protection you will have to buy the Plus edition. This will get you the Ad-Watch on-access component that will block spyware as it tries to download or install.



Figure 14: Ad-Aware SE screenshot.

Likewise, I also suggest Spybot Search & Destroy to home users, and technical support staff too for cleaning up spyware infected/infested computers on their networks. Like Ad-Aware it works in two modes, on-demand and it also has an on-access component, known as Tea-Timer which not only will block spyware in real-time it also monitors the registry.

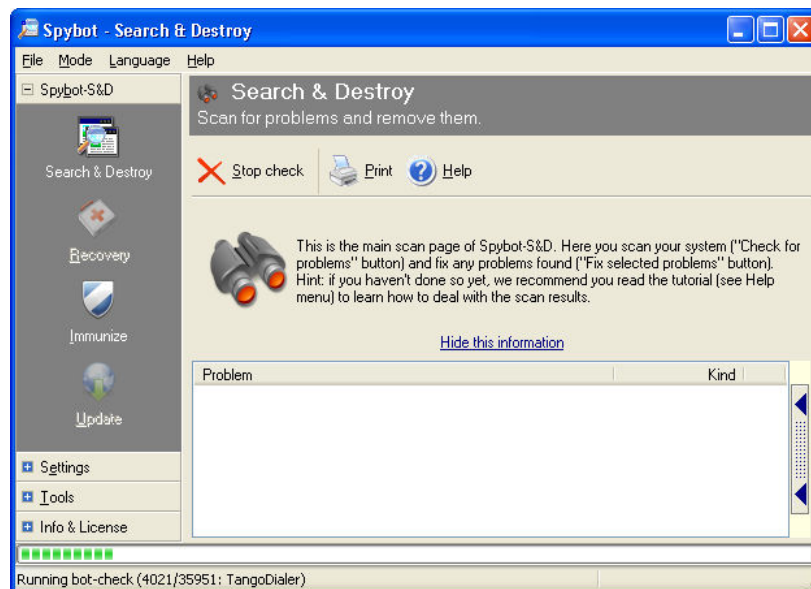


Figure 15: Spybot Search & Destroy screenshot.

Both of these anti-spyware tools well respected and updated regularly to detect new threats and are available in many different languages.

Before I finish this section of the paper, I would like to bring your attention to the fact that you need to be very careful when selecting an anti-spyware solution/tool, as there are a number of them that are spyware in their own right. You can find a list of the known 'bogus' anti-spyware and anti-malware tools here: http://www.spywarewarrior.com/rogue_anti-spyware.htm

3.6 Network Information

This section will cover a couple of tools that are very useful for gathering and acting on network data.

3.6.1 Wireshark

“Wireshark is the world's foremost network protocol analyzer, and is the de facto (and often de jure) standard across many industries and educational institutions.

Features

Wireshark has a rich feature set which includes the following:

- * Deep inspection of hundreds of protocols, with more being added all the time
- * Live capture and offline analysis
- * Standard three-pane packet browser
- * Multi-platform: Runs on Windows, Linux, OS X, Solaris, FreeBSD, NetBSD, and many others
- * Captured network data can be browsed via a GUI, or via the TTY-mode TShark utility
- * The most powerful display filters in the industry
- * Rich VoIP analysis
- * Read/write many different capture file formats: tcpdump (libpcap), Catapult DCT2000, Cisco Secure IDS iplog, Microsoft Network Monitor, Network General Sniffer® (compressed and uncompressed), Sniffer® Pro, and NetXray®, Network Instruments Observer, Novell LANalyzer, RADCOM WAN/LAN Analyzer, Shomiti/Finisar Surveyor, Tektronix K12xx, Visual Networks Visual UpTime, WildPackets EtherPeek/TokenPeek/AiroPeek, and many others
- * Capture files compressed with gzip can be decompressed on the fly
- * Live data can be read from Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, FDDI, and others (depending on your platform)
- * Decryption support for many protocols, including IPsec, ISAKMP, Kerberos, SNMPv3, SSL/TLS, WEP, and WPA/WPA2
- * Coloring rules can be applied to the packet list for quick, intuitive analysis
- * Output can be exported to XML, PostScript®, CSV, or plain text”

Website: <http://www.wireshark.org/>

3.6.2 Snort

“SNORT® is an open source network intrusion prevention and detection system utilizing a rule-driven language, which combines the benefits of signature, protocol and anomaly based inspection methods. With millions of downloads to date, Snort is the most widely deployed intrusion detection and prevention technology worldwide and has become the de facto standard for the industry.”

Website: <http://www.snort.org/>

3.7 Forensics

The tools covered in this section are really a last resort and should only be used by those that have received training in computer forensics. These tools are most useful when you are carrying out an investigation that may become a criminal case or where you need to capture evidence without changing or otherwise modifying a systems content.

3.7.1 Encase

“EnCase® Forensic is the industry standard in computer forensic investigation technology. With an intuitive GUI, superior analytics, enhanced email/Internet support and a powerful scripting engine, EnCase® provides investigators with a single tool, capable of conducting large-scale and complex investigations from beginning to end. Law enforcement officers, government/corporate investigators and consultants around the world benefit from the power of EnCase® Forensic in a way that far exceeds any other forensic solution.

* Acquire data in a forensically sound manner using software with an unparalleled record in courts worldwide.

* Investigate and analyze multiple platforms — Windows, Linux, AIX, OS X, Solaris and more — using a single tool.

- * Save days, if not weeks, of analysis time by automating complex and routine tasks with prebuilt EnScript® modules, such as Initialized Case and Event Log analysis.
- * Find information despite efforts to hide, cloak or delete.
- * Easily manage large volumes of computer evidence, viewing all relevant files, including "deleted" files, file slack and unallocated space.
- * Transfer evidence files directly to law enforcement or legal representatives as necessary.
- * Review options allow non-investigators, such as attorneys, to review evidence with ease.
- * Reporting options enable quick report preparation.”

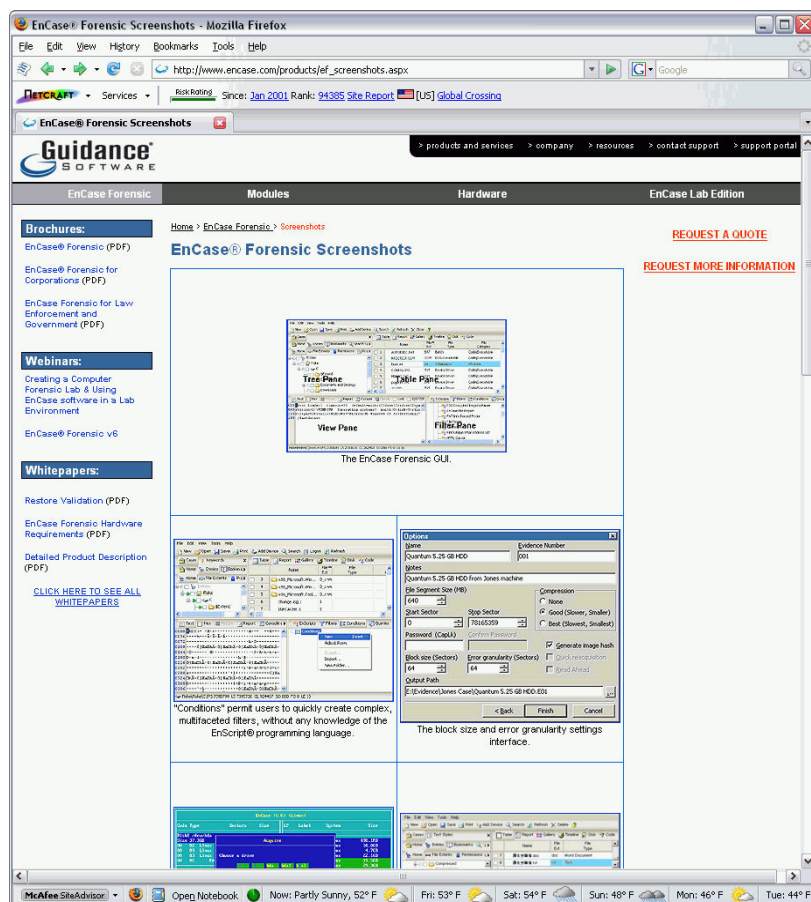


Figure 16: Encase website screenshot

Website: <http://www.guidancesoftware.com/>

3.7.2 F.I.R.E

“FIRE is a portable bootable cdrom based distribution with the goal of providing an immediate environment to perform forensic analysis, incident response, data recovery, virus scanning and vulnerability assessment.

Also provides necessary tools for live forensics/analysis on win32, sparc solaris and x86 linux hosts just by mounting the cdrom and using trusted static binaries available in /statbins.”



Figure 17: F.I.R.E website screenshot

Website: <http://biatchux.dmzs.com/>

3.8 Tricks

This section will discuss a few tricks that can be useful, such as using clean-up scripts to speed up remediation.

The following example was created to kill the running process, registry keys and files created by a specific SDbot variant which was undetectable at the time it was originally found. This script, and other variants of it, were used to automate the testing of systems for the malware, and if found the script kills the running malware processes, removes the malwares registry keys and finally deletes the specific malware files.

VB Scripting for quick and dirty cleanup, example:

```
'RemSdbot2.vbs - SDbot remover for specific variant.
'© Martin Overton, 2007 (martin@arachnophiliac.com)
'Version 0.99.2'
'Created to detect and remove an infection of the following Sdbot variant
'
'FileName: rundll.exe
'FileDateTime: 19/01/2007 14:05:00
'Filesize: 1364992
'MD5: 71fd1205f6d7550967bda6bf4491a50a
'CRC32: 36E8176E
'File Type: PE Executable
'
'To make this a silent script just rem out the Wscript.Echo lines

Wscript.Echo "SDBot Cleanup Script 2 - Click OK to proceed"

Const HKEY_CURRENT_USER = &H80000001
Const HKEY_LOCAL_MACHINE = &H80000002

strComputer = "."

' Check to see if infected marker [run key] exists
'
Set objRegistry=GetObject("winmgmts:\\\" & _
    strComputer & "\root\default:StdRegProv")

strKeyPath = "Software\Microsoft\Windows\CurrentVersion\Run"
strValueName = "Microsoft"
objRegistry.GetStringValue HKEY_LOCAL_MACHINE, strKeyPath, strValueName, strValue

If IsNull(strValue) Then
    Wscript.Echo "The registry key does not exist - This system does not seem to be
infected - Script Stopped"
    Wscript.Quit
Else

' If infected marker [run key] exists, then grab filename and terminate process
'
Set objWMIService = GetObject _
    ("winmgmts:\\\" & strComputer & "\root\cimv2")
Set colProcessList = objWMIService.ExecQuery _
    ("Select * from Win32_Process Where Name ='" & strValue & "'")
For Each objProcess in colProcessList
    objProcess.Terminate()
Next

'Pause for 10 seconds
'
Wscript.Sleep 10000

' Check to see if infected file exists, if so then delete it
'
Set objFSO = CreateObject("Scripting.FileSystemObject")
Const ReadOnly = 1

Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objFile = objFSO.GetFile("C:\windows\system32\" & strValue)

If objFile.Attributes AND ReadOnly Then
    objFile.Attributes = objFile.Attributes XOR ReadOnly
```

```
End If

If objFSO.FileExists("C:\windows\system32\" & strValue) Then
objFSO.DeleteFile("C:\windows\system32\" & strValue)

Else
    Wscript.Echo "The file does not exist - Script Stopped."
    Wscript.Quit
End If

' Remove the Sdbot variant registry keys
,
strKeyPath = "Software\Microsoft\Windows\CurrentVersion\Run"
strValueName = "Microsoft"
objRegistry.DeleteValue HKEY_LOCAL_MACHINE, strKeyPath, strValueName
objRegistry.DeleteValue HKEY_CURRENT_USER, strKeyPath, strValueName

strKeyPath = "Software\Microsoft\Windows\CurrentVersion\RunServices"
strValueName = "Microsoft"
objRegistry.DeleteValue HKEY_LOCAL_MACHINE, strKeyPath, strValueName

End If

Wscript.Echo "Script completed - This system should now be clean"
```

3.8.1 Clean Boot Disks

Using live Linux or a PE boot disk, such as Bart_PE can be very handy, not only in clean booting a suspected system but also in scanning the same system with little or no risk that any malcode will still be active on it. It needs not be a CD or DVD [from an ISO image], it could also be an external USB hard disk or a USB flash drive instead.

3.8.2 Techniques

Check the relevant registry keys for odd entries, common ones used include:

```
HKEY_LOCAL_MACHINE
Software\Microsoft\Windows\CurrentVersion\Run
Software\Microsoft\Windows\CurrentVersion\RunOnce
Software\Microsoft\Windows\CurrentVersion\RunOnceX

HKEY_CURRENT_USER
Software\Microsoft\Windows\CurrentVersion\Run
Software\Microsoft\Windows\CurrentVersion\RunOnce
```

However, there are lots of others that are used, mis-used and created by malware. Other than do this by hand you could use a tool such as AutoRuns, HijackThis or WinPatrol instead.

3.9 Real World Example 1

User noticed that their anti-virus was disabled, and so reported it to the helpdesk of the company affected.

The local support teams noticed that the system that had its anti-virus software disabled was making lots of outbound DNS lookups for odd websites that were not business related.

Further investigation of the suspected system found a file that looked to be involved, a sample was acquired and analysed in several sandboxes as well as tested against 30+ anti-malware tools; very few reported the file as either suspicious or infected.

Here's part of the analysis report, along with recommendations for remediation and suggestions for improvements to the protection of their infrastructure, including an early warning system:

Overview:

This malware is a share crawling parasitic file infector [virus] that once executed on a new system will create a number of new files [4 DLLs and 1 Sys file]. These are listed below:

- %Windir%\%SYSDIR%\lv362285.dl_
- %Windir%\%SYSDIR%\lv362285.dll
- %Windir%\%SYSDIR%\uo105244.dl_
- %Windir%\%SYSDIR%\uo105244.dll
- %Windir%\%SYSDIR%\drivers\mhqook.sys

It then proceeds to attempt to connect to the following domain names and download files found hosted there. This may include other malware components such as KillWin, bot clients, spyware, adware, other Trojans, etc. These may also perform a similar routine, which can quickly turn a clean system into a heavily infected one.

- *makemegood24.com*
- *446df1.makemegood24.com*
- *aaakemegood24.com*
- *perfectchoicel.com*
- *4475e1.perfectchoicel.com*
- *bperfectchoicel.com.local*
- *cash-ddt.net*

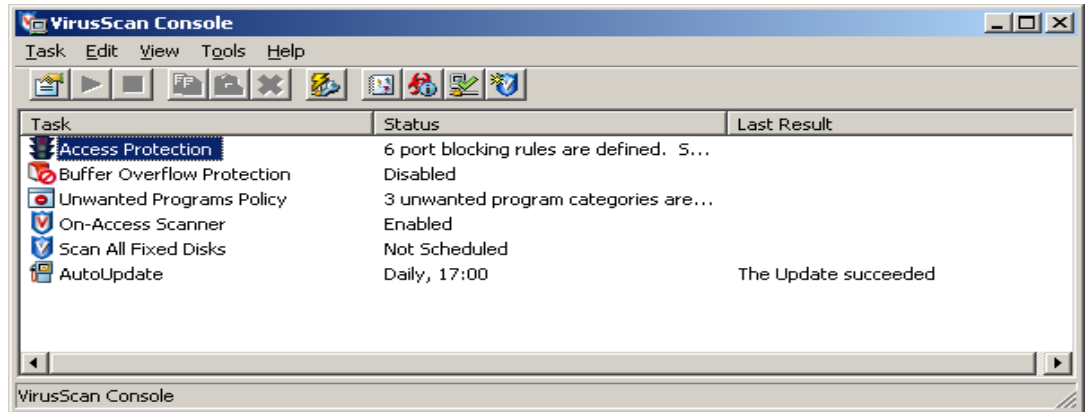
As part of this download process, it may disable any security software on the newly infected system, including personal firewalls and anti-virus processes.

The next phase is for the original malware file to search for new files [Windows 32bit PE file-types] to infect on the local machine and all systems that the newly infected system has access to, such as Windows shares. All infected files will grow by 57,344 bytes.

Recommendations:

1. Block all DNS activity to the domains used by the malware, as this will help to minimise the impact to that of the original installed malware. This can be achieved via a number of ways, such as DNS Black-holing [Nul Routing] those domain names, or blocking access to the domains via URL filtering at the proxy or other internet gateway.
2. Identify all infected systems and remove them from the network until remediated, preferably by re-imaging, or from a known clean backup.
3. Up-date all anti-virus tools used to latest version and/or ensure if you are using McAfee VirusScan that Access Protection is enabled [see below] and configured to disable files to be

created/modified in the Windows and Windows\System directories. You can also block all IRC traffic via this feature.



4. Once all systems on the network are clean, then the following should be installed on all systems and configured to minimise a repeat of this incident:
 - Up-to-date anti-virus with on-access scanning enabled by default, and Access Protection and Buffer Overflow protection enabled by default [assuming McAfee VirusScan 8.x or later used].
 - Personal firewall installed and correctly configured [preferably locked]. This can raise the alarm when new [unknown or modified] programs or processes try to 'phone home' or otherwise access the internet.
 - Ensure that all systems are patched as soon as possible by new patches released by vendors.

The following should also be considered for use as part of an early warning system and to help speed up identification of newly, or missed infected systems.

- WormCharmer [SMB-Lure] or similar honeypot/honeynet. This acts as a sacrificial goat and is designed to be attacked by malware [new or old] without risking infecting the host or your network.
- IDS or IPS; this can be as simple as using SNORT with freely available malware/exploit detection signatures [rules] to identify a possible infected system.

3.10 Real World Example 2

An unknown malware was causing clients running anti-virus on a network to lose connection to the anti-virus management server. So, with the help of local resources on site we managed to obtain a sample which was suspected to be the culprit.

The anti-virus deployed on the network and workstations did not detect the malware as it was brand new.

The following data allowed me to understand what the malware was doing and from this clean-up scripts could be created as well as blocking the infection vector used by the malware.

I leave it as an exercise to the reader to try and work out what this specific malware does when it is executed. Consider it a test of your knowledge.

CWSandbox Results:

```

Analysis Number      1
Parent ID           0
Process ID          588
Filename            c:\temp\ff37e574c7694879ff73777886a82dee.exe
Filesize            215040 bytes
MD5                 ff37e574c7694879ff73777886a82dee
Start Reason        AnalysisTarget
Termination Reason  NormalTermination
Start Time          00:00.218
Stop Time           00:04.281
DLL-Handling
Loaded DLLs
c:\temp\ff37e574c7694879ff73777886a82dee.exe
C:\WINDOWS\System32\ntdll.dll
C:\WINDOWS\system32\kernel32.dll
C:\WINDOWS\system32\user32.dll
C:\WINDOWS\system32\GDI32.dll
C:\WINDOWS\system32\ADVAPI32.dll
C:\WINDOWS\system32\RPCRT4.dll
C:\WINDOWS\system32\MPR.dll
C:\WINDOWS\System32\ODBC32.dll
C:\WINDOWS\system32\msvcrt.dll
C:\WINDOWS\system32\COMCTL32.dll
C:\WINDOWS\system32\SHELL32.dll
C:\WINDOWS\system32\SHLWAPI.dll
C:\WINDOWS\system32\comdlg32.dll
C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-
Controls_6595b64144ccf1df_6.0.2600.1612_x-ww_7c379b08\
C:\WINDOWS\System32\odbcint.dll
C:\WINDOWS\system32\WININET.dll
C:\WINDOWS\system32\CRYPT32.dll
C:\WINDOWS\system32\MSASN1.dll
C:\WINDOWS\system32\OLEAUT32.dll
C:\WINDOWS\system32\OLE32.DLL
C:\WINDOWS\System32\WS2_32.dll
C:\WINDOWS\System32\WS2HELP.dll
C:\WINDOWS\System32\wsock32.dll
C:\WINDOWS\System32\pstorec.dll
C:\WINDOWS\System32\ATL.DLL
C:\WINDOWS\System32\Wship6.dll
C:\WINDOWS\System32\iphlpapi.dll
C:\WINDOWS\System32\Secur32.dll
user32.dll
USER32.dll
Filesystem
New Files
C:\WINDOWS\System32\crsss.exe
Opened Files
\SystemRoot\AppPatch\sysmain.sdb
\SystemRoot\AppPatch\sytest.sdb
\Device\NamedPipe\ShimViewer
C:\WINDOWS\System32\crsss.exe
Chronological order
Copy File: c:\temp\ff37e574c7694879ff73777886a82dee.exe to
C:\WINDOWS\System32\crsss.exe
Open File: \SystemRoot\AppPatch\sysmain.sdb (OPEN_EXISTING)

```

```
Open File: \SystemRoot\AppPatch\systest.sdb (OPEN_EXISTING)
Open File: \Device\NamedPipe\ShimViewer (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\crsss.exe ()
Find File: crsss.exe
Registry
Process Management    Creates Process - Filename () CommandLine:
(C:\WINDOWS\System32\crsss.exe --install c:\temp\ff37e574c7694879ff73777886a82dee.exe)
As User: () Creation Flags: (DETACHED_PROCESS)
Kill Process - Filename () CommandLine: () Target PID: (588) As User: () Creation
Flags: ()
System Info    Get System Directory
The following process was started by process: 1
Analysis Number    2
Parent ID    1
Process ID    1020
Filename    C:\WINDOWS\System32\crsss.exe --install
c:\temp\ff37e574c7694879ff73777886a82dee.exe
Filesize    215040 bytes
MD5    ff37e574c7694879ff73777886a82dee
Start Reason    CreateProcess
Termination Reason    NormalTermination
Start Time    00:03.750
Stop Time    01:00.531
DLL-Handling
Loaded DLLs
C:\WINDOWS\System32\crsss.exe
C:\WINDOWS\System32\ntdll.dll
C:\WINDOWS\system32\kernel32.dll
C:\WINDOWS\system32\user32.dll
C:\WINDOWS\system32\GDI32.dll
C:\WINDOWS\system32\ADVAPI32.dll
C:\WINDOWS\system32\RPCRT4.dll
C:\WINDOWS\system32\MPR.dll
C:\WINDOWS\System32\ODBC32.dll
C:\WINDOWS\system32\msvcrt.dll
C:\WINDOWS\system32\COMCTL32.dll
C:\WINDOWS\system32\SHELL32.dll
C:\WINDOWS\system32\SHLWAPI.dll
C:\WINDOWS\system32\comdlg32.dll
C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-
Controls_6595b64144ccf1df_6.0.2600.1612_x-ww_7c379b08\
C:\WINDOWS\System32\odbcint.dll
C:\WINDOWS\system32\WININET.dll
C:\WINDOWS\system32\CRYPT32.dll
C:\WINDOWS\system32\MSASN1.dll
C:\WINDOWS\system32\OLEAUT32.dll
C:\WINDOWS\system32\OLE32.DLL
C:\WINDOWS\System32\WS2_32.dll
C:\WINDOWS\System32\WS2HELP.dll
C:\WINDOWS\System32\wsock32.dll
C:\WINDOWS\System32\pstorec.dll
C:\WINDOWS\System32\ATL.DLL
C:\WINDOWS\System32\Wship6.dll
C:\WINDOWS\System32\iphlpapi.dll
C:\WINDOWS\System32\Secur32.dll
user32.dll
psapi.dll
SHLWAPI.dll
VERSION.dll
shell32.dll
Filesystem
New Files
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\WER7.tmp.dir00\appcompat.txt
Opened Files
\\.PIPE\lsarpc
C:\WINDOWS\System32\advapi32.dll
C:\WINDOWS\System32\advapi32.dll
C:\WINDOWS\System32\gdi32.dll
C:\WINDOWS\System32\gdi32.dll
C:\WINDOWS\System32\kernel32.dll
C:\WINDOWS\System32\kernel32.dll
C:\WINDOWS\System32\ntdll.dll
C:\WINDOWS\System32\ntdll.dll
C:\WINDOWS\System32\ole32.dll
C:\WINDOWS\System32\ole32.dll
C:\WINDOWS\System32\oleaut32.dll
C:\WINDOWS\System32\oleaut32.dll
```

```

C:\WINDOWS\System32\shell32.dll
C:\WINDOWS\System32\shell32.dll
C:\WINDOWS\System32\user32.dll
C:\WINDOWS\System32\user32.dll
C:\WINDOWS\System32\WININET.DLL
C:\WINDOWS\System32\WININET.DLL
C:\WINDOWS\System32\winsock.dll
C:\WINDOWS\System32\winsock.dll
\SystemRoot\AppPatch\sysmain.sdb
\SystemRoot\AppPatch\sysstest.sdb
\Device\NamedPipe\ShimViewer
C:\WINDOWS\System32\dwwin.exe
C:\WINDOWS\System32\drwtsn32.exe
Deleted Files
c:\temp\ff37e574c7694879ff73777886a82dee.exe
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\WER7.tmp.dir00\appcompat.txt
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\WER7.tmp
Chronological order
Delete File: c:\temp\ff37e574c7694879ff73777886a82dee.exe
Get File Attributes: C:\WINDOWS\ Flags: (SECURITY_ANONYMOUS)
Open File: \\.\PIPE\lsarpc (OPEN_EXISTING)
Create File: C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\WER7.tmp.dir00\appcompat.txt
Find File: C:\WINDOWS\System32\*
Open File: C:\WINDOWS\System32\advapi32.dll ()
Find File: advapi32.dll
Open File: C:\WINDOWS\System32\advapi32.dll (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\gdi32.dll ()
Find File: gdi32.dll
Open File: C:\WINDOWS\System32\gdi32.dll (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\kernel32.dll ()
Find File: kernel32.dll
Open File: C:\WINDOWS\System32\kernel32.dll (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\ntdll.dll ()
Find File: ntdll.dll
Open File: C:\WINDOWS\System32\ntdll.dll (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\ole32.dll ()
Find File: ole32.dll
Open File: C:\WINDOWS\System32\ole32.dll (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\oleaut32.dll ()
Find File: oleaut32.dll
Open File: C:\WINDOWS\System32\oleaut32.dll (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\shell32.dll ()
Find File: shell32.dll
Open File: C:\WINDOWS\System32\shell32.dll (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\user32.dll ()
Find File: user32.dll
Open File: C:\WINDOWS\System32\user32.dll (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\WININET.DLL ()
Find File: WININET.DLL
Open File: C:\WINDOWS\System32\WININET.DLL (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\winsock.dll ()
Find File: winsock.dll
Open File: C:\WINDOWS\System32\winsock.dll (OPEN_EXISTING)
Open File: \SystemRoot\AppPatch\sysmain.sdb (OPEN_EXISTING)
Open File: \SystemRoot\AppPatch\sysstest.sdb (OPEN_EXISTING)
Open File: \Device\NamedPipe\ShimViewer (OPEN_EXISTING)
Open File: C:\WINDOWS\System32\dwwin.exe ()
Find File: dwwin.exe
Delete File: C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\WER7.tmp.dir00\appcompat.txt
Delete File: C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\WER7.tmp
Open File: C:\WINDOWS\System32\drwtsn32.exe ()
Find File: drwtsn32.exe
Mutexes      Creates Mutex: CRSSSSSSSS
Registry
Changes
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run "Win32 Security Service" = C:\WINDOWS\System32\crsss.exe
Reads
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run "Win32 Security Service"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "DoReport"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "ShowUI"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "AllOrNone"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "IncludeMicrosoftApps"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "IncludeWindowsApps"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "DoTextLog"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "IncludeKernelFaults"

```

```
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "IncludeShutdownErrs"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "NumberOfFaultPipes"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "NumberOfHangPipes"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "MaxUserQueueSize"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting "ForceQueueMode"
HKEY_LOCAL_MACHINE\System\Setup "SystemSetupInProgress"
HKEY_LOCAL_MACHINE\Software\Microsoft\PCHealth\ErrorReporting\ExclusionList
"crsss.exe"
Process Management      Creates Process - Filename () CommandLine:
(C:\WINDOWS\System32\dwwin.exe -x -s 1556) As User: () Creation Flags:
(CREATE_DEFAULT_ERROR_MODE)
Kill Process - Filename () CommandLine: () Target PID: (1300) As User: () Creation
Flags: ()
Kill Process - Filename () CommandLine: () Target PID: (1020) As User: () Creation
Flags: ()
Enum Processes
Enum Modules - Target PID: (1020)
Enum Modules - Target PID: (1020)
Open Process - Filename () Target PID: (4)
Open Process - Filename () Target PID: (592)
Open Process - Filename () Target PID: (640)
Open Process - Filename () Target PID: (664)
Open Process - Filename () Target PID: (708)
Open Process - Filename () Target PID: (724)
Open Process - Filename () Target PID: (744)
Open Process - Filename () Target PID: (880)
Open Process - Filename () Target PID: (948)
Open Process - Filename () Target PID: (1060)
Open Process - Filename () Target PID: (1204)
Open Process - Filename () Target PID: (1256)
Open Process - Filename (C:\WINDOWS\Explorer.EXE) Target PID: (1424)
Open Process - Filename () Target PID: (1544)
Open Process - Filename () Target PID: (1948)
System Info      Get System Directory
User Management      Revert To Self
Network Activity
The following process was started by process: 2
Analysis Number      3
Parent ID            2
Process ID           1108
Filename              C:\WINDOWS\System32\dwwin.exe -x -s 1556
Filesize              180224 bytes
MD5                   9a02cc6c840d09ae5ba5758d4adc451c
Start Reason          CreateProcess
Termination Reason    Timeout
Start Time            00:06.359
Stop Time             01:00.453
DLL-Handling
Loaded DLLs
C:\WINDOWS\System32\dwwin.exe
C:\WINDOWS\System32\ntdll.dll
C:\WINDOWS\system32\kernel32.dll
C:\WINDOWS\system32\ADVAPI32.DLL
C:\WINDOWS\system32\RPCRT4.dll
C:\WINDOWS\system32\COMCTL32.DLL
C:\WINDOWS\system32\GDI32.dll
C:\WINDOWS\system32\USER32.dll
C:\WINDOWS\system32\OLEAUT32.DLL
C:\WINDOWS\system32\MSVCRT.DLL
C:\WINDOWS\system32\OLE32.DLL
C:\WINDOWS\system32\SHELL32.DLL
C:\WINDOWS\system32\SHLWAPI.dll
C:\WINDOWS\system32\VERSION.DLL
C:\WINDOWS\system32\WININET.DLL
C:\WINDOWS\system32\CRYPT32.dll
C:\WINDOWS\system32\MSASN1.dll
C:\WINDOWS\WinSxS\x86_Microsoft.Windows.Common-
Controls_6595b64144ccf1df_6.0.2600.1612_x-ww_7c379b08\
C:\WINDOWS\System32\wsock32.dll
C:\WINDOWS\System32\WS2_32.dll
C:\WINDOWS\System32\WS2HELP.dll
C:\WINDOWS\System32\pstorec.dll
C:\WINDOWS\System32\ATL.DLL
C:\WINDOWS\System32\Wship6.dll
C:\WINDOWS\System32\iphlpapi.dll
C:\WINDOWS\System32\Secur32.dll
.\UxTheme.dll
```

```

imm32.dll
ole32.dll
riched20.dll
shfolder.dll
shell32.dll
PSAPI.DLL
C:\WINDOWS\System32\1033\dwintl.dll
comctl32.dll
RASAPI32.DLL
RTUTILS.DLL
SHELL32.dll
netapi32.dll
Filesystem
New Files
C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\D011.dmp
Opened Files
\\.\PIPE\lsarpc
c:\autoexec.bat
Chronological order
Get File Attributes: C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp Flags: (SECURITY_ANONYMOUS)
Create File: C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\D011.dmp
Open File: \\.\PIPE\lsarpc (OPEN_EXISTING)
Get File Attributes: c:\autoexec.bat Flags: (SECURITY_ANONYMOUS)
Open File: c:\autoexec.bat (OPEN_EXISTING)
Find File: C:\Documents and Settings\All Users\Application
Data\Microsoft\Network\Connections\Pbk\*.pbk
Find File: C:\WINDOWS\System32\Ras\*.pbk
Find File: C:\Documents and Settings\Administrator\Application
Data\Microsoft\Network\Connections\Pbk\*.pbk
INI Files
Read INI File
WIN.INI [windows] ScrollInset =
WIN.INI [windows] DragDelay =
WIN.INI [windows] DragMinDist =
WIN.INI [windows] ScrollDelay =
WIN.INI [windows] ScrollInterval =
WIN.INI [richedit30] flags =
Mutexes      Creates Mutex: RasPbFile
Registry
Reads
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion "DigitalProductId"
HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Settings "Anchor Color"
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\AeDebug "Debugger"
Process Management      Enum Modules - Target PID: (1020)
Service Management      Open Service Manager - Name: "SCM"
System Info      Get System Directory
Get Computer Name
User Management      Impersonate User - Domain: () User: (Administrator)
Virtual Memory      VM Read - Target: (1020) Address: ($0012F340) Size: (8)
VM Read - Target: (1020) Address: ($0012F42C) Size: (80)
VM Read - Target: (1020) Address: ($0012F448) Size: (716)
VM Read - Target: (1020) Address: ($7FFDE000) Size: (28)
VM Read - Target: (1020) Address: ($7FFE0284) Size: (256)
VM Read - Target: (1020) Address: ($7FFDD000) Size: (28)
VM Read - Target: (1020) Address: ($7FFDC000) Size: (28)
VM Read - Target: (1020) Address: ($7FFDB000) Size: (28)
VM Read - Target: (1020) Address: ($7FFDA000) Size: (28)
VM Read - Target: (1020) Address: ($7FFD9000) Size: (28)
Window Enum Windows
Analysis Number      4
Parent ID      0
Process ID      708
Filename
Filesize      -1 bytes
MD5
Start Reason      SCM
Termination Reason      Unknown
Start Time      00:08.187
Stop Time      00:00.000
Analysis Number      5
Parent ID      0
Process ID      708
Filename
Filesize      -1 bytes
MD5
Start Reason      SCM
Termination Reason      Unknown

```

Start Time 00:08.203
 Stop Time 00:00.000

File analysis:

Here is another analysis of the same file, using Norman Sandbox, a virus scanning service [similar to Jotti or VirusTotal], FileAlyzer as well as some comments about what the malware does.

```

FileName: crsss.exe
FileDateTime: 04/03/2007 16:37:16
Filesize: 215040
MD5: ff37e574c7694879ff73777886a82dee
CRC32: 493C2838
File Type: PE Executable
=====
Norman SandBox Reporter
http://www.norman.com/Product/Sandbox-products/Reporter/
crsss.exe : Not detected by Sandbox (Signature: NO_VIRUS)
 [ General information ]
   * File length: 215040 bytes.
   * MD5 hash: ff37e574c7694879ff73777886a82dee.
(C) 2004-2006 Norman ASA. All Rights Reserved.
=====
Scan report of: crsss.exe
@Proventia-VPS Malicious (Cancelled)
AntiVir TR/Rinbot.F
Avast! -
AVG Win32/CryptExe
BitDefender Backdoor.Vanbot.R
ClamAV -
Command -
Dr Web BackDoor.IRC.Sdbot.1142
eSafe Win32.Rinbot.A
eTrust-VET Win32/Nirbot.V
eTrust-VET (BETA) Win32/Nirbot.V
Ewido -
F-Prot -
F-Secure Backdoor.Win32.VanBot.ay
F-Secure (BETA) Backdoor.Win32.VanBot.ay
Fortinet W32/RINBOT.L!worm
Fortinet (BETA) W32/RINBOT.L!worm
Ikarus Trojan.Win32.Rinbot.F
Kaspersky Backdoor.Win32.VanBot.ay
McAfee W32/Sdbot.worm.gen.ai
McAfee (BETA) W32/Sdbot.worm.gen.ai
Microsoft -
Nod32 Win32/Rinbot.F trojan
Norman -
Panda W32/Vanbot.M.worm
Panda (BETA) W32/Vanbot.M.worm
QuickHeal -
Rising -
Sophos W32/Delbot-O
Symantec W32.Rinbot.A
Symantec (BETA) W32.Rinbot.A
Trend Micro WORM_RINBOT.L
Trend Micro (BETA) WORM_RINBOT.L
UNA Backdoor.VanBot.CFC6
VBA32 Trojan.Win32.Rinbot.F
VirusBuster Backdoor.Vanbot.Gen!Pac
WebWasher Trojan.Rinbot.F
YY_Spybot -
=====
PEInfo (Copyright by McAfee) report of the submitted files:
crsss.exe SZ:215040 EP:0x0008D35D DS: 0x45E7738D 2007-3-2 00:45:01
MD5:0xFF37E574C7694879FF73777886A82DEE
SectNum:8 VSIZE : RVA : FSIZE : FFFF : FLAGS : CRC32
0 : .text 0001C000: 00001000: 00000000: 00000400: E0000020: 00000000
1 : fabsk18p 00006000: 0001D000: 00000000: 00000400: E0000060: 00000000
2 : .data 00014000: 00023000: 00000000: 00000400: C0000040: 00000000
3 : .rsrc 00001000: 00037000: 00001000: 00000400: 40000040: 176A2128
4 : 99cvbjdu 00001000: 00038000: 00000000: 00001400: C0000040: 00000000
5 : ut7h7i2x 00022000: 00039000: 00000000: 00001400: E0000020: 00000000
6 : znnrn47v 00033000: 0005B000: 00032381: 00001400: E0000060: 64426022
7 : tdbkm0a1 00001000: 0008E000: 00001000: 00033800: 40000080: C9FCB827
RS:0x10000000560000074C0B54 RDS: 0x00000000 1970-1-1 00:00:00
*EP: 0xE8F7FEFFFF0574110000FFE0E8EBFEFFFF056B010000FFE0E804000000FFFFFF

```

```

IMPS: kernel32.dll(12), user32.dll(2)
=====
*****
FileAlyzer © 2003-2005 Patrick M. Kolla. All Rights Reserved.
*****
File: crsss.exe
Date: 08/03/2007 09:12:38
**** General *****
      Location: \\10.109.37.2\c\samples\mail\crsss2\
      Size: 215040
      Version:
      CRC-32: 493C2838
      MD5: FF37E574C7694879FF73777886A82DEE
      SHA1: C4B2C067293E9F96CB56C1287D610664802F66F2
      Read only: Yes
      Hidden: No
      System file: No
      Directory: No
      Archive: Yes
      Symbolic link: No
      Time stamp: 04 March 2007 16:37:16
      Creation: 07 March 2007 21:47:12
      Last access: 08 March 2007 09:13:48
      Last write: 04 March 2007 16:37:16
**** PE Header *****
      Signature: 00004550
      Machine: 014C - Intel 386
      Number of sections: 0008
      Time/Date stamp: 45E7738D
      Pointer to symbol table: 00000000
      Number of symbols: 00000000
      Size of optional header: 00E0
      Characteristics: 0103
      Magic: 010B
      Linker version (major): 08
      Linker version (minor): 00
      Size of code: 0001C000
      Size of initialized data: 0000C000
      Size of uninitialized data: 00000000
      Address of entry point: 0008D35D
      Base of code: 0005B000
      Base of data: 0001D000
      Image base: 00400000
      Section alignment: 00001000
      File alignment: 00002000
      OS version (major): 0004
      OS version (minor): 0000
      Image version (major): 0000
      Image version (minor): 0000
      Sub system version (major): 0004
      Sub system version (minor): 0000
      Win32 version: 00000000
      Size of image: 0008F000
      Size of headers: 00001000
      Checksum: 00035CFB
      Sub system: 0002 - Windows graphical user interface (GUI) subsystem
      DLL characteristics: 0000
      Size of stack reserve: 00100000
      Size of stack commit: 00001000
      Size of heap reserve: 00100000
      Size of heap commit: 00001000
      Loader flags: 00000000
      Number of RVA: 00000010
**** PE Sections *****
      CRC-32: EA3EE0E7
      MD5: E64AE8A957D5ED7FBEC48B998EBA21C5
----- PE Sections -----
      Section VirtSize VirtAddr PhysSize PhysAddr  Flags
      .text 0001C000 00001000 00000000 00000400 E0000020
      fbskl8p 00006000 0001D000 00000000 00000400 E0000060
      .data 00014000 00023000 00000000 00000400 C0000040
      .rsrc 00001000 00037000 00001000 00000400 40000040
      99cvbjdu 00001000 00038000 00000000 00001400 C0000040
      ut7h7i2x 00022000 00039000 00000000 00001400 E0000020
      znnrn47v 00033000 0005B000 00032381 00001400 E0000060
      tdbkm0a1 00001000 0008E000 00001000 00033800 40000080
**** Import/Export table *****

```

```
--- Export table -----  
--- Import table (libraries: 2) -----  
kernel32.dll (imports: 6)  
  GetModuleHandleA  
  LoadLibraryA  
  GetProcAddress  
  ExitProcess  
  VirtualAlloc  
  VirtualFree  
user32.dll (imports: 1)  
  MessageBoxA  
=====
```

Further information:
It's doing lookups for:
x.roffleaffles.us
x.pennysheet.com
crusade.godhatesfags.com
Tries to connect to IRC servers running on port 7998, and 8080. For 7998, it joins
channel "##GHF" with password "weh4t3youall"
It uses the SYM06-010 exploit.

Further analysis showed that this file was also downloading other malware components.

In this case it was recommended that a range of ports were blocked; to stop the malware phoning home and joining it's IRC channel where it would get new instructions.

Blocks were also put in place on the DNS, so that any requests for the three domain names would be effectively black-holed.

A clean-up script, similar to the VBS one shown earlier in this paper was used to disinfect systems, which were then patched with the required Microsoft update that the malware had used to infect the systems in the first place.

The anti-virus vendor eventually supplied detection and clean-up signatures; however, this took almost three full days from supplying them with the initial [confirmed] malware samples.

A number of other recommendations were also made which included installing early warning systems and improved processes and procedures for dealing with future outbreaks.

Conclusions

Hopefully I have shown you that even if you are faced with a new malware threat that isn't detected by your anti-malware defences you can still, in most cases, find the infection, how it got in, how it communicates and with the right tools and methodologies even remove it safely before your anti-malware vendor comes up with a solution.

I must make clear that this is not a solution to be used by those not already used to handling and combating malware and other related security threats; home users need not apply, however most academic campuses, large businesses and other organisations should already have at least one person [hopefully more than one] who has the required skills and experience to be able to do this. They almost certainly already work in the security team [or a related function] and have a network of colleagues outside of the main security team that they can call on; such as programmers, network specialists, server and desktop support staff. In all these cases there should be full buy-in from management who are regularly kept up to date and who will deal with requests from more resources and handle any backlash from areas that are affected, either by the malware, or are suffering from collateral damage [loss of internet access, etc.].

As with other security threat, especially malware related ones, you need to deploy a multi-layered approach to minimise the chance of malware getting onto your computers. This means not only do you need good technological solutions, and overlapping technologies at that, but these need to be backed up with good security policies, procedures, education and constant vigilance.

Please do not see this paper as an exhaustive or complete look at detecting and combating new malware and malware forensics, to do this real justice would require enough material to fill a large book.

Appendix A – Suggested Reading

- Implementing Anti-Virus [Malware] Controls in the Corporate Arena, (Overton, Martin) - Proceedings of the 16th Compsec International Conference, 1999 pp 575-586
- You are the Weakest Link, Goodbye! – Malware Social Engineering Comes of Age, (Overton, Martin) - Virus Bulletin, March 2002 pp 14-17
- Canning More Than SPAM with Bayesian Filtering, (Overton, Martin) - Virus Bulletin International Conference 2004
- Anti-Malware Tools: Intrusion Detection Systems, (Overton, Martin) - EICAR International Conference 2005
- Bots and botnets - risks, issues and prevention, (Overton, Martin) - Virus Bulletin International Conference 2005
- Spyware: Risks, Issues and Prevention, (Overton, Martin) - EICAR International Conference 2006
- Rootkits - Risks, Issues and Prevention, (Overton, Martin) - Virus Bulletin International Conference 2006
- The Journey, So Far: Trends, Graphs and Statistics, (Overton, Martin) - Virus Bulletin International Conference 2007
- 2007: The Year of the Social Engineer? (Overton, Martin) - Virus Bulletin, January 2008 pp S2-S5
- AVIEN Malware Defense Guide (Harley, David, et al) – Syngress – ISBN 978-1-59749-164-8
- Computer Forensics: Incident Response Essentials (Kruse, Warren and Heiser, Jay) – Addison-Wesley – ISBN 0-201-70719-5
- The Art of Computer Virus Research and Defense (Szor, Peter) – Addison-Wesley – ISBN 0-321-30454-3
- Malware Forensics: Investigating and Analyzing Malicious Code (Aquilina, James, et al) – Syngress – ISBN 978-1-59749-268-3
- Reversing: Secrets of Reverse Engineering (Eilam, Eldad) – Wiley – ISBN 0-7645-7481-7
- Windows Forensic Analysis (Carvey, Harlan) – Syngress – ISBN 978-1-59749-156-3
- Reverse engineering Code with IDA Pro (Kaminsky, Dan, et al) – Syngress – ISBN 978-1-59749-237-9

Appendix B – So, you 'Think' your computer is infected, what should you do?

First question for you is:

Do you have anti-virus installed and enabled, and is it up to date? [Yes, I know that is two questions]

Second question for you is:

Do you have a firewall installed and enabled?

If you have XP then you can use the XP Firewall instead [if you must].

Third question for you is:

Do you have anti-spyware/adware installed and enabled?

Fourth question for you is:

Do you use Windows Update to ensure that your system is fully patched [at least once a week]? A significant number of malware will get onto systems by exploiting known vulnerabilities in the operating system or applications. So, make it harder for them to 'own' your box, update it!

Fifth question for you is:

Do you still use Internet Explorer?

If so, then you are making it easier for adware, spyware and some malware to infect you via your browser, yes Internet Explorer is a 'Holey Browser, Batman'. I would strongly suggest that you use another one such as Firefox or Mozilla instead as it tends to have less holes for the nasties on the web to crawl in through.

Have you noticed the theme yet? No, well just to make it clear; There is NO excuse for not having protection against Malware, Spyware and Hackers installed on that shiny new PC [or that old grubby one for that matter].

So, if you have done all of the above and still think you are infected by something new, proceed to the next section:

Why do you think you are infected?

If the answer is "*my system keeps crashing, behaving badly or won't do what I want it to do...*" then a virus or other malware may be the least likely of your problems. The most likely causes are faulty memory or other hardware component, a corrupted file system (component or data corruption) or software/operating system mis-configuration or dare-I-say-it, "*user error*". So, check these first before jumping to conclusions about being infected.

If you have tried all the above suggestions, and ruled out all the other possibilities listed above, especially the "*end-user*" problem and still think you have a new Pox on your box, then it is time to get a second opinion. Just as you would if you think your Doctor has mis-diagnosed you.

The first step is to use one or more other virus scanners. I would strongly recommend the Kaspersky, BitDefender, McAfee and TREND ones for starters.

Online Virus Scanners:

<http://www.bitdefender.com/scan/licence.php> *BitDefender*
<http://housecall.trendmicro.com/> *TREND*
<http://www.pandasoftware.com/activescan/> *Panda*
<http://us.mcafee.com/root/mfs/default.asp> *McAfee*
<http://www.kaspersky.com/remoteviruschk.html> *Kaspersky*
<http://www.ravantivirus.com/scan> *RAV*
<http://security.symantec.com/sscv6/home.asp> *Symantec*

