

NORTON ANTI-VIRUS 2009 VERSUS MICROSOFT SECURITY ESSENTIALS: A COMPARATIVE ANTI-MALWARE TEST

Dennis Technology Lab, 26/08/2009

This test aims to compare the effectiveness of free anti-virus programs with Norton AntiVirus 2009 and Microsoft Security Essentials.

Both products have been exposed to genuine internet threats that real customers could have encountered during the test period. Crucially, this exposure was carried out in a realistic way, reflecting a customer's experience as closely as possible. For example, each test system visits websites, downloads files and receives email messages exactly as an average user would.

Products are awarded marks for detecting real threats and providing adequate protection.

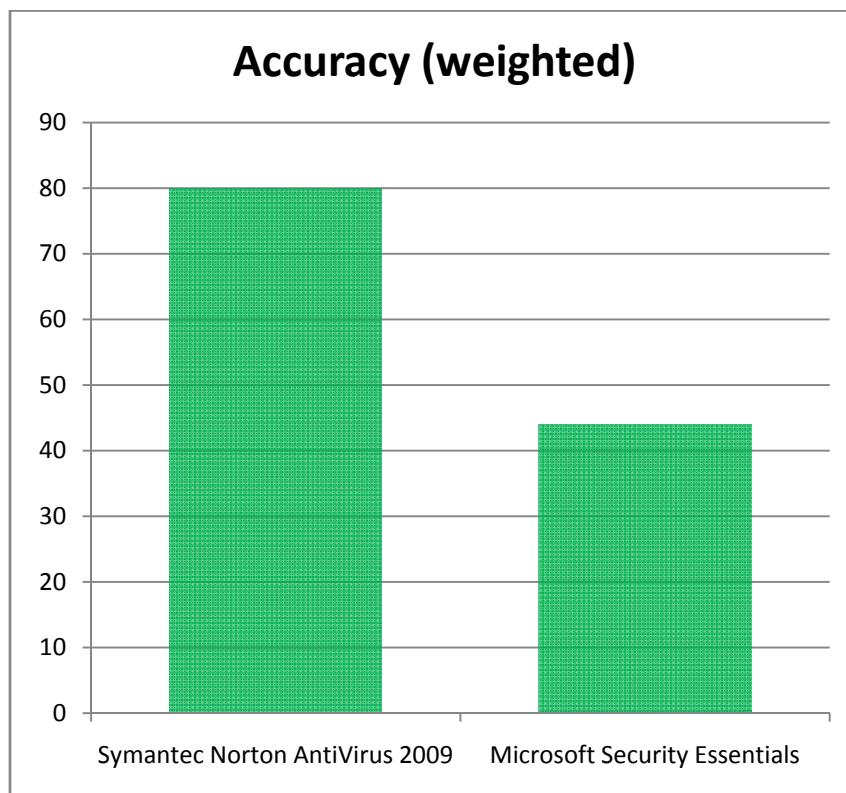
TABLE OF CONTENTS

1. Overall Accuracy <i>Results weighted according to different levels of effectiveness.</i>	2	6. Terms <i>Definitions of technical terms used in the report.</i>	10
2. Overall Protection <i>Combined, un-weighted protection results.</i>	3	Appendix A: Malicious Samples <i>A full list of the source of malware used in the test.</i>	11
3. Protection Details <i>Results categorized according to different levels of protection.</i>	4	Appendix B: Threat Report <i>Detailed notes about each product's reaction to a sample.</i>	13
4. The Tests <i>An overview of the testing techniques used.</i>	5	Appendix C: Tools <i>Reference to software tools used to run the test.</i>	19
5. Test Details <i>Technical details of the testing methodology.</i>	6		

1. OVERALL ACCURACY

Each product has been scored for accuracy. We have awarded two points for defending against a threat, one for neutralizing it and removed two points every time a product allowed a compromise of the system.

Product	Defended Threat	Neutralized Threat	Target Compromised	Overall Accuracy
Symantec Norton AntiVirus 2009	45	0	5	80
Microsoft Security Essentials	33	4	13	44

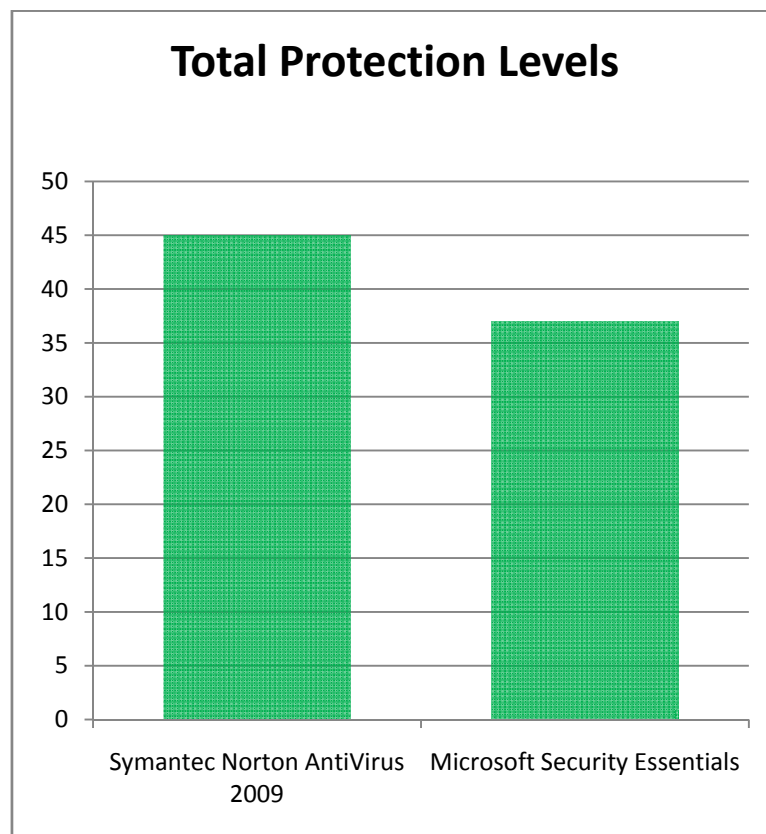


Norton AntiVirus 2009's higher score is due to its ability to block more threats before they could start to install themselves.

2. OVERALL PROTECTION

The following illustrates the general level of protection provided by each of the security products, combining the defended and neutralized incidents into an overall figure. This figure is not weighted with an arbitrary scoring system, as above.

Product	Defended Threat	Neutralized Threat	Overall Protection
Symantec Norton AntiVirus 2009	45	0	45
Microsoft Security Essentials	33	4	37

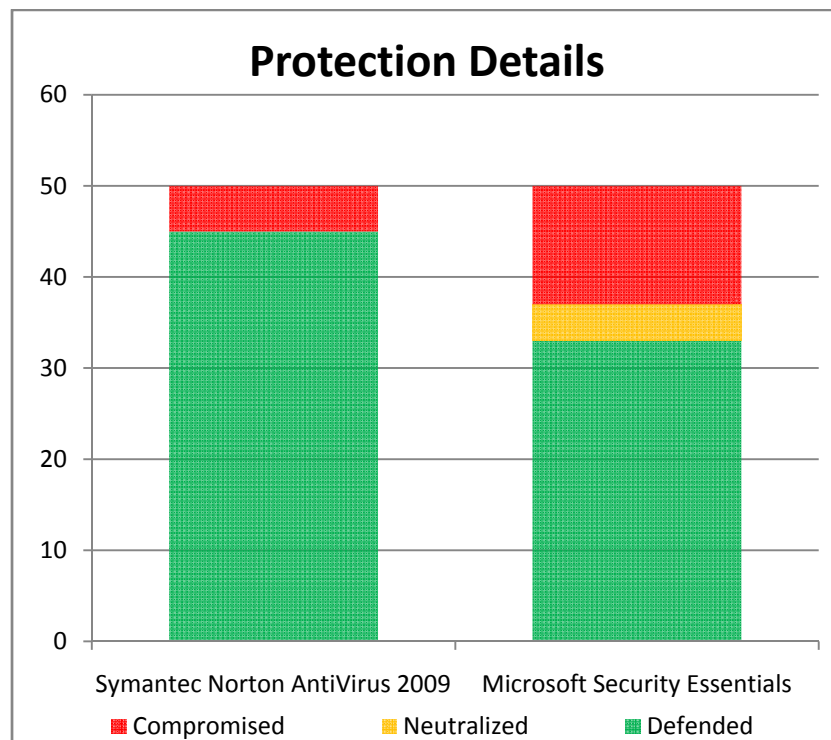


Norton AntiVirus 2009 prevented more malware from infecting the test systems than did Microsoft Security Essentials

3. PROTECTION DETAILS

The security products provided different levels of protection. When a product **defended** against a threat, it prevented the malware from gaining a foothold on the target system. A threat might have been able to infect the system and, in some cases, the product **neutralized** it later. When it couldn't, the system was **compromised**.

Product	Defended Threat	Neutralized Threat	Target Compromised
Symantec Norton AntiVirus 2009	45	0	5
Microsoft Security Essentials	33	4	13



Norton AntiVirus 2009's 'Defended' score (in green) shows a consistently strong ability to prevent malware from running

4. THE TESTS

4.1 The threats

Providing a realistic user experience was important in order to illustrate what really happens when a user encounters a threat on the internet. For example, in these tests web-based malware was accessed by visiting an original, infected website using a web browser, and not downloaded from a CD or internal test website.

All target systems were fully exposed to the threats. This means that malicious files were run and allowed to perform as they were designed, subject to checks by the installed security software. A minimum time period of five minutes was provided to allow the malware an opportunity to act.

4.2 Test rounds

Tests were conducted in rounds. Each round recorded the exposure of every product to a specific threat. For example, in 'round one' both of the products were exposed to the same malicious website. In 'round 49' each product downloaded the same email complete with an infected attachment.

At the end of each round the test systems were completely reset to remove any possible trace of malware before the next test began.

Incident	Product	Remediation			
		Complete	Defended	Neutralized	Compromised
42	Symantec Norton AntiVirus 2009	1	1		
42	Microsoft Security Essentials	1	1		
43	Symantec Norton AntiVirus 2009	1	1		
43	Microsoft Security Essentials	1	1		
44	Symantec Norton AntiVirus 2009	1	1		
44	Microsoft Security Essentials	1		1	
45	Symantec Norton AntiVirus 2009	1	1		
45	Microsoft Security Essentials	1		1	
46	Symantec Norton AntiVirus 2009	1	1		
46	Microsoft Security Essentials				1
47	Symantec Norton AntiVirus 2009	1	1		
47	Microsoft Security Essentials	1	1		
48	Symantec Norton AntiVirus 2009	1	1		
48	Microsoft Security Essentials	1	1		
49	Symantec Norton AntiVirus 2009	1	1		
49	Microsoft Security Essentials	1	1		
50	Symantec Norton AntiVirus 2009	1	1		
50	Microsoft Security Essentials	1	1		

Each 'round' exposed every product to one specific threat. In this case, rounds 42 to 50 show a range of responses to a particular threat

4.3 Monitoring

Close logging of the target systems was necessary to gauge the relative successes of the malware and the anti-malware software. This included recording activity such as network traffic, the creation of files and processes and changes made to important files.

4.4 Levels of protection

The products displayed different levels of protection. Sometimes a product would prevent a threat from executing, or at least making any significant changes to the target system. In other cases a threat might be able to perform some tasks on the target, after which the security product would intervene and remove some or all of the malware. Finally, a threat may be able to bypass the security product and carry out its malicious tasks unhindered. It may even be able to disable the security software. Occasionally Windows' own protection system defends against a threat, while the anti-virus program can ignore it. Another outcome is that the malware may crash for various reasons. The different levels of protection provided by each product were recorded following analysis of the log files.

4.5 Types of protection

All of the products tested provided two main types of protection: real-time and on-demand. Real-time protection monitors the system constantly in an attempt to prevent a threat from gaining access. On-demand protection is essentially a 'virus scan' that is run by the user at an arbitrary time.

The test results note each product's behaviour as a threat is introduced and afterwards. The real-time protection mechanism was monitored throughout the test, while an on-demand scan was run towards the end of each test to measure how safe the product determined the system to be.

5. TEST DETAILS

5.1 The targets

To create a fair testing environment, each product was installed on a clean Windows XP Professional target system. The operating system was updated with Windows XP Service Pack 2 (SP2), although no later patches or updates were applied. The high prevalence of internet threats that rely on Internet Explorer 6, and other vulnerable Windows components that have been updated since SP2 was released, suggest that there are many systems with this level of patching currently connected to the internet. We used this level of patching to remain as realistic as possible.

A selection of legitimate but old software was pre-installed on the target systems. These posed security risks, as they contained known vulnerabilities.

A different security product was then installed on each system. Each product's update mechanism was used to download the latest version with the most recent definitions and other elements. Due to the dynamic nature of the tests, which are carried out in real-time with live malicious websites, the products' update systems were allowed to run automatically and were also run manually before each test round was carried out. The products were also allowed to 'call home' should they be programmed to query databases in real-time.

Microsoft's update servers were blocked using a web proxy. This technique prevented the systems from installing further Windows updates accidentally during the course of the test. Windows Automatic Updates was disabled as a further precaution. However, Microsoft Security Essentials was permitted access to servers necessary for it to receive updates and answers to queries sent in real time.

Each target system contained identical hardware, including an Intel Core 2 Duo processor, 1GB RAM, a 160GB hard disk and a DVD-ROM drive. Each was connected to the internet via its own virtual network (VLAN) to avoid malware cross-infecting other targets.

5.2 Threat selection

The internet threats were selected by Dennis Technology Lab (DTL) from a number of sources and based on particular criteria. To achieve a good spread of threats that users were actively encountering, we used a number of systems to gather suspicious web links (URLs). These sources include DTL's own email honeypots, which are designed to receive non-legitimate email, and a feed of suspicious URLs supplied daily by MessageLabs, which is part of Symantec.

Only three of the fifty samples were extracted from MessageLabs data. Most of the URLs were picked from lists generated by DTL's own malicious site detection system, which uses popular search engine keywords submitted to Google. It analyses sites that are returned in the search results from a number of search engines and adds them to a database of malicious websites. In all cases, a control system was used to confirm that the URLs linked to actively malicious sites.

5.3 Test stages

There were three main stages in each individual test:

1. Introduction
2. Observation
3. Remediation

During the *Introduction* stage, the target system was exposed to a threat. Before the threat was introduced, a snapshot was taken of the system. This created a list of Registry entries and files on the hard disk. We used Regshot (see **Appendix D: Tools**) to take and compare system snapshots. The threat was then introduced.

Immediately after the system's exposure to the threat, the *Observation* stage is reached. During this time, which typically lasted at least 10 minutes, the tester monitored the system both visually and using a range of third-party tools. The tester reacted to pop-ups and other prompts according to the directives described below (see **5.6 Observation and intervention, page 8**).

In the event that hostile activity to other internet users was observed, such as when spam was being sent by the target, this stage was cut short. The *Observation* stage concluded with another system snapshot. This 'infected' snapshot was compared to the original 'clean' snapshot and a report generated. The system was then rebooted.

The *Remediation* stage is designed to test the products' ability to clean an infected system. If it defended against the threat in the *Observation* stage then there should be few (if any) legitimate alerts during this procedure. An on-demand scan was run on the target, after which a 'scanned' snapshot was taken. This was compared to the original 'clean' snapshot and a report was generated. All log files, including the snapshot reports and the product's own log files, were removed from the target. The target was then reset to a clean state, ready for the next test.

5.4 Threat introduction

Malicious websites were visited in real-time using Internet Explorer, while infected email attachments were downloaded using Outlook Express. All of this risky behavior was conducted using live internet connections. URLs were typed manually into Internet Explorer's address bar, while email attachments were extracted from their original emails and loaded onto a private POP3 email server hosted on the test network.

In cases where email-borne malware was detected by a product before a user could access it (e.g. it removed an infected attachment and added a text-based warning to the body of the email), the product is considered to have defended against the threat successfully. If the attachment could be downloaded and run, it was executed. In such cases, the test continued in the same way as with the tests featuring web-based threats.

Web-hosted malware often changes over time. Visiting the same site over a short period of time can expose systems to what appear to be a range of threats (although it may be the same threat, slightly altered to avoid detection). In order to improve the chances that each target system received the same experience from a malicious web server, we used a caching proxy set to 'offline' mode. When the first target system visited a site, the page's content, including malicious code, was downloaded and stored. When each consecutive target system visited the site, it should have received the same content, with some provisos.

In this test we noted that many sites were only malicious for short periods of time, often due to the use of iframes, which point to other sites. In some cases the offline proxy allowed the tests to continue but in a significant number of cases the sites became fully unavailable. In such cases we abandoned the test round and restarted with an alternative URL.

5.5 Secondary downloads

Established malware may attempt to download further files (secondary downloads), which will also be cached and re-served to other targets in some circumstances. These circumstances include cases where:

1. The download request is made using HTTP (e.g. `http://badsite.example.com/...`) and
2. The same filename is requested each time (e.g. `badfile1.exe`)

There are scenarios where target systems will receive different secondary downloads. These include cases where:

1. The download request is made using HTTPS or a non-web protocol such as FTP or
2. A different filename is requested each time (e.g. badfile2.exe; random357.exe) or
3. The same filename is requested over HTTP but the file has been modified on the web server. In this case even the original download may differ between target systems

5.6 Observation and intervention

Throughout each test, the target system was observed both manually and in real-time. This enabled the tester to take comprehensive notes about the system's perceived behaviour, as well as to compare visual alerts with the products' log entries. At certain stages the tester was required to act as a regular user. To achieve consistency, the tester followed a policy for handling certain situations including dealing with pop-ups displayed by products or the operating system, system crashes, invitations by malware to perform tasks and so on.

This user behaviour policy included the following directives:

1. Act naively. Allow the threat a good chance to introduce itself to the target by clicking OK to malicious prompts, for example.
2. Don't be too stubborn in retrying blocked downloads. If a product warns against visiting a site, don't take further measures to visit that site.
3. Where malware is downloaded as a Zip file, or similar, extract it to the Desktop then attempt to run it. If the archive is protected by a password, and that password is known to you (e.g. it was included in the body of the original malicious email), use it.
4. Always click the default option. This applies to security product pop-ups, operating system prompts (including Windows firewall) and malware invitations to act.
5. If there is no default option, wait. Give the prompt 20 seconds to choose a course of action automatically.
6. If no action is taken automatically, choose the first option. Where options are listed vertically, choose the top one. Where options are listed horizontally, choose the left-hand one.

5.7 Remediation

When a target is exposed to malware, the threat may have a number of opportunities to infect the system. The security product also has a number of chances to protect the target. The snapshots explained in 5.3 Test stages (see page 6) provided information that was used to analyse a system's final state at the end of a test.

Before, during and after each test, a 'snapshot' of the target system was taken to provide information about what had changed during the exposure to malware. For example, comparing a snapshot taken before a malicious website was visited to one taken after might highlight new entries in the Registry and new files on the hard disk. Snapshots were also used to determine how effective a product was at removing a threat that had managed to establish itself on the target system. This analysis gives an indication as to the levels of protection that a product has provided.

These levels of protection have been recorded using three main terms: defended, neutralized, and compromised. A threat that was unable to gain a foothold on the target was *defended* against; one that was prevented from continuing its activities was *neutralized*; while a successful threat was considered to have *compromised* the target.

A defended incident occurs where no malicious activity is observed with the naked eye or third-party monitoring tools following the initial threat introduction. The snapshot report files are used to verify this happy state.

If a threat is observed to run actively on the system, but not beyond the point where an on-demand scan is run, it is considered to have been neutralized. Comparing the snapshot reports should show that malicious files were created and Registry entries were made after the introduction. However, as long as the 'scanned' snapshot report shows that either the files have been removed or the Registry entries have been deleted, the threat has been neutralized.

The target is compromised if malware is observed to run after the on-demand scan. In some cases a product will request a further scan to complete the removal. For this test we considered that secondary scans were acceptable, but further scan requests would be ignored. Even if no malware was observed, a compromise result was recorded if snapshot reports showed the existence of new, presumably malicious files on the hard disk, in conjunction with Registry entries designed to run at least one of these files when the system booted. An edited 'hosts' file or altered system file also counted as a compromise.

5.8 Automatic monitoring

Logs were generated using third-party applications, as well as by the security products themselves. Manual observation of the target system throughout its exposure to malware (and legitimate applications) provided more information about the security products' behaviour. Monitoring was performed directly on the target system and on the network.

Client-side logging

A combination of Process Explorer, Process Monitor, TcpView and Wireshark were used to monitor the target systems. Regshot was used between each testing stage to record a system snapshot. A number of DTL-created scripts were also used to provide additional system information. Each product was able to generate some level of logging itself.

Process Explorer and TcpView were run throughout the tests, providing a visual cue to the tester about possible malicious activity on the system. In addition, Wireshark's real-time output, and the display from the web proxy (see Network logging, below), indicated specific network activity such as secondary downloads.

Process Monitor also provided valuable information to help reconstruct malicious incidents. Both Process Monitor and Wireshark were configured to save their logs automatically to a file. This reduced data loss when malware caused a target to crash or reboot.

In-built Windows commands such as 'systeminfo' and 'sc query' were used in custom scripts to provide additional snapshots of the running system's state.

GMER was available to the testers, but did not provide any useful information that was not already gathered by the tools above.

Network logging

All target systems were connected to the internet. The local network was configured with VLANs to avoid cross-infection and other interference.

The internet connection incorporated a transparent web proxy and network monitoring system. All traffic to and from the internet had to pass through this system. Further to that, all web traffic had to pass through the proxy as well. This allowed the tester to capture files containing the complete network traffic. It also provided a quick and easy view of web-based traffic, which was displayed to the tester in real-time.

The network monitor was a dual-homed Linux system running as a transparent router, passing all web traffic through a Squid proxy. This was configured in 'offline' mode during testing, with additional rules to prevent access to certain Microsoft update sites.

6. TERMS

Compromised	Malware continues to run on an infected system, even after an on-demand scan.
Defended	Malware was prevented from running on, or making changes to, the target.
Introduction	Test stage where a target system is exposed to a threat.
Neutralized	Malware was able to run on the target, but was then removed by the security product.
Observation	Test stage during which malware may affect the target.
On-demand (protection)	Manual 'virus' scan, run by the user at an arbitrary time.
Prompt	Questions asked by software, including malware, security products and the operating system. With security products, prompts usually appear in the form of pop-up windows. Some prompts don't ask questions but provide alerts. When these appear and disappear without a user's interaction, they are called 'toasters'.
Real-time (protection)	The 'always-on' protection offered by many security products.
Remediation	Test stage that measures a product's abilities to remove any installed threat.
Round	Test series of multiple products, exposing each target to the same threat.
Snapshot	Record of a target's file system and Registry contents.
Target	Test system exposed to threats in order to monitor the behavior of security products.
Threat	A program or other measure designed to subvert a system.
Update	Code provided by a vendor to keep its software up to date. This includes virus definitions, engine updates and operating system patches.

APPENDIX A: MALICIOUS SAMPLES

Incident	Source	Mechanism	Delivery	Details
1	DTL wHP1	HTTP	URL	wap-cat.ru
2	DTL wHP1	HTTP	URL	rapidsharefiles.net
3	DTL wHP1	HTTP	URL	marksesl.com/village.html
4	DTL wHP1	HTTP	URL	ryera.ifrance.com/regulation-d-affect-banks.html
5	DTL wHP1	HTTP	URL	http://anglingalliance.com/tpl/inc/style/image5341.html
6	DTL wHP1	HTTP	URL	blobabe-zip.isuisse.com/html/uomo-maturo
7	DTL wHP1	HTTP	URL	http://gcadvocate.org
8	ML HP	HTTP	Email URL	yahoo-cartoes.webcindario.com/Cartoes-Amor-Saudades.com
9	DTL wHP1	HTTP	URL	wrfm.de/filme/film1993.php
10	DTL wHP1	HTTP	URL	www.eguru.com
11	DTL wHP1	HTTP	URL	www.moldinstruction.com
12	DTL wHP1	HTTP	URL	www.zoopolitico.it/italia/ministro-mara-t182811.html
13	DTL wHP1	HTTP	URL	www.nordmarx.com/dvd.dvd.htm
14	DTL wHP1	HTTP	URL	mohoar.pytalhost.net/comment-665.html
15	DTL wHP1	HTTP	URL	mancos.lib.co.us/img/prepaid-credit-card-business.html
16	DTL wHP1	HTTP	URL	www.islandbusinesslink.com
17	DTL wHP1	HTTP	URL	fivestarbabes.com/Dana_Lightspeed
18	DTL wHP1	HTTP	URL	hotpornogames.com
19	DTL wHP1	HTTP	URL	danve.1gb.bg/136-slum-village.html
20	DTL wHP1	HTTP	URL	ledecorfrancais.com
21	DTL wHP1	HTTP	URL	eblis666.8m.net/Story/37.html
22	DTL wHP1	HTTP	URL	personalvaluation.com
23	DTL wHP1	HTTP	URL	www.al-hoty.com
24	DTL wHP1	HTTP	URL	www.buckrogers.org
25	DTL wHP1	HTTP	URL	diablo2guide.com
26	DTL wHP1	HTTP	URL	megaporntube.com
27	DTL wHP1	HTTP	URL	deadspin.org
28	DTL wHP1	HTTP	URL	121.126.171.132/.../007341.exe
29	ML HP	HTTP	Email URL	http://visualizar89219.webcindario.com/fotos.com
30	DTL wHP1	HTTP	URL	wipmsig.org

31	DTL wHP1	HTTP	URL	http://gameend.3322.org/42.exe?frandom=8321
32	DTL wHP1	HTTP	URL	http://www.dresseur.com/preAdmission/001_test.php
33	DTL wHP1	HTTP	URL	www.sportplanes.com
34	DTL wHP1	HTTP	URL	www.beyonce-central.com
35	DTL wHP1	HTTP	URL	hairywomenpayperview.coim
36	DTL wHP1	HTTP	URL	www.newtorrents.info/search/porn
37	DTL wHP1	HTTP	URL	zestyincest.com
38	ML HP	HTTP	Email URL	http://visualizar89219.webcindario.com/fotos.com
39	DTL wHP1	HTTP	URL	bilalphilips.com
40	DTL wHP1	HTTP	URL	oxfordhotel.co.sg/oxford
41	DTL wHP1	HTTP	URL	appartmentwladyslawowo.pl
42	DTL wHP1	HTTP	URL	http://www.linux.co.th/images/foto789.exe
43	DTL wHP1	HTTP	URL	home.no/andrewbrownjf/minotar.html
44	DTL wHP1	HTTP	URL	www.barriozona.com
45	DTL wHP1	HTTP	URL	fuelwaterseparator.net
46	DTL wHP1	HTTP	URL	kungfurodeo.com
47	DTL wHP1	HTTP	URL	bigbendopg.org
48	DTL wHP1	HTTP	URL	www.purdie.co.uk
49	DTL eHP1	Email	Attachment	UPSNR_be23c5be
50	DTL wHP1	HTTP	URL	from-yesterday.org/plutonium/

APPENDIX B: THREAT REPORT

The following table is a summary. The full report is available as an Excel spreadsheet, which includes the Notes referred to in some Report entries.

	INTRODUCTION				MANUAL SCAN			Complete Remediation	Defended	Neutralized	Compromised
	Product	Alert	Effect	Report	Alert	Effect	Report				
1	Symantec Norton AntiVirus 2009	Toaster	Blocked					█	█		
1	Microsoft Security Essentials	Pop-up	prompt asked for Remove option. Succeeded.	Multiple - see notes				█	█		
2	Symantec Norton AntiVirus 2009	Toaster	Blocked	Auto-Protect has detected Backdoor Trojan	Tracking cookie			█	█		
2	Microsoft Security Essentials	none	none	none	Report	requires Apply All	Trojan: Win 32/Alureon.gen!] and Trojan Downloader: JSE Renos	█		█	
3	Symantec Norton AntiVirus 2009	Toaster	Blocked	JS.Downloader	Tracking cookie			█	█		
3	Microsoft Security Essentials	Toaster	Suspended (Clean computer)	Trojan:Win32/Want	None			█	█		
4	Symantec Norton AntiVirus 2009	Toaster	Blocked	Auto-Protect has detected Trojan Fakealert	Tracking Cookies have been detected	Prompt to fix	OK	█	█		
4	Microsoft Security Essentials	Toaster	none	Potential threat details: Trojan Downloader: JS/Renos	none	n/a	n/a	█	█		
5	Symantec Norton AntiVirus 2009	Toaster	Blocked	Trojan.Fakeavalert	Tracking cookie			█	█		
5	Microsoft Security Essentials	Toaster	Suspended	TrojanDownloader:JS/Renos; Trojan:Win32/Winwebsec (after user interaction)	None			█	█		
6	Symantec Norton AntiVirus 2009	Toaster	Blocked	Trojan Fakealert; JS Downloader; JS Downloader; Internet AntiVirus	Tracking Cookie	prompt to Fix	OK	█	█		
6	Microsoft Security Essentials	Toaster	prompt asked for Remove option.	Trojan: JS/FakeIA	none	n/a	n/a	█	█		

			Succeeded.									
7	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Acrobat PDF Suspicious File Download 3	Tracking cookie							
7	Microsoft Security Essentials	Toaster	Blocked	Exploit:Win32/Pdfjsc.AV	None (one of the two PDFs remained on the desktop)							
8	Symantec Norton AntiVirus 2009	Toaster	Blocked	Suspicious.MH690.A detected by Auto-Protect.	Tracking cookie							
8	Microsoft Security Essentials											
9	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Malicious Toolkit Variant Activity	none	n/a	n/a					
9	Microsoft Security Essentials	Toaster	Remove prompt; succeeded	VirTool: JS/Obfuscator	none	n/a	n/a					
10	Symantec Norton AntiVirus 2009	none	n/a	n/a	none	n/a	n/a					
10	Microsoft Security Essentials	Toaster	Remove prompt; succeeded		none	n/a	n/a					
11	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Malicious Toolkit Variant Activity 2	Tracking Cookie	Prompt to fix	OK					
11	Microsoft Security Essentials	Toaster	Remove prompt; succeeded	Trojan Downloader: Win32/Brodolab.X	none	n/a	n/a					
12	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Misleading Application File Download								
12	Microsoft Security Essentials	Toaster	Blocked	Trojan.JS/FakeIA								
13	Symantec Norton AntiVirus 2009	Toaster	Blocked	Auto Protect Tojan Maliframe!html	Tracking Cookie	prompt to Fix	OK					
13	Microsoft Security Essentials	none	n/a	n/a	No threats reported	n/a	n/a					
14	Symantec Norton AntiVirus 2009	Toaster	Blocked	Auto-Protect has detected JS.Downloader. Auto-Protect has detected Trojan.Fakeavalert. Auto-Protect has detected JS.Downloader. Auto-Protect has detected InternetAntivirus			Found tracking cookie					
14	Microsoft Security Essentials	Toaster	Blocked	Trojan:JS/FakeIA			Found nothing					
15	Symantec Norton AntiVirus 2009	Toaster	Blocked	MSIE ADODB Stream Object File Installation Weakness	none	n/a	n/a					
15	Microsoft Security Essentials	none	n/a	n/a	none	n/a	n/a					
16	Symantec Norton	Toaster	Blocked	HTTP Acrobat PDF Suspicious File	Tracking	prompt to Fix	OK					

	AntiVirus 2009			Download 3; Auto Protect bloced security risk downloads.	Cookie						
16	Microsoft Security Essentials	Toaster	Remove prompt; succeeded	exploit: Win32/Pdfjcc.A; TrojanDropper: Win32/Preald.3	none	n/a	n/a				
17	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Malicious Toolkit Variant Activity 2			Found 12 tracking cookies				
17	Microsoft Security Essentials						Found nothing				
18	Symantec Norton AntiVirus 2009	Toaster	Blocked	Auto-Protect has detected JS.Downloader.			Found tracking cookie				
18	Microsoft Security Essentials	Toaster	Blocked	Trojan:Win32/Wantvi.I TrojanDownloader:Win32/Swif.M							
19	Symantec Norton AntiVirus 2009	Toaster	Blocked	Trojan Fakeavalert; Tracking Cookie detected	none	n/a	n/a				
19	Microsoft Security Essentials	Toaster	Remove prompt; succeeded	Trojan Downloader: JS/Renos	none	n/a	n/a				
20	Symantec Norton AntiVirus 2009	Toaster	Blocked	MSIE ADODB.Stream Object File Installation Weakness	Tracking Cookie	prompt to Fix	OK				
20	Microsoft Security Essentials	none	n/a	n/a	none	n/a	n/a				
21	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP MS Unsafe ActiveX Obj Instantiation			Found tracking cookie				
21	Microsoft Security Essentials	Toaster	Blocked	Virus:VBS/Mondezimia.gen TrojanDownloader:Win32/Agenttiny Trojan:VBS/Small							
22	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Acrobat PDF Suspicious File Download 3			Found nothing				
22	Microsoft Security Essentials	Toaster	Blocked	Trojan:JS/Redirector.AQ			Found nothing				
23	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Malicious Toolbar Variant Activity 2	Tracking Cookie	prompt to Fix	OK				
23	Microsoft Security Essentials	Toaster	Remove prompt; succeeded	TrojanDownloader: Win32/Dontovo.A	none	n/a	n/a				
24	Symantec Norton AntiVirus 2009	none	n/a	n/a	Tracking Cookie	prompt to Fix	OK				
24	Microsoft Security Essentials	Toaster	prompt to Remove/Succeeded	Trojan: JS/Redirecotr.AQ	none	n/a	n/a				
25	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Acrobat PDF Suspicious File Download 2			Found nothing				
25	Microsoft Security Essentials	none	n/a	n/a			Found nothing				
26	Symantec Norton AntiVirus 2009	Toaster	Blocked	An intrusion attempt by TESTPC was blocked			Found nothing				
26	Microsoft Security Essentials	Toaster	Blocked	TrojanDownloader:HTML/Adodb.gen!B			Found nothing				
27	Symantec Norton AntiVirus 2009	Toaster	Blocked	MSIE ADODB.Stream Object File Installation Weakness			Found nothing				
27	Microsoft Security	Toaster	Blocked	VirTool:Win32/Obfuscator.FI							

	Essentials												
28	Symantec Norton AntiVirus 2009	Toaster	Blocked	Auto-Protect has detected Downloader Tracking Cookie detected by Virus scanner									
28	Microsoft Security Essentials	none	n/a	n/a			Found nothing						
29	Symantec Norton AntiVirus 2009	Toaster	Removed	Suspicious MH690.A	none	n/a	n/a						
29	Microsoft Security Essentials	Toaster	prompt to Remove/Succeeded	TrojanDownloader: Win32/Small.gen!AO	none	n/a	n/a						
30	Symantec Norton AntiVirus 2009	Toaster	Blocked	MSIE ADODB.Stream Object File Installation Weakness			Found nothing						
30	Microsoft Security Essentials	Toaster	Blocked	TrojanDownloader:JS/Psyme.gen			Found nothing						
31	Symantec Norton AntiVirus 2009	none	n/a	n/a	Tracking Cookie	prompt to Fix	OK						
31	Microsoft Security Essentials	Toaster	prompt to Remove/Succeeded	Trojan: Win32/Killer.DK	none	n/a	n/a						
32	Symantec Norton AntiVirus 2009	none	n/a	n/a	Tracking Cookie	prompt to Fix	OK						
32	Microsoft Security Essentials	none	n/a	n/a	none	n/a	n/a						
33	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Malicious Toolkit Variety Activity 2; Pop-up saying "This page cannot be displayed."	Tracking Cookie	prompt to Fix	OK						
33	Microsoft Security Essentials	none	n/a	n/a	none	n/a	n/a						
34	Symantec Norton AntiVirus 2009	Toaster	Blocked	HTTP Malicious Toolbar Variant Activity 2	Tracking Cookie	prompt to Fix	OK						
34	Microsoft Security Essentials	none	n/a	n/a	none	n/a	n/a						
35	Symantec Norton AntiVirus 2009	none					Found nothing						
35	Microsoft Security Essentials	none											
36	Symantec Norton AntiVirus 2009	Toaster	Blocked	MSIE ADODB Stream Object File Installation Weakness	Tracking Cookie	prompt to Fix	OK						
36	Microsoft Security Essentials												
37	Symantec Norton AntiVirus 2009	Toaster	Blocked	MSIE ADODB.Stream Object File Installation Weakness			Found 1 tracking cookie						
37	Microsoft Security Essentials	Toaster	Blocked	Trojan:Win32/Wantvi.I TrojanDownloader:Win32/Swif.M			Found nothing						
38	Symantec Norton AntiVirus 2009	Toasters (2)	Removed/Blocked	MH690A; MH690A	Tracking Cookie	prompt to Fix	OK						
38	Microsoft Security Essentials	Toaster	prompt to Remove/Succeeded	TrojanDownloader: Win32/Small.gen/AO	none	n/a	n/a						
39	Symantec Norton	Toaster	Blocked	HTTP Malicious Toolkit Variant Activity 2									

47	Microsoft Security Essentials	Toaster	Blocked	TrojanDownloader:Win32/Dontovo.A									
48	Symantec Norton AntiVirus 2009	Toaster	Blocked	MSIE MS MPEG2TuneRequestControl ActiveX Instantiation	none	n/a	n/a						
48	Microsoft Security Essentials	Toaster	prompt to Clean/Succeeded	Backdoor: WinNT/Rustock.AN; PWS: WIN32/Zbot.gen!R	Alert (manual)	Recommenation to quarantine/Succeeded	MSE detected 1 potential threat: PWS:Win32/Daurso.A						
49	Symantec Norton AntiVirus 2009	n/a	Windows Popup										
49	Microsoft Security Essentials	Toaster	Blocked	TrojanDownloader:Win32/Bredolab.X									
50	Symantec Norton AntiVirus 2009	Toaster	Blocked	MSIE ADODB.Stream Object Fill Installation Weakness	Tracking Cookie	prompt to Fix	OK						
50	Microsoft Security Essentials	Toaster	prompt to Clean/Succeeded	PWS: Win32/Codtree.Gen!A/Spammer: Win32/Tedroo.l	none	n/a	n/a						

APPENDIX C: TOOLS

Windows Command-Line Tools

Those used included 'systeminfo' and 'sc query'. The systeminfo command "enables an administrator to query for basic system configuration information". The sc command is "used for communicating with the NT Service Controller and services. Running 'sc query' with various options will list services that are running or disabled.

Regshot

<http://sourceforge.net/projects/regshot>

Regshot is an open-source Registry comparison utility that takes a snapshot of the Registry and compares it with a second one.

Wireshark

www.wireshark.org

Wireshark is a network protocol analyzer capable of storing network traffic, including binaries, for later analysis.

Process Monitor

<http://technet.microsoft.com/en-us/sysinternals/bb896645.aspx>

Process Monitor is a monitoring tool that shows real-time file system, Registry and process/thread activity.

Process Explorer

<http://technet.microsoft.com/en-us/sysinternals/bb896653.aspx>

Process Explorer shows information about which handles and DLLs processes have opened or loaded. It also provides a clear and real-time indication when new processes start and old ones stop.

TcpView

<http://technet.microsoft.com/en-us/sysinternals/bb897437.aspx>

TcpView displays network connections to and from the system in real-time.

GMER

www.gmer.net

GMER is an application that detects and removes rootkits.

Squid

www.squid-cache.org

Squid is a caching web proxy that supports HTTP, HTTPS, FTP and other protocols.

Tcpdump

www.tcpdump.org

Tcpdump is a packet capture utility that can create a copy of network traffic, including binaries.

Ebttables

<http://ebttables.sourceforge.net>

The ebttables program is a filtering tool for a bridging firewall. It can be used to force network traffic transparently through the Squid proxy.