Real World Enterprise Security Exploit Prevention Test

March 2015

1 Executive summary

Web browsing is an integral part of both home and corporate internet users' daily activity. The web is almost ubiquitous and people use it for communication, social life, gaming, business, shopping, education, etc. People browse the web very often with outdated software (both at home and in the enterprise), and these outdated applications have known vulnerabilities. Some of these vulnerabilities let the attackers run code on the victim's computer, without any warning on the victim side. After the victim's computer is infected, the attackers can use this malicious code to steal money from their internet banking application, steal credit card data, steal personal information, steal confidential corporate information, or even lock the computer until the victim pays a ransom.

Drive-by download exploits are one of the biggest threats and concerns in an enterprise environment because no user interaction is needed to start the malware on the victim machine. Even traditional, legitimate sites used by enterprises on a daily basis get infected by malware. Flash-based exploits are especially popular among organized criminals because it is a very popular browser plugin. Outdated Flash environments are very "popular" in enterprise environments because of the lack of central management, administrator level privileges needed to update, etc. Exploits and drive-by download attacks are commonly used in Advanced Persistent Threat (APT) attacks as well.

Home users and small-to-medium businesses often lack the knowledge and awareness about exploits, exploit prevention, targeted attacks and the importance of software updates. Big enterprises face the challenge of managing complex IT systems and, consequently, run a high risk of becoming a target of exploit and malware based attacks.

Endpoint protection systems have had a long journey from traditional signature-based protection to that which is implemented in a modern protection system. Advanced heuristics, sandboxing, intrusion prevention systems, URL filtering, cloud based reputation systems, Javascript analysers, memory corruption protection, etc. are now used to combat modern malware threats. In order to test an endpoint protection system, one has to test all modules of the protection employed by that system, and the test has to be done in a way which emulates standard user behaviour accurately. Today, the vast majority of threats are delivered via the web. This is the reason why our test focuses exclusively on web-based exploits. When an endpoint protection system cannot protect its users against malicious software (malware), the damage might be catastrophic. To cite a few examples of threats which can cause catastrophic damage, there is malware which steals confidential information, or malware which can wipe important documents or whole workstations. Attacks like these can cause huge damage to corporate intellectual property or can block business processes for weeks. Our test incorporated a wide range of different malware types, thus emulating a real world scenario as closely as possible.

This assessment was commissioned and sponsored by Kaspersky Lab to serve as an independent efficacy assessment of Kaspersky Endpoint Security (KES) for Windows product and its Automatic Exploit Prevention (AEP) module. KES is an endpoint protection solution, integrating anti-malware solutions like traditional signature matching, proactive defence technologies, cloud reputation services, personal firewall and IPS, etc. The purpose of the AEP module of KES is to monitor the system for known or unknown exploit behaviour and when detected, block the exploit code payload execution.

The objective of this report is to provide an assessment of the ability of KES, with full functionality and the AEP technology inside KES, to prevent drive-by exploitation when KES is installed on an endpoint. In order to put performance in perspective, it was tested alongside six competitor products. Each of the products was installed on an endpoint and tested against 300 unique exploit sites.

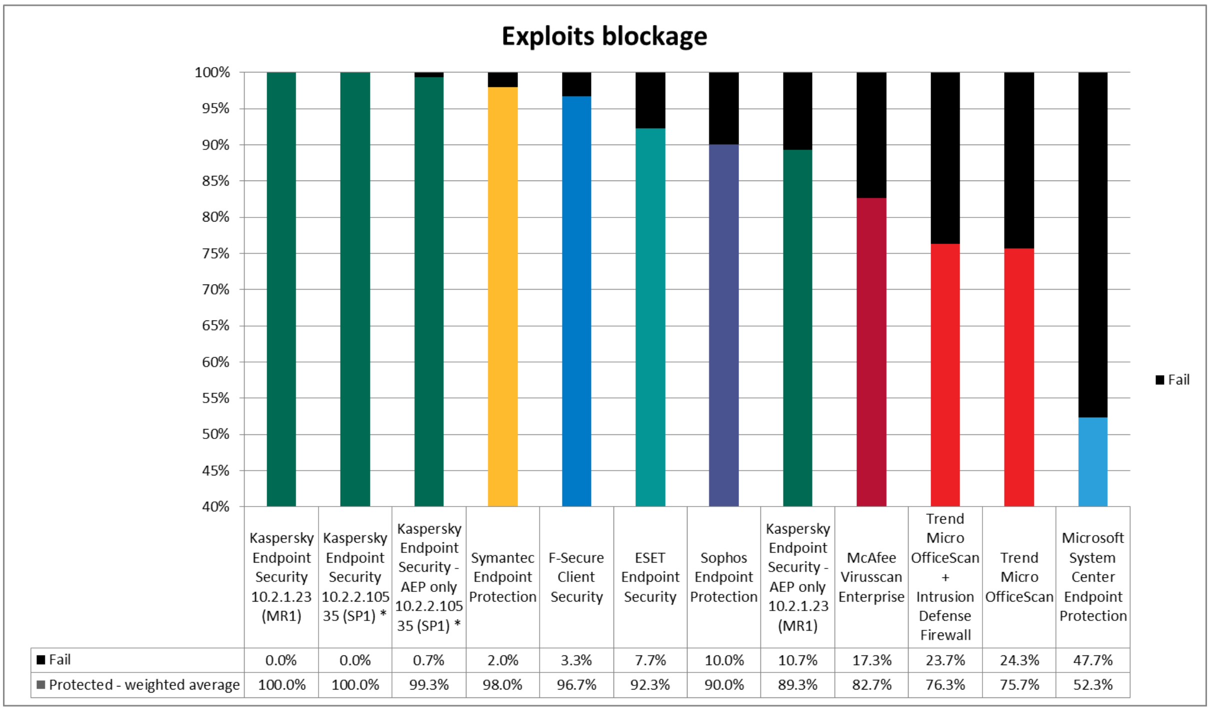
This test is quite unique, based on the followings:

- Large number of in-the-wild exploits (300)

- Diverse set of exploit kits (12)
- Diverse set of exploits (10 different CVEs)
- Internet Explorer, Firefox and Chrome exploits used
- Large number of enterprise endpoint protection systems – 10 products in 12 different configuration
- Use of in-the-wild in memory malware
- Test with 0-day sample
- Minimal delay between exploit acquire and test
- Manual test and result analysis

The final result of the exploit protection test is summarized in the table below. KES SP1 is included in the graph since the weighted average along all test cases is calculated and shown. KES SP1 was released in the middle of the test, thus only test-cases 151-300 were tested with this new release.

(For detailed results along full set of test please refer to Chapter 3.)



	Kaspersky Endpoint Security 10.2.1.23 (MR1)	Kaspersky Endpoint Security 10.2.2.10535 (SP1) *	Kaspersky Endpoint Security - AEP only 10.2.2.10535 (SP1) *	Symantec Endpoint Protection	F-Secure Client Security	ESET Endpoint Security	Sophos Endpoint Protection	Kaspersky Endpoint Security - AEP only 10.2.1.23 (MR1)	McAfee Virusscan Enterprise	Trend Micro OfficeScan + Intrusion Defense Firewall	Trend Micro OfficeScan	Microsoft System Center Endpoint Protection
■ Fail	0.0%	0.0%	0.7%	2.0%	3.3%	7.7%	10.0%	10.7%	17.3%	23.7%	24.3%	47.7%
■ Protected - weighted average	100.0%	100.0%	99.3%	98.0%	96.7%	92.3%	90.0%	89.3%	82.7%	76.3%	75.7%	52.3%

Figure 1 - Result of 300-sample test

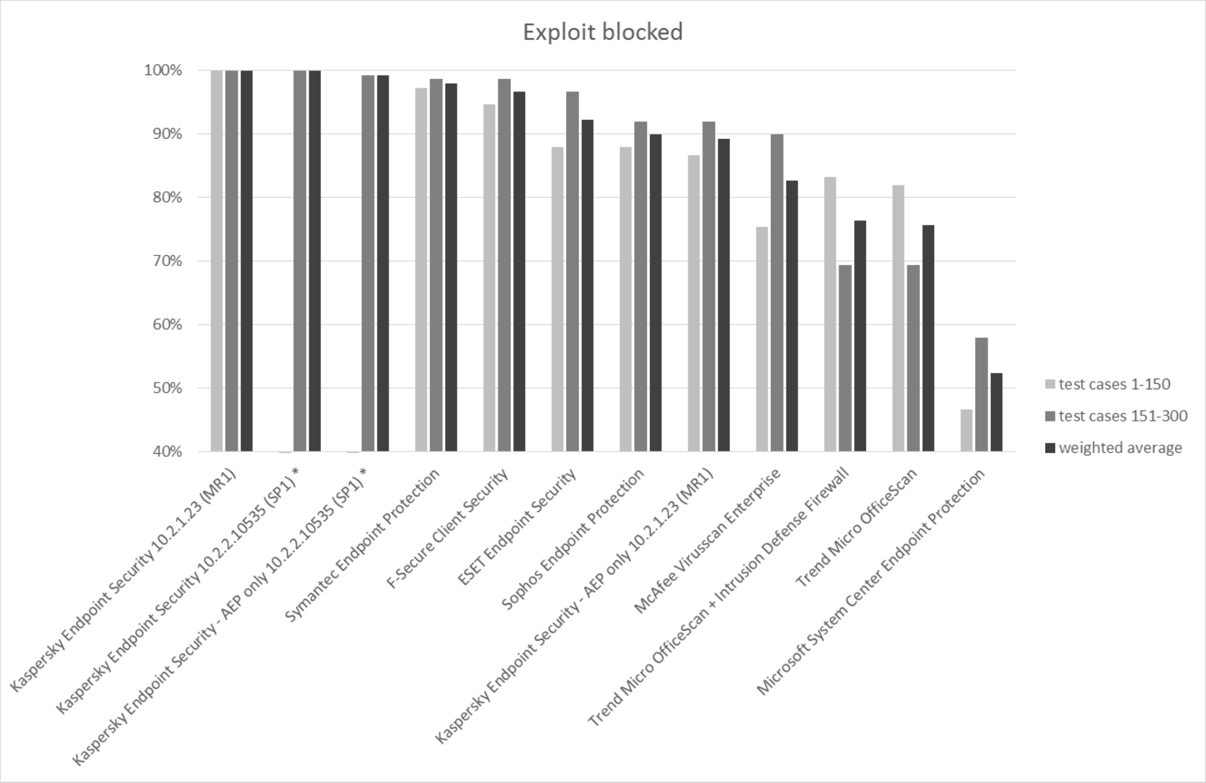


Figure 2 - Detailed results, latest KES added

Product / test cases	1-150	151-300	weighted average
Kaspersky Endpoint Security 10.2.1.23 (MR1)	100.0%	100.0%	100.0%
Kaspersky Endpoint Security 10.2.2.10535 (SP1)	n.a.	100.0%	100.0%
Kaspersky Endpoint Security - AEP only 10.2.2.10535 (SP1)	n.a.	99.3%	99.3%
Symantec Endpoint Protection	97.3%	98.7%	98.0%
F-Secure Client Security	94.7%	98.7%	96.7%
ESET Endpoint Security	88.0%	96.7%	92.3%
Sophos Endpoint Protection	88.0%	92.0%	90.0%
Kaspersky Endpoint Security - AEP only 10.2.1.23 (MR1)	86.7%	92.0%	89.3%
McAfee Virusscan Enterprise	75.3%	90.0%	82.7%
Trend Micro OfficeScan + Intrusion Defense Firewall	83.3%	69.3%	76.3%
Trend Micro OfficeScan	82.0%	69.3%	75.7%
Microsoft System Center Endpoint Protection	46.7%	58.0%	52.3%

The conclusion drawn from the results is that only one endpoint protection systems stand out from the crowd with it's excellent protection:

- Kaspersky Endpoint Security 10

2 Certifications

The following certification is given to Kaspersky Endpoint Security:

3 Test methodology

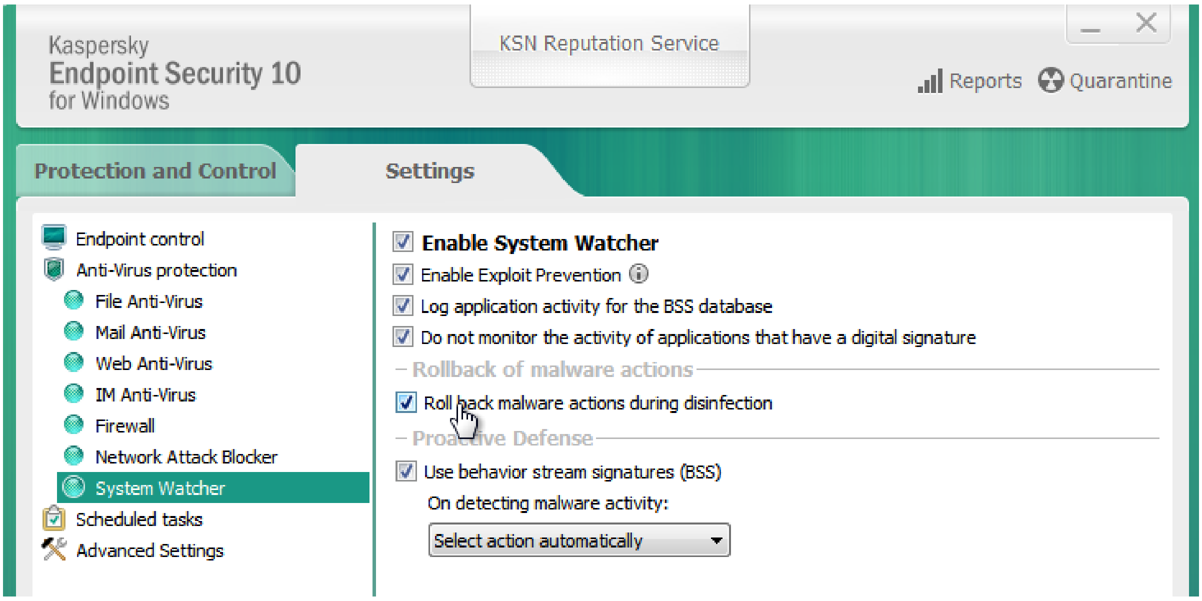
The test was conducted as follows:

1. One default install Windows 7 Enterprise 64 Service Pack 1 virtual machine (VmWare) endpoint was created. (Windows 7 64-bit was the most popular OS for the target audience.) The default HTTP/HTTPS proxy was configured to point to a proxy running on a different machine. SSL/TLS traffic was intercepted on the proxy, and AV's have been either configured to skip the proxy, or the SSL decryption was disabled for the AV's update/cloud connections.
2. The security of the OS was weakened by the following actions:
    a. Microsoft Defender was disabled
    b. Internet Explorer Smartscreen was disabled
3. The following vulnerable software was installed, based on 2014 Q3 statistics:
    a. Java 1.7.0.17
    b. Adobe Reader 9.3.0
    c. Flash Player 15.0.0.152 or Flash Player 16.0.0.287 in a small number of cases
    d. Silverlight 5.1.10411.0
    e. Internet Explorer 8.0.7601.17514
    f. Firefox 33.1.1
    g. Chrome 38.0.2125.101

    These version numbers were specified with the following two requirements:

    1. The highest number of in-the-wild exploits should be able to exploit this specific version, thus increasing the coverage of the tests.
    2. The version must currently be popular among users.
4. Windows Update was disabled.
5. From this point, 13 different snapshots were created from the virtual machine, each with different endpoint protection products and one with none. This procedure ensured that the base system was exactly the same in all test systems. The following endpoint security suites, with the following configuration, were defined for this test:
    a. No additional protection, this snapshot was used to infect the OS and to verify the exploit replay (see 3.8 for details).
    b. Kaspersky Endpoint Security 10.2.1.23 (MR1) with default configuration
    c. Kaspersky Endpoint Security 10.2.1.23 (MR1) with AEP module only. This means that the following modules were turned off: File anti-virus, Web Anti-virus, Application Privilege Control, Application Startup Control and Web control.

        The System Watcher was configured as default, as seen on the following screenshot:



☑ **Enable System Watcher**

☑ Enable Exploit Prevention

☑ Log application activity for the BSS database

☑ Do not monitor the activity of applications that have a digital signature

Rollback of malware actions

☑ Roll back malware actions during disinfection

Proactive Defense

☑ Use behavior stream signatures (BSS)

On detecting malware activity:

Select action automatically

Figure 3 – KES configuration

d. Kaspersky Endpoint Security 10.2.2.10535 (SP1) with default configuration. This new version was released in the middle of the test, which means that only tests 151 – 300 were performed with this new version.

e. Kaspersky Endpoint Security 10.2.2.10535 (SP1) with AEP module only.

f. ESET Endpoint Security 5.0.2008.0, before December 29 (6% of the test). After, ESET Endpoint Security 6.1.2109.0 with exploit prevention

g. F-Secure Client Security 11.60 build 284

h. McAfee Endpoint Protection including VirusScan Enterprise + AntiSpyware Enterprise 8.8.0.1247, Siteadvisor 3.5.0.1121, and Host Intrusion Prevention 8.0, because these components are in the default suite.

i. Microsoft System Center Endpoint 4.3.220.0 and with Smartscreen enabled

j. Symantec Endpoint Protection 12.1.5 build 5337

k. Sophos Endpoint Security and Control 10.3

l. Trend Micro OfficeScan 11.0.1028

m. Trend Micro OfficeScan 11.0.1028 with Intrusion Defense Firewall 7.5.0.5914. See Appendix 1 for details on this configuration

The endpoint systems were installed with default configuration, potentially unwanted software removal was enabled, and if it was an option during install, cloud/community participation was enabled. The management servers were installed onto a different server. The purpose of management servers is to centrally administer, update and analyse logs in an enterprise environment. Installing the management server on a different server is highly recommended by vendors, so it does not interfere with the testing, machine resources are not used by the management server, etc.

6. Two sources of exploits were used during the test. Both sources provided us with exploits in real-time – with delays measured in hours. In spite of other "real world protection tests", no binary downloads (e.g. exe) were tested. ActiveX, VBscript based downloaders and Office macro documents were out of scope.

7. The virtual machine was reverted to a clean state and traffic was replayed by the proxy server. The replay meant that the browser was used as before, but instead of the original webservers, the proxy server answered the requests based on the recorded traffic. In this replay, no other traffic was allowed, which meant that unmatched requests (previously not recorded) were answered with HTTP 404 codes. When the "replayed exploit" was able to infect the OS, the exploit traffic was marked as a source for the tests. This method guarantees that exactly the same traffic will be seen by the endpoint protection systems, even if the original exploit kit goes down during the tests. Although this might be

axiomatic, it is important to note that no exploit traffic test case was deleted after this step of the test. All tests are included in the final results. In the case of HTTPS traffic, the original site was contacted, without replaying.

8. After new exploit traffic was approved, the endpoint protection systems were tested, in a random order. Before the exploit site was tested, it was verified that the endpoint protection had been updated to the latest version with the latest signatures and that every cloud connection was working. If there was a need to restart the system, it was restarted. In the proxy setup, unmatched requests were allowed to pass through and SSL/TLS was decrypted to ensure malware delivery and C&C. No VPN was used during the test. When user interaction was needed from the endpoint protection (e.g. site visit not recommended, etc.), the block/deny action was chosen. When user interaction was needed from Windows, we chose the run/allow options, except for UAC. No other processes were running on the system, except the Process Monitor from Sysinternals and Wireshark (both installed to non-default directories).
9. After navigating to the exploit site, the system was monitored to check for new processes, loaded DLLs or C&C traffic. For an analysis of the results, please see Section 3.8.
10. After an endpoint protection suite was tested, a new endpoint protection was randomly selected for the test until all endpoint protection products had been tested.
11. The process went back to step 7. until all 300 exploit site test cases were reached.

The following hardware was dedicated to the virtual machine:

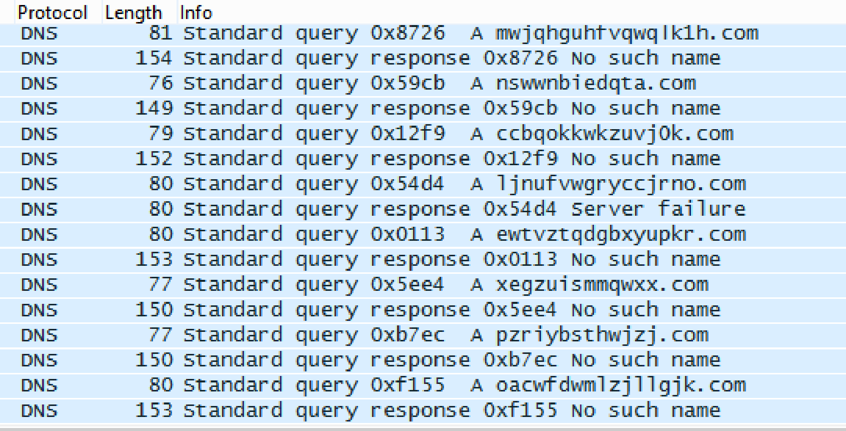
- 2 GB RAM memory
- 2 processors dedicated from AMD FX 8370E CPU
- 20 GByte free space
- 1 network interface
- SSD drive

3.1 Source of exploits

During our test, we used two independent sources of exploits:

1. Kafeine - http://malware.dontneedcoffee.com/
2. Ukatemi Technologies - a start-up from CrySyS Lab. - http://ukatemi.com/

Only exploit traffic which drops and immediately starts malware was included in the test. This was verified via Process Monitor, looking for Operation = Process Create or Operation = Load Library, either direct malware execution, or via regsrv32, cmd.exe, wscript.exe, java.exe, etc. In the case of in-memory malware (e.g. Bedep dropped via Angler EK), we checked the network dump for domain names associated with the malware.



```
Protocol Length Info
DNS          81 Standard query 0x8726  A mwjqhguhfvqwqlk1h.com
DNS         154 Standard query response 0x8726 No such name
DNS          76 Standard query 0x59cb  A nswwnbiedqta.com
DNS         149 Standard query response 0x59cb No such name
DNS          79 Standard query 0x12f9  A ccbqokkwkzuvj0k.com
DNS         152 Standard query response 0x12f9 No such name
DNS          80 Standard query 0x54d4  A ljnufvwgryccjrno.com
DNS          80 Standard query response 0x54d4 Server failure
DNS          80 Standard query 0x0113  A ewtvztqdgbxyupkr.com
DNS         153 Standard query response 0x0113 No such name
DNS          77 Standard query 0x5ee4  A xegzuismmqwxx.com
DNS         150 Standard query response 0x5ee4 No such name
DNS          77 Standard query 0xb7ec  A pzriybsthwjzj.com
DNS         150 Standard query response 0xb7ec No such name
DNS          80 Standard query 0xf155  A oacwfdwmlzjllgjk.com
DNS         153 Standard query response 0xf155 No such name
```

Figure 4 - Bedep trying to connect to C&C server

Both exploit sources provided us with exploits acquired privately, and not shared with any of the vendors. 60% of the exploit traffic was replayed within 8 hours of collection. The different endpoint protection systems were tested with a delay of not more than 6 hours and the endpoint protection systems were tested in a random order.

"300 different exploit traffics used" means that both the domain of the infected site and the domain of the exploit kit site were distinct from the other test cases.

In some cases, the replay from the original infected URL was not successful (for example the redirection URL's were dynamically changing) or the original infected site was not available. In these cases the replay was started from the landing page of the exploit site. The infected URL and redirection chain could not be seen by the endpoint protection system, so it had no chance to block the exploit at these early stages, but it still had many other opportunities to block the malware. As the test focused on exploit protection, and not URL blocking, we believed that this was acceptable.

While collecting the exploit traffic, the client was emulated via VPN so as to appear to reside in different countries of the world (UK, US, Germany, Russia, etc.), in order to simulate the global threat more accurately.

For the diversity of the exploits used, please see Section 3.9.

Details of the samples, including their URLs and code, were provided to partner vendors only after the sample was tested with all vendors.

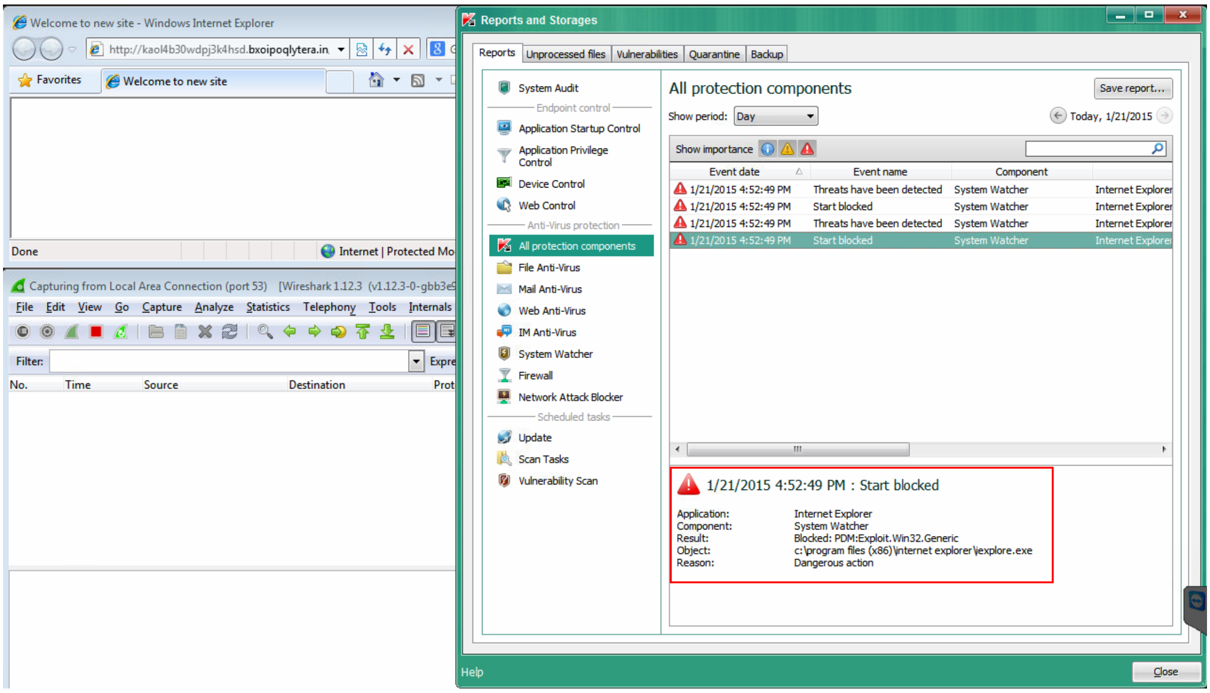
3.2 False positive test

No false positive test was carried out because there is no relevant false positive test which can truly measure the false positive ratio of such a complex scenario. Doing a false positive test on the URL block component cannot be considered as a valid false positive test because it only measures one component of the endpoint protection system. For example, if only the AEP module of Kaspersky is turned on, a false positive test cannot be carried out via visiting clean URLs, or by running legitimate applications.

3.3 0-day exploit test – CVE-2015-0311

On 2015 January 21, Kafeine spotted a previously unknown, true zero- day vulnerability exploited in-the-wild by the Angler exploit kit[1]. He quickly shared the sample with AV companies, and after sharing with them, he shared it with us. As we had Kaspersky Endpoint Protection with Automatic Exploit Prevention only configuration from a snapshot updated the day before, it was a perfect sample to test 0-day protection of KES. The screenshot below shows that the exploit was blocked by the exploit prevention module. This is a live example of KES proactive protection. As the sample had been shared with AV vendors, this test case was not included in the test.

[1] http://malware.dontneedcoffee.com/2015/01/unpatched-vulnerability-0day-in-flash.html



All protection components

Event date	Event name	Component	
1/21/2015 4:52:49 PM	Threats have been detected	System Watcher	Internet Explorer
1/21/2015 4:52:49 PM	Start blocked	System Watcher	Internet Explorer
1/21/2015 4:52:49 PM	Threats have been detected	System Watcher	Internet Explorer
1/21/2015 4:52:49 PM	Start blocked	System Watcher	Internet Explorer

1/21/2015 4:52:49 PM : Start blocked

Application:	Internet Explorer
Component:	System Watcher
Result:	Blocked: PDM:Exploit.Win32.Generic
Object:	c:\program files (x86)\internet explorer\iexplore.exe
Reason:	Dangerous action

3.4 In-memory malware

To our knowledge, this was the first in-the-wild exploit test using in-memory malware samples. The samples were exclusively Bedep samples, dropped by the Angler exploit kit. In-memory malware can bypass traditional AV protections, as there is no malware written to the hard disk, thus the malware is not checked at all. This threat can still be blocked before exploitation by URL block, analysing the JavaScript/HTML/SWF files, or even by blocking the exploit itself. Because the malware is encrypted on the network level (e.g. using XTEA), it is not possible to detect the malware delivery by traditional methods. Also, the threat can be blocked after infection, when it starts to drop more malware onto the victim OS – which are traditional, persistent malware. Also, because Angler exploit kit uses the so-called domain shadowing technology, replaying the exploit even 4 hours after acquire can result in that the malware cannot contact the C&C server at all, as these servers are down. This fast-flux technology makes URL blocking highly ineffective.

3.5 Alternative browsers used

Some of the test cases were compatible with Firefox (Flash vulnerability) or Chrome (Silverlight vulnerability). Although it was not easy to crawl these exploit samples, it is important to note that in an enterprise environment, not 100% of the users use the company's official browser (typically Internet Explorer), but other browsers like Firefox or Chrome. Although these browsers have not previously been targeted by common exploit kits, this is not the case today. It is important to patch these browsers and the plugins inside ASAP, as criminals usually target them in 1-2 weeks after the patch becomes available. And although these are not 0day attacks, they are still very effective as many users do not rush to update their Flash player in both Internet Explorer and Firefox (as they are different). The Silverlight Chrome exploit is not a 100% drive-by-exploit, as one has to enable the outdated plugin in Chrome in order to trigger the exploit.

3.6 Exploit kit detecting AV software

Some exploit kits (e.g. the Angler, or Nuclear exploit kit) use the CVE-2013-7331 vulnerability to detect the presence of AV software or virtualization before trying to exploit vulnerabilities in the browser or associated plugins. Exploit kits can also detect tools related to virtualization (e.g. VMWare tools) or tools related to exploit analysis (e.g. Fiddler, Wireshark). To defeat these problems, we used the following strategy:

- We did not install VMWare tools, Fiddler on the victim
- All other tools (e.g. Wireshark) were installed to non-default directories.

As the installed tools had the same install directories on all machines, we were able to test the exploit on a non-protected computer first. Given that none of the exploit kit samples detected our machine, we were able to test all AV with these exploits. As exploit kits were actively evading some AV products (e.g. Kaspersky, Symantec, Trend Micro), whenever this happened, we marked the results as "exploit blocked". Had the exploit kits not evaded these configurations, the results would have changed. This configuration ensured that the test was the same as in the real world. "Real world tests" not aware of this Internet Explorer vulnerability can miss important exploits in the tests.

For more information on this trick, please refer to http://malware.dontneedcoffee.com/2014/10/cve-2013-7331-and-exploit-kits.html

3.7 The advantage of manual test vs automated test

Although testing of all 300 samples against 12 configurations manually is a huge task, we believe the effort was worth it. The problem was that in some cases, parts of the tests could not be automated easily. For example, the exploit replay did not work out of the box, malware used a new method to start, in memory malware in the samples, etc. As expected, these cases, which needed manual testing, were the cases where most AV engines fail – because they rely too much on automated tests or automated detection of new samples.

3.8 Analysis of the exploit results

The testing was carried out between December 18, 2014 and March 2, 2015.

We defined the following stages, where the exploit can be prevented by the endpoint protection system:

1. Blocking the URL (infected URL, exploit kit URL, redirection URL, malware URL) by the URL database (local or cloud). For example, a typical result is the browser displaying a "site has been blocked" message by the endpoint protection.
2. Analysing and blocking the page containing a malicious HTML code, Javascripts (redirects, iframes, obfuscated Javascripts, etc.), or Flash files.
3. Blocking the downloaded payload by analysing the malware before it is started. For example, the malware payload download (either the clear-text binary or the encrypted/encoded binary) can be seen in the proxy traffic, but no malware process starts. We call this "AV signature blocked" later, but a reputation based block or heuristic based block is also included in this category.
4. Blocking the exploit before the shellcode or malware can be executed.
5. There was a successful start by the dropped malware.
6. There was a successful start by the dropped malware, but after some time, all dropped malware was terminated and deleted ("malware starts, but blocked later").

The first four exploit prevention stages were counted together to simplify the results. At this stage, no malicious processes had run on the victim computer. This was expected behaviour of an endpoint protection system; the attacker had no chance to execute any untrusted code on the victim.

If the endpoint protection system did not block the exploit, but let the payload download and run malware, it was a complete fail of the product. In some of the cases, the endpoint protection systems were able to detect some or all parts of the malware, but this was not marked as "system protected" for the following reasons:

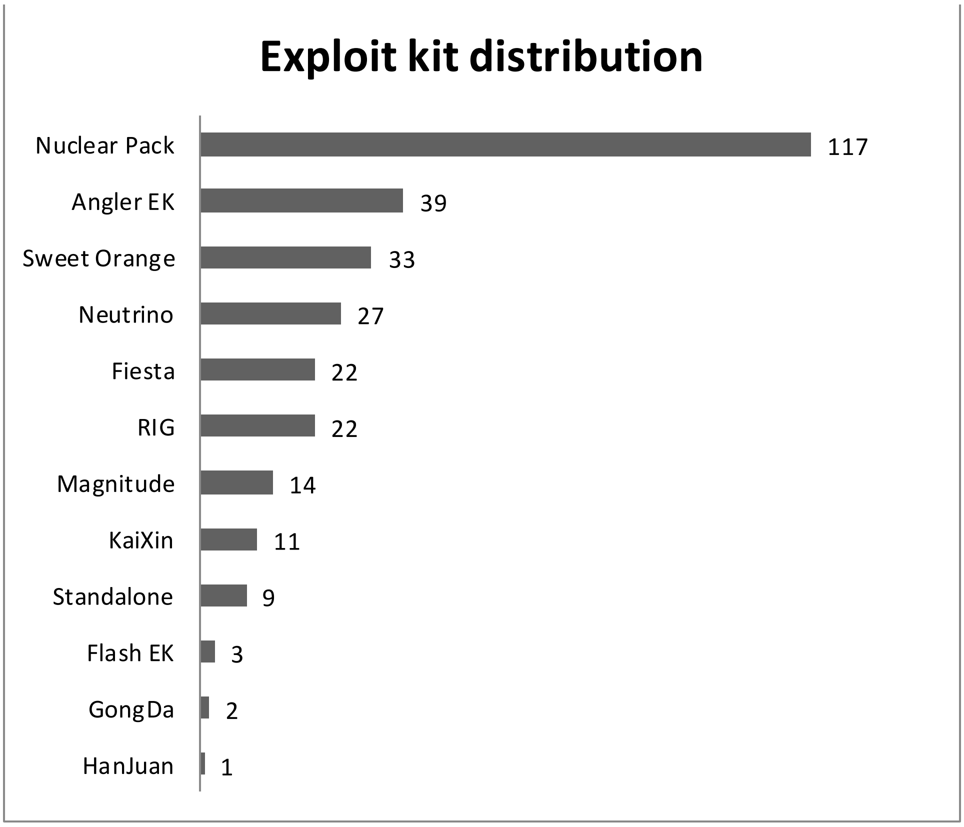
- The scope of the test was exploit prevention and not the detection of malware running on the system.
- It is not possible to determine what kind of commands have been executed or what information exfiltrated by the malware. Data exfiltration cannot be undone or remediated.
- It cannot be determined if the malware exited because the endpoint protection system blocked it, or if malware quit because it detected monitor processes (procmon.exe), virtualization, or quit because it did not find its target environment.

- Checking for malware remediation can be too time-consuming and remediation scoring very difficult. For example, we experienced several alerts that the endpoint protection system blocked a URL/page/exploit/malware, but still the malware was able to execute and run on the system. On other occasions, the malware code was deleted from the disk by the endpoint protection system, but the malware process was still running, or some parts of the malware were detected and killed, while others were not.
- Sometimes the products blocked some or all parts of the malware from running, but failed to notify/alert the user or administrator about the incident.

We believe that such zero-tolerance scoring helps enterprises to choose the best products, using simple metrics. Manually verifying the successful remediation of the malware in an enterprise environment is a very resource-intensive process and costs a lot of money. In our view, malware needs to be blocked before it has a chance to run.

3.9 Analysis of the exploit kits used in the test

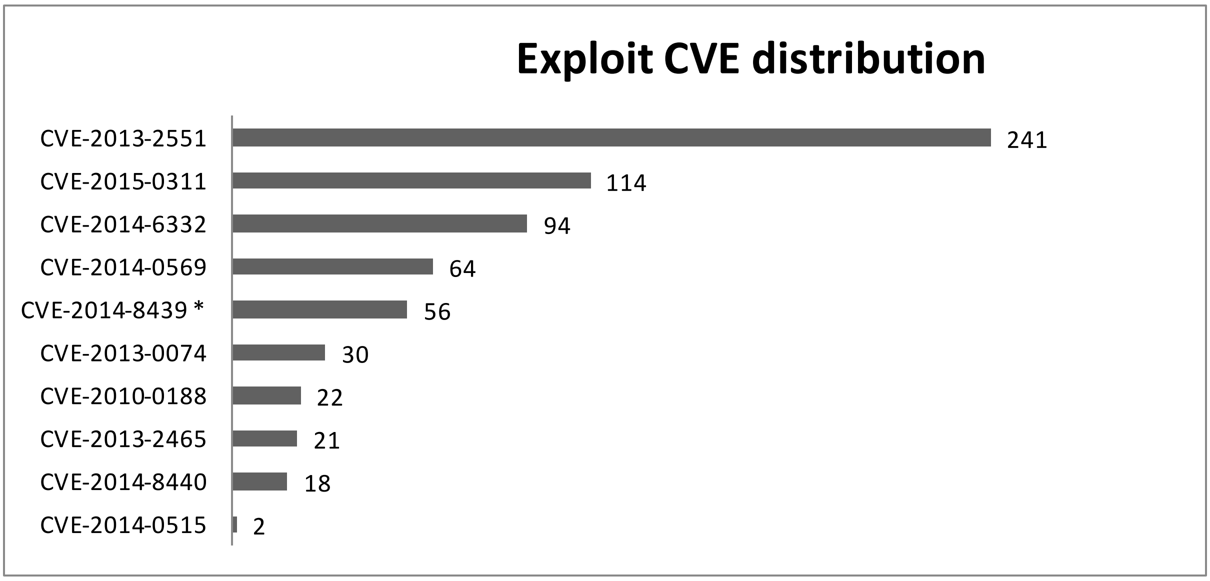
The following graph displays the distribution of the detected exploit kits. The distribution is the result of choosing random malicious sites at the time of testing. The exploit kit names have been determined via Emerging Threats Pro IDS alerts and manual analysis.



Graph 1 - Distribution of exploit kits

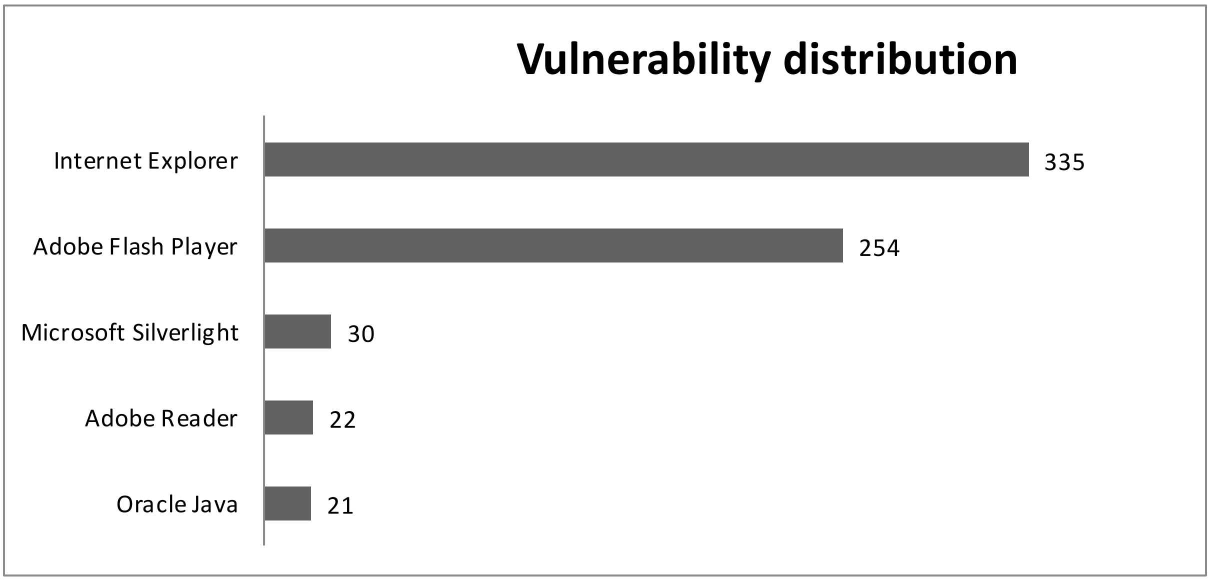
3.10 Analysis of the exploits used in the test

The following graph displays the distribution of the detected exploits. Some exploit kits delivered more than one exploit.



Graph 2 - Distribution of exploit CVEs

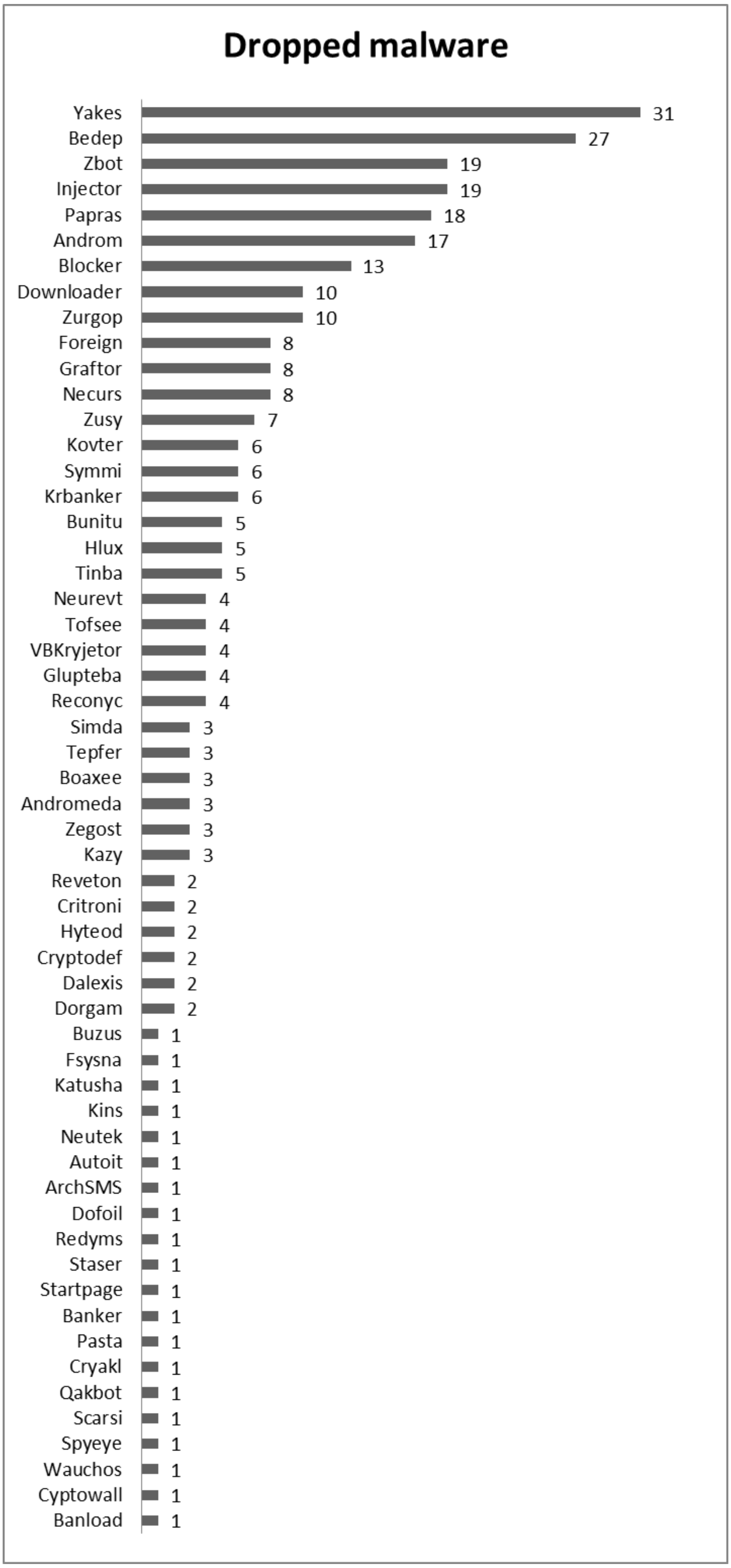
The following graph displays the distribution of the exploited software.



Graph 3 - Distribution of exploited software

3.11 Analysis of the dropped malware

The following graph displays the distribution of the detected malware family name. When it was not a generic but a specific name, the Kaspersky malware family name was used. When it was a generic name or not detected by Kaspersky, we looked at other AV detected family names. When other AV vendors had Generic as well, generic was used. We were not able to recover some malware (this was not included in the statistics).



Graph 4 - Distribution of dropped malware

We would like to highlight some of the top malware in the test:

Yakes is a typical downloader, which can download other malware to the system. It is also known for modifying the hosts file.

Victims infected with Bedep become part of a botnet, and can be used for tasks like DDOS, click fraud or spamming. Bedep can infect the system by either dropping malware to the disk, or executing itself in memory.

ZBot (usually) refers to the financial stealer malware family based on Zeus. Zeus can steal passwords from the browser, inject web forms for additional data stealing, steal private certificates, etc.

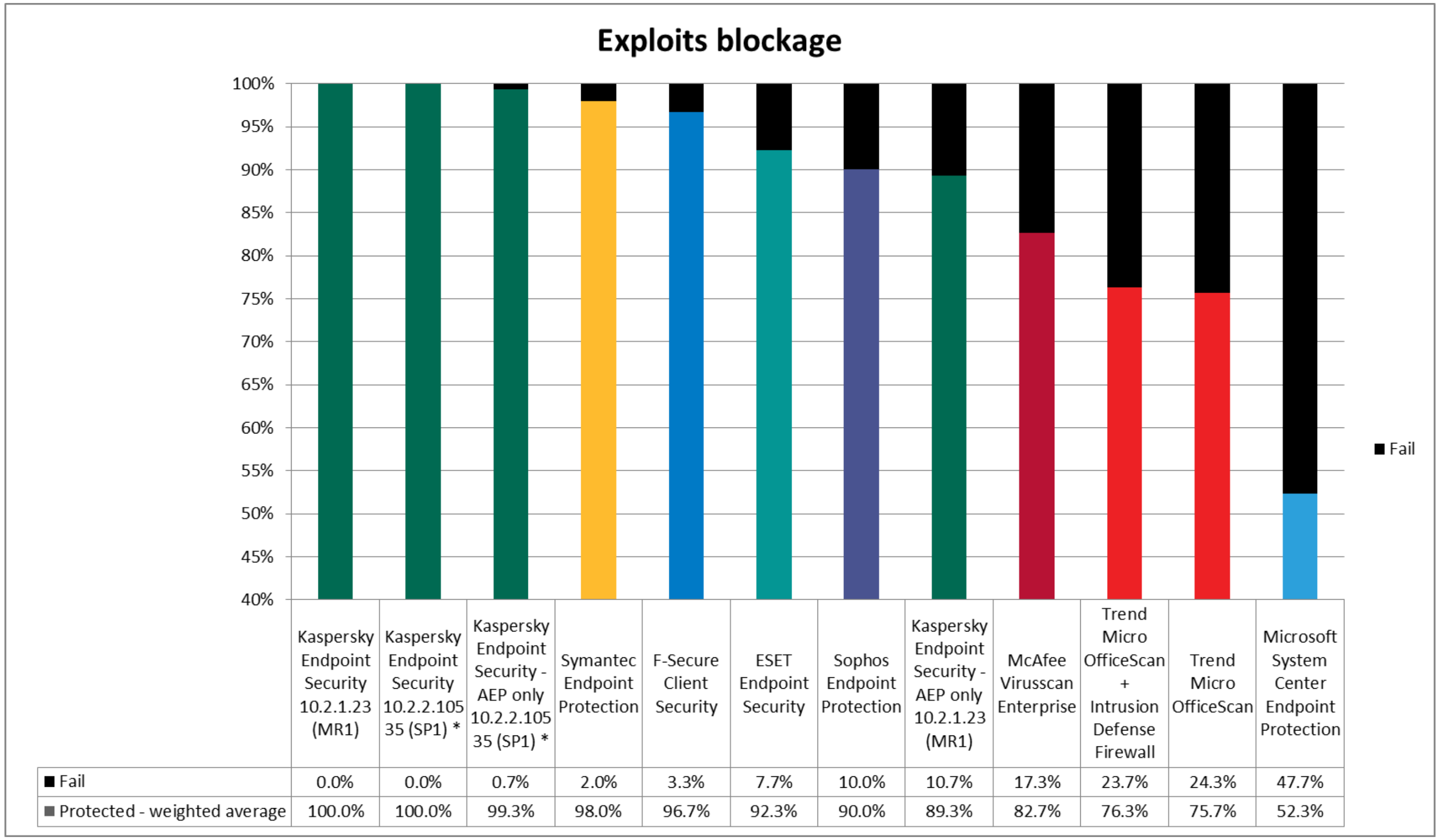
All malware families in the test are in-the-wild malware, and most can be considered as high-risk when they infect a workstation.

4 Final results

This section is where the threat was blocked either at the URL, HTML/Javascript the exploit or the payload delivery/start stage.

4.1 URL blocked or HTML/JS/SWF blocked or exploit blocked or "AV signature" blocked

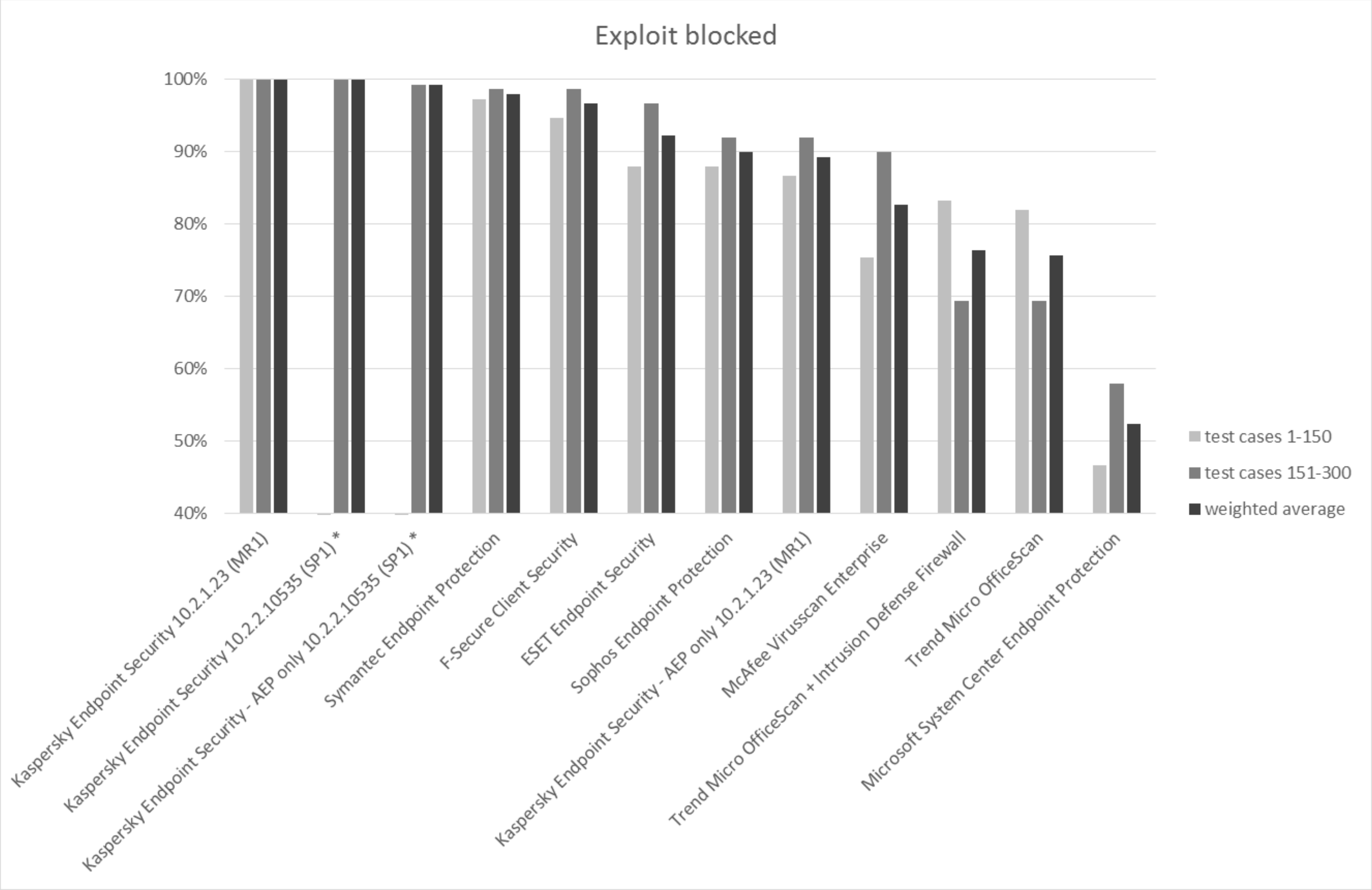
In this set of results, all 300 exploit traffics are included. KES SP1 is included in the graph since the weighted average along all test cases is calculated and shown. KES SP1 was released in the middle of the test, thus only test-cases 151-300 were tested with this new release.



	Kaspersky Endpoint Security 10.2.1.23 (MR1)	Kaspersky Endpoint Security 10.2.2.10535 (SP1) *	Kaspersky Endpoint Security - AEP only 10.2.2.10535 (SP1) *	Symantec Endpoint Protection	F-Secure Client Security	ESET Endpoint Security	Sophos Endpoint Protection	Kaspersky Endpoint Security - AEP only 10.2.1.23 (MR1)	McAfee Virusscan Enterprise	Trend Micro OfficeScan + Intrusion Defense Firewall	Trend Micro OfficeScan	Microsoft System Center Endpoint Protection
■ Fail	0.0%	0.0%	0.7%	2.0%	3.3%	7.7%	10.0%	10.7%	17.3%	23.7%	24.3%	47.7%
■ Protected - weighted average	100.0%	100.0%	99.3%	98.0%	96.7%	92.3%	90.0%	89.3%	82.7%	76.3%	75.7%	52.3%

Graph 5 - Distribution of exploit and malware blocking capability

The following detailed graph shows the results of samples 1-150 and 151-300, as well as the weighted average.



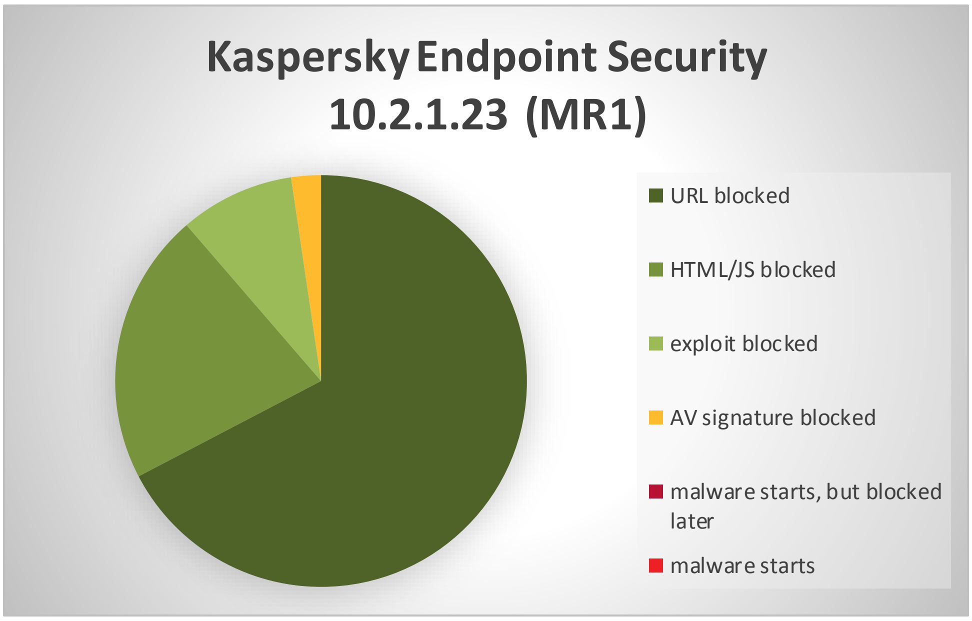
Product / test-cases	1-150	151-300	weighted average
Kaspersky Endpoint Security 10.2.1.23 (MR1)	100.0%	100.0%	100.0%
Kaspersky Endpoint Security 10.2.2.10535 (SP1)	n.a.	100.0%	100.0%
Kaspersky Endpoint Security - AEP only 10.2.2.10535 (SP1)	n.a.	99.3%	99.3%
Symantec Endpoint Protection	97.3%	98.7%	98.0%
F-Secure Client Security	94.7%	98.7%	96.7%
ESET Endpoint Security	88.0%	96.7%	92.3%
Sophos Endpoint Protection	88.0%	92.0%	90.0%
Kaspersky Endpoint Security - AEP only 10.2.1.23 (MR1)	86.7%	92.0%	89.3%
McAfee Virusscan Enterprise	75.3%	90.0%	82.7%
Trend Micro OfficeScan + Intrusion Defense Firewall	83.3%	69.3%	76.3%
Trend Micro OfficeScan	82.0%	69.3%	75.7%
Microsoft System Center Endpoint Protection	46.7%	58.0%	52.3%

The results lead to the conclusion that only one endpoint protection systems stand out from the crowd with it’s excellent protection:

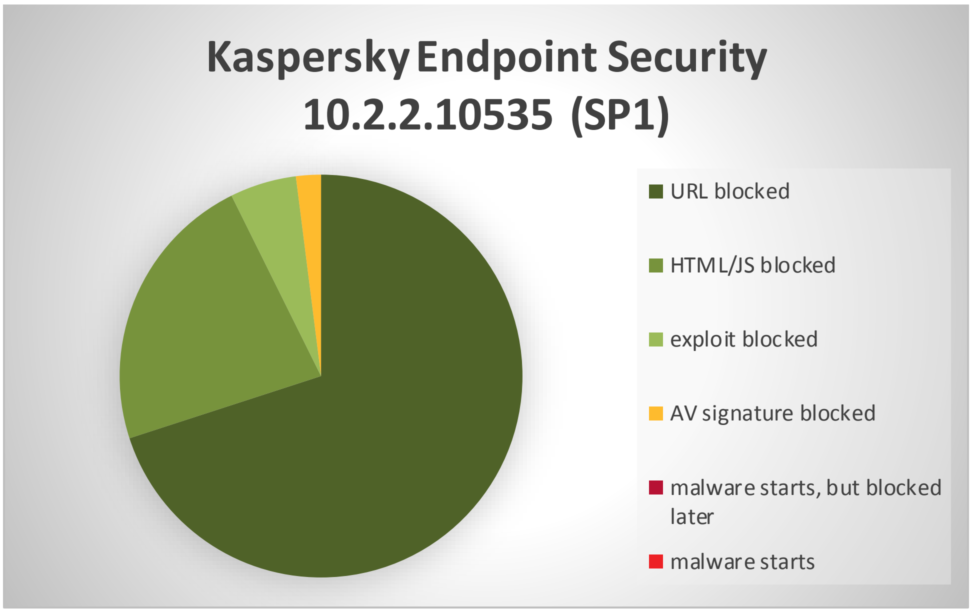
- Kaspersky Endpoint Security 10

4.2 Layers of endpoint protection

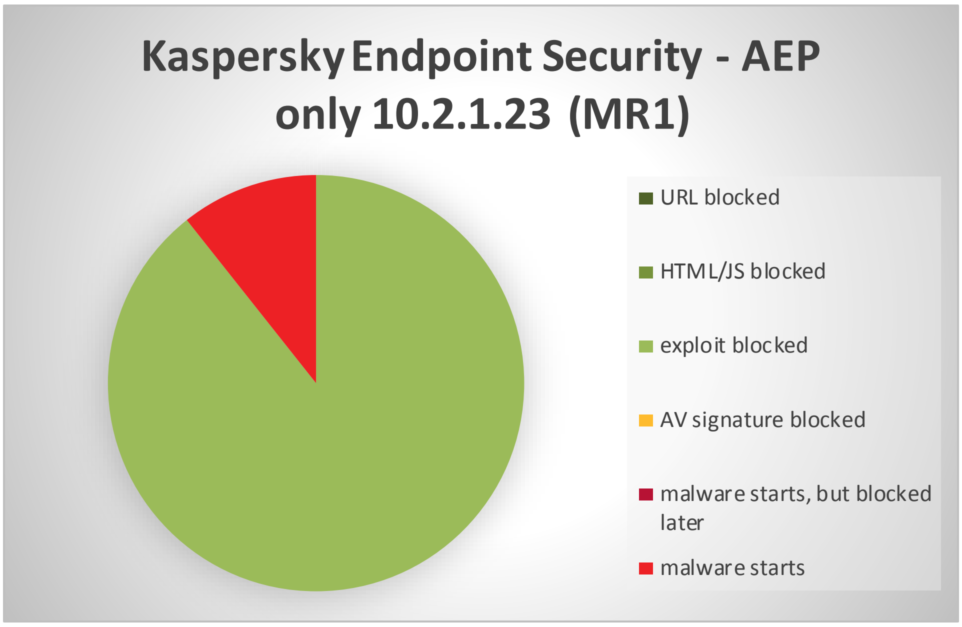
For data labels see paragraph 4.1. It is important to note that, in our opinion, the sooner an exploit is blocked in the exploit chain, the better. Some protection layers are more reactive (e.g. URL blocking, AV signature blocking) and others are more proactive (analysing of HTML files, exploit protection, behaviour analysis). Sometimes it is hard to draw conclusions from these test results, because exploit protection can work before the malware is downloaded (e.g. malicious shell-code never runs), or sometimes exploit protection blocks the execution of malware after it has been downloaded by the malicious shellcode. Whenever malware had a chance to start, but was blocked later (e.g. by behaviour analysis), we marked this as a fail ("malware starts, but blocked later" category), given that some malicious action could have already been taken by the malware. Also, when AV signature detection blocked malware execution, it meant that the malicious shellcode had already been able to run on the victim system – which is usually "download and execute malware" only, but it is possible (but not prevalent) that the shellcode serves other function.

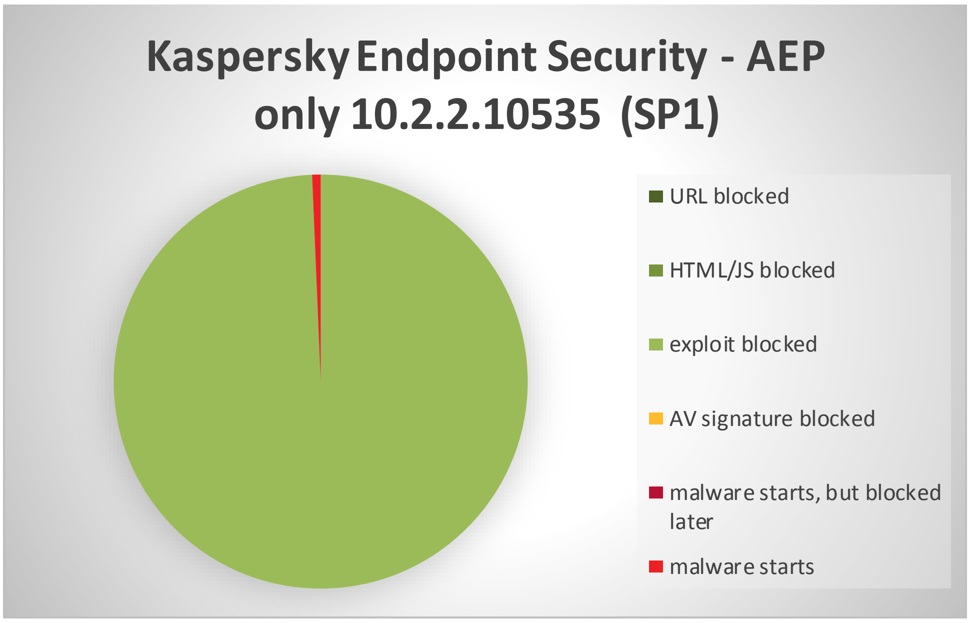


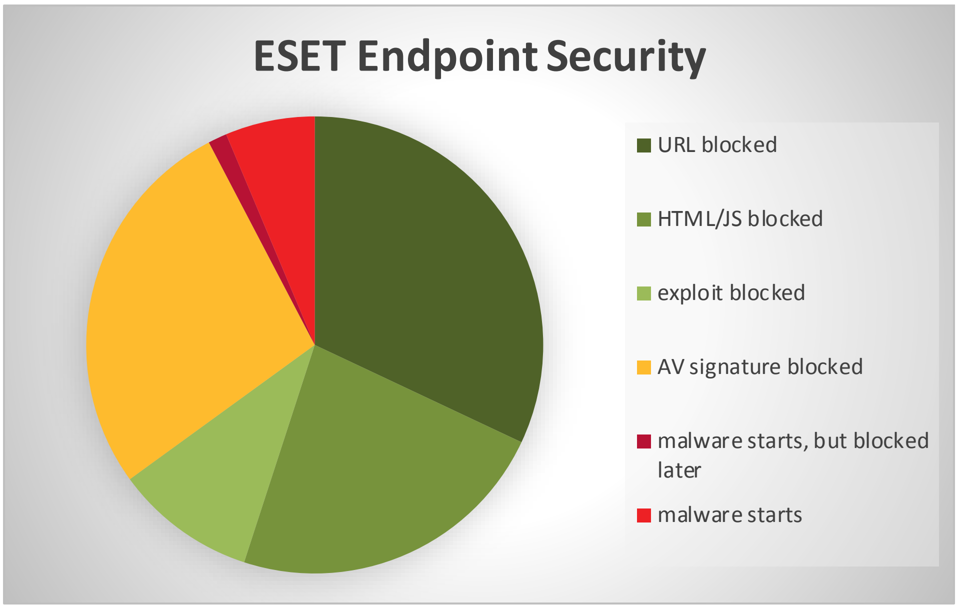
Kaspersky Endpoint Security 10.2.1.23 (MR1)

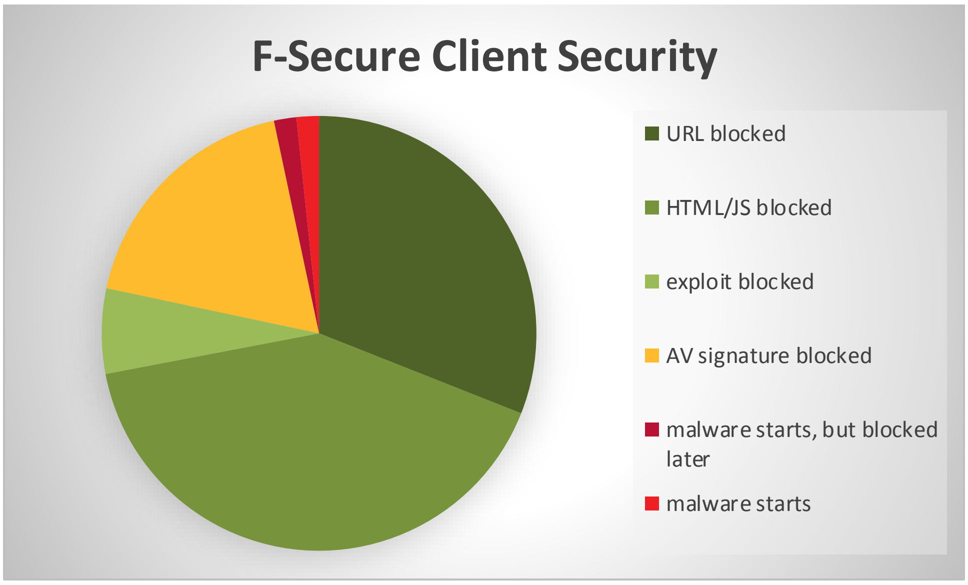


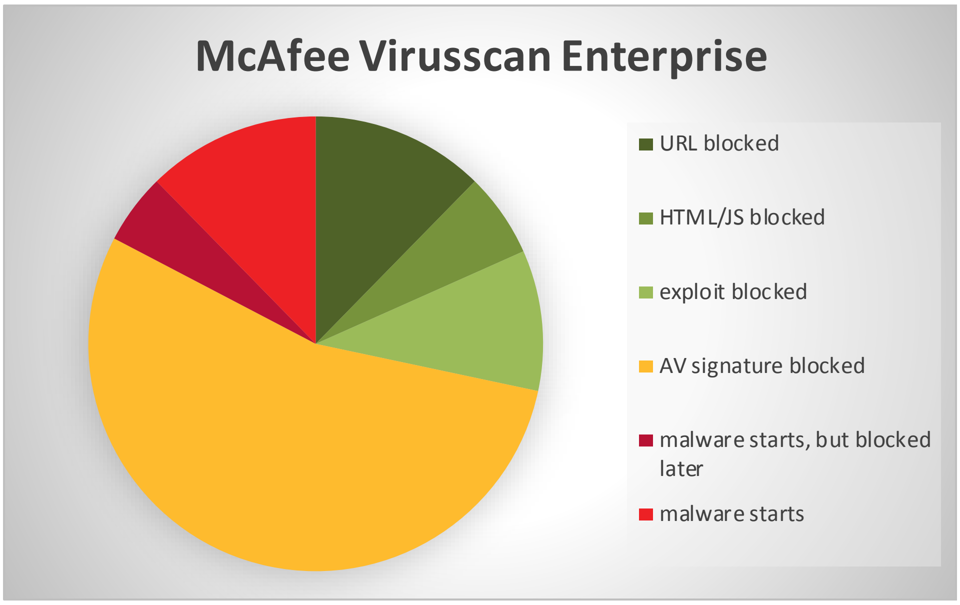
Kaspersky Endpoint Security 10.2.2.10535 (SP1)

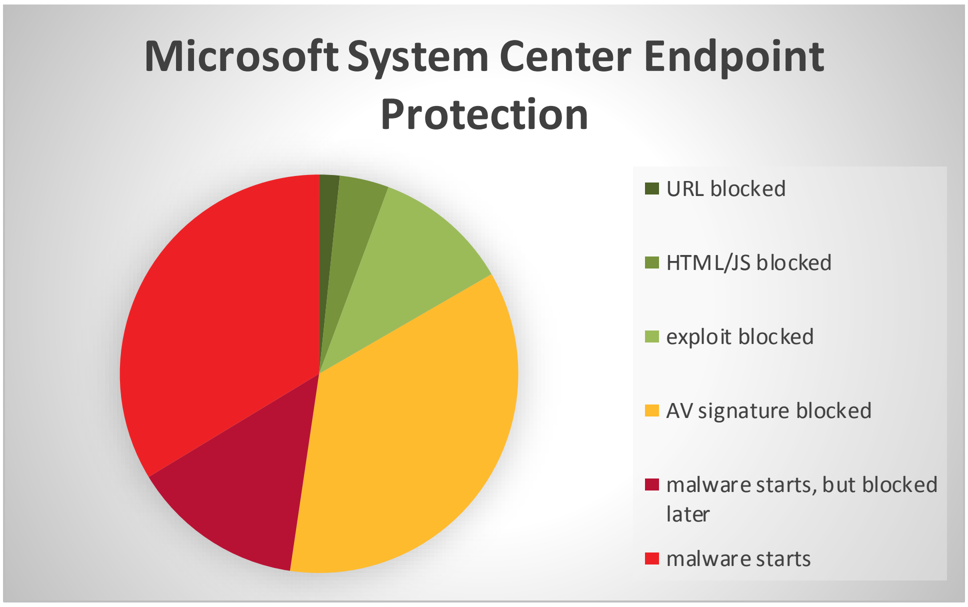

Kaspersky Endpoint Security - AEP only 10.2.1.23 (MR1)
URL blocked
HTML/JS blocked
exploit blocked
AV signature blocked
malware starts, but blocked later
malware starts

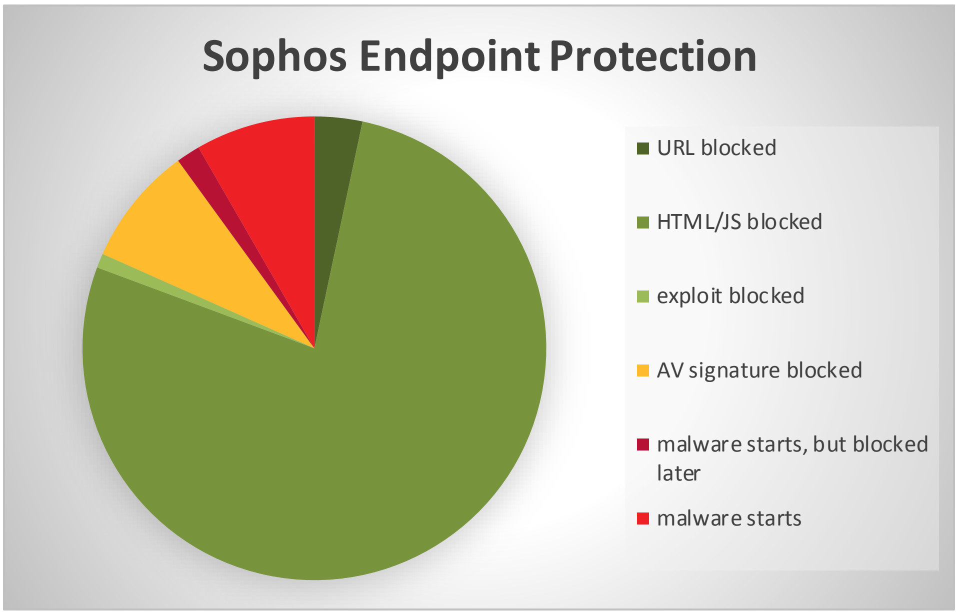

Kaspersky Endpoint Security - AEP only 10.2.2.10535 (SP1)
URL blocked
HTML/JS blocked
exploit blocked
AV signature blocked
malware starts, but blocked later
malware starts

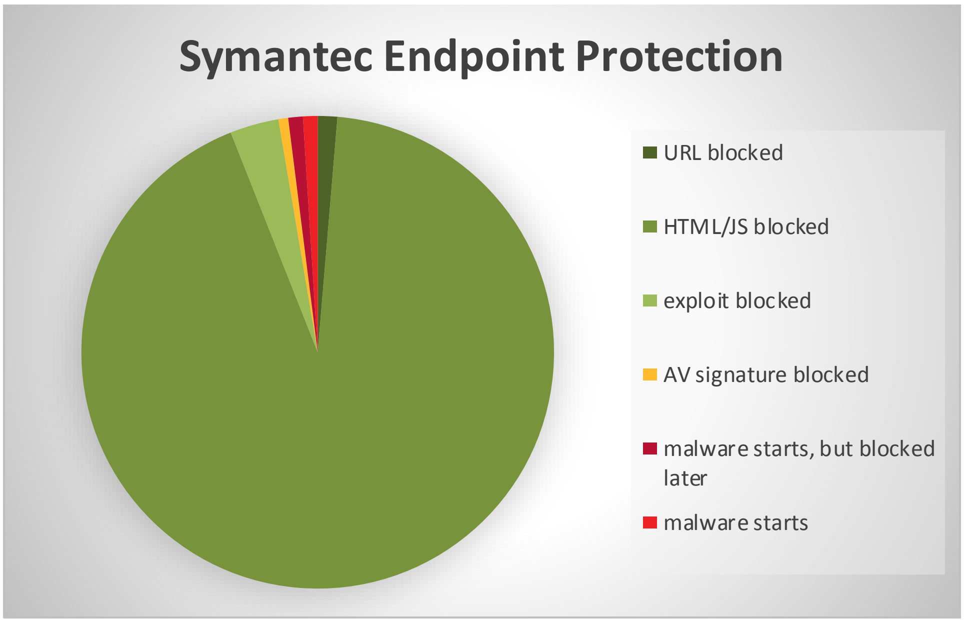

ESET Endpoint Security
URL blocked
HTML/JS blocked
exploit blocked
AV signature blocked
malware starts, but blocked later
malware starts

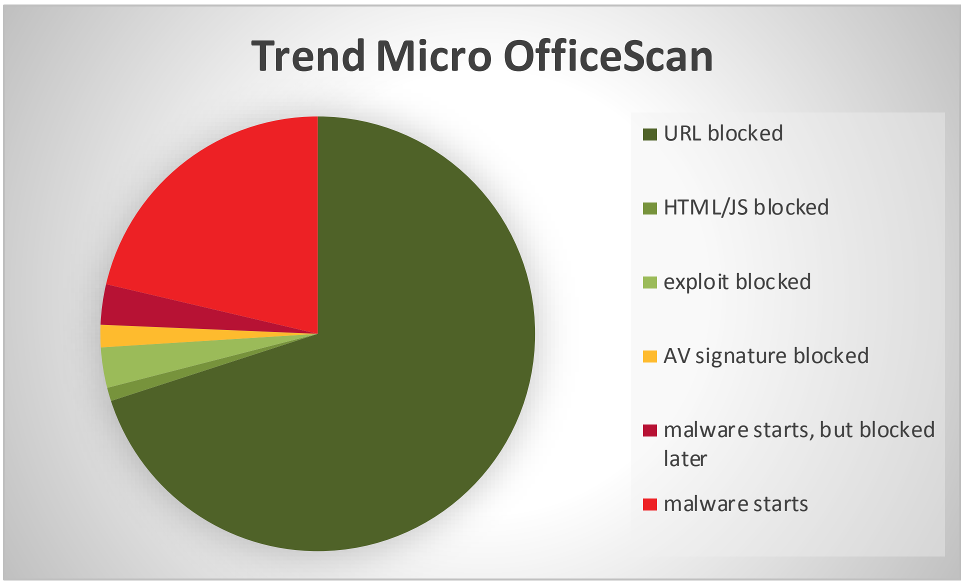

F-Secure Client Security
URL blocked
HTML/JS blocked
exploit blocked
AV signature blocked
malware starts, but blocked later
malware starts

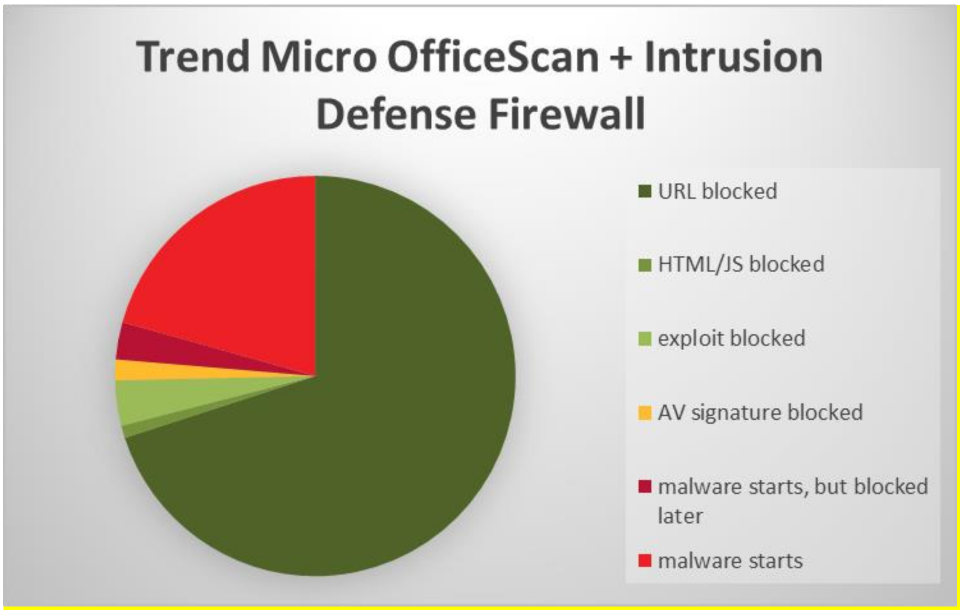

McAfee Virusscan Enterprise
URL blocked
HTML/JS blocked
exploit blocked
AV signature blocked
malware starts, but blocked later
malware starts


Microsoft System Center Endpoint Protection
URL blocked
HTML/JS blocked
exploit blocked
AV signature blocked
malware starts, but blocked later
malware starts


Sophos Endpoint Protection
URL blocked
HTML/JS blocked
exploit blocked
AV signature blocked
malware starts, but blocked later
malware starts


Symantec Endpoint Protection
URL blocked
HTML/JS blocked
exploit blocked
AV signature blocked
malware starts, but blocked later
malware starts





4.3 Analysis of the final results

Some AV products are better in one layer of protection, others in other layers. When the results were analysed, something became obvious in the data. When an AV is only good at URL protection or traditional AV protection (signatures), it is easily bypassed. When an AV can detect exploit kits very well, it can fail on standalone exploits. Only products having very good protection at every layer can achieve exceptional results. Products without exploit protection are doomed to fail.

In a home environment, a 2% difference in the results is negligible, but in the case of enterprises with ten-thousand computers, small differences add up and that matters.

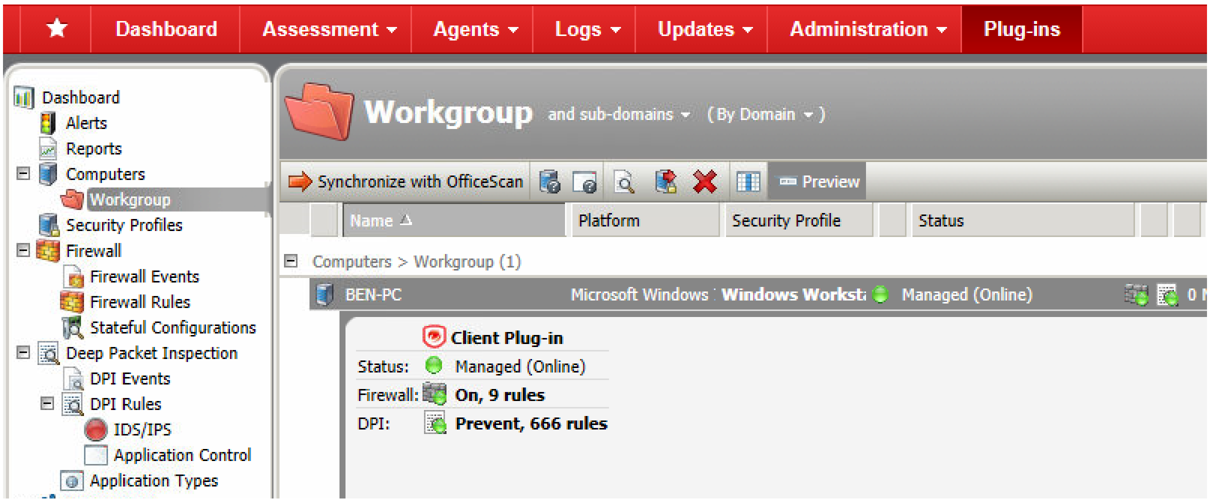
If you wish to contact us regarding this test, please use the form on our web page:

http://www.mrg-effitas.com/contact/

5 Appendix

5.1 Trend Micro OfficeScan Intrusion Defense Firewall issues

When we started the tests in December 2014, we believed we had configured the Intrusion Defense Firewall module properly on desktop clients. When we completed the tests at the end of February, we were shocked to see that the Intrusion Defense Firewall (IDF) had blocked 0 attacks. We immediately started to investigate the issue, and found the root cause. Although the IDF module was updated on the client, it was running, after scanning the client for recommendations (unpatched vulnerabilities), and enabling the recommended rules on the client, the IDF rules were not effective because the proxy in our test was not running on port 8080. After manually updating the rules to be effective on our proxy configuration, we checked the effectiveness of the rules with both manual rules (blocking "Hello IDF!") and with basic, non-obfuscated Metasploit exploits. As this new configuration successfully blocked both the manual and Metasploit exploits, we retested the test cases where the default Trend Micro OfficeScan without the IDF module had failed. Although we were still surprised by the low detection rate of the IDF module, we are sure it was configured properly at the time. It is important to note that many enterprises use the webproxy on ports other than 8080 (e.g. Squid on 3128), and these companies' web traffic with default policies are not protected by the IDF module at all. The problem is not mentioned in the official deployment guide.



Name	Platform	Security Profile	Status
BEN-PC	Microsoft Windows	Windows Workst	Managed (Online)

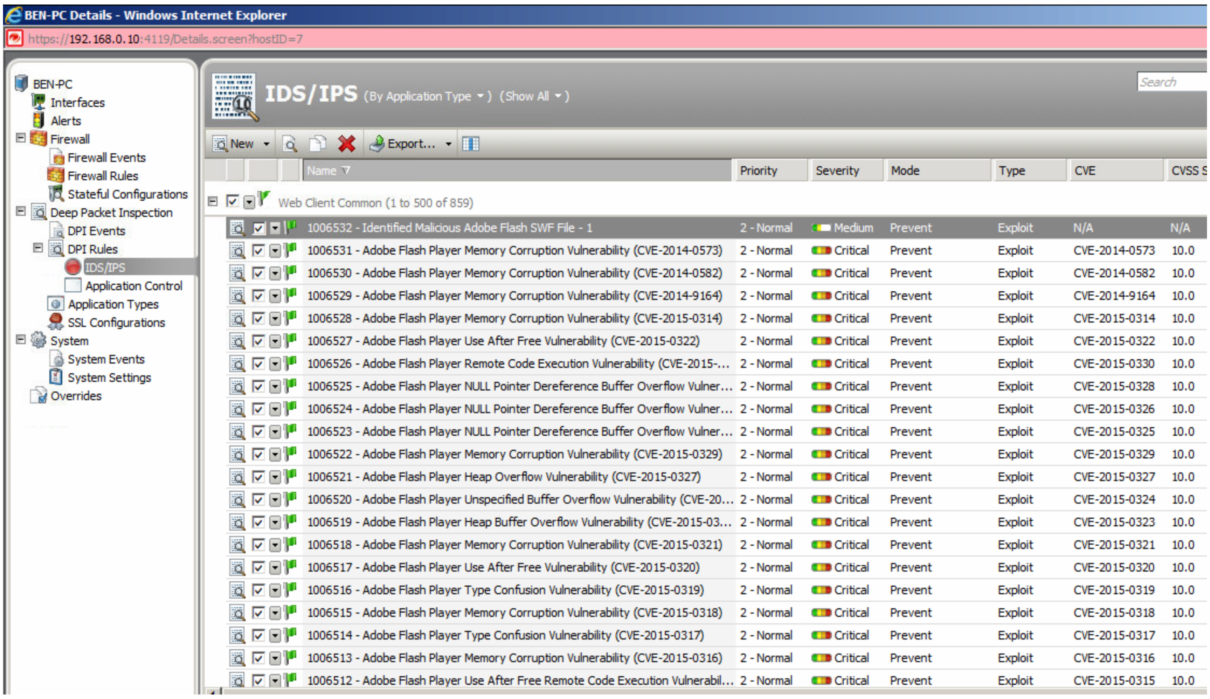
Client Plug-in

Status: Managed (Online)

Firewall: On, 9 rules

DPI: Prevent, 666 rules

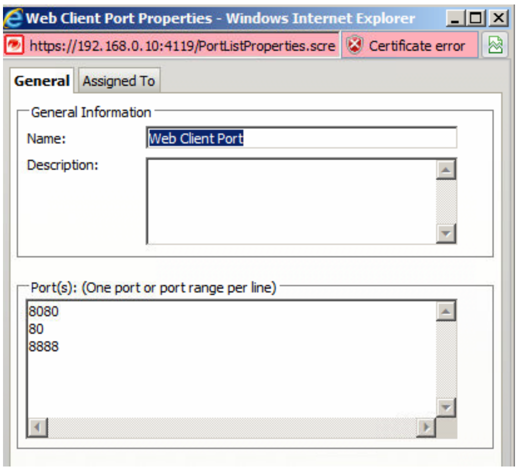
Figure 5 - DPI rules enabled on the client



Web Client Common (1 to 500 of 859)

Name	Priority	Severity	Mode	Type	CVE	CVSS S
1006532 - Identified Malicious Adobe Flash SWF File - 1	2 - Normal	Medium	Prevent	Exploit	N/A	N/A
1006531 - Adobe Flash Player Memory Corruption Vulnerability (CVE-2014-0573)	2 - Normal	Critical	Prevent	Exploit	CVE-2014-0573	10.0
1006530 - Adobe Flash Player Memory Corruption Vulnerability (CVE-2014-0582)	2 - Normal	Critical	Prevent	Exploit	CVE-2014-0582	10.0
1006529 - Adobe Flash Player Memory Corruption Vulnerability (CVE-2014-9164)	2 - Normal	Critical	Prevent	Exploit	CVE-2014-9164	10.0
1006528 - Adobe Flash Player Memory Corruption Vulnerability (CVE-2015-0314)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0314	10.0
1006527 - Adobe Flash Player Use After Free Vulnerability (CVE-2015-0322)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0322	10.0
1006526 - Adobe Flash Player Remote Code Execution Vulnerability (CVE-2015-...	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0330	10.0
1006525 - Adobe Flash Player NULL Pointer Dereference Buffer Overflow Vulner...	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0328	10.0
1006524 - Adobe Flash Player NULL Pointer Dereference Buffer Overflow Vulner...	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0326	10.0
1006523 - Adobe Flash Player NULL Pointer Dereference Buffer Overflow Vulner...	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0325	10.0
1006522 - Adobe Flash Player Memory Corruption Vulnerability (CVE-2015-0329)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0329	10.0
1006521 - Adobe Flash Player Heap Overflow Vulnerability (CVE-2015-0327)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0327	10.0
1006520 - Adobe Flash Player Unspecified Buffer Overflow Vulnerability (CVE-20...	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0324	10.0
1006519 - Adobe Flash Player Heap Buffer Overflow Vulnerability (CVE-2015-03...	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0323	10.0
1006518 - Adobe Flash Player Memory Corruption Vulnerability (CVE-2015-0321)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0321	10.0
1006517 - Adobe Flash Player Use After Free Vulnerability (CVE-2015-0320)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0320	10.0
1006516 - Adobe Flash Player Type Confusion Vulnerability (CVE-2015-0319)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0319	10.0
1006515 - Adobe Flash Player Memory Corruption Vulnerability (CVE-2015-0318)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0318	10.0
1006514 - Adobe Flash Player Type Confusion Vulnerability (CVE-2015-0317)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0317	10.0
1006513 - Adobe Flash Player Memory Corruption Vulnerability (CVE-2015-0316)	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0316	10.0
1006512 - Adobe Flash Player Use After Free Remote Code Execution Vulnerabil...	2 - Normal	Critical	Prevent	Exploit	CVE-2015-0315	10.0

Figure 6 - Recent IDF rules enabled


Web Client Port Properties - Windows Internet Explorer
https://192.168.0.10:4119/PortListProperties.scre
Certificate error
General
Assigned To
General Information

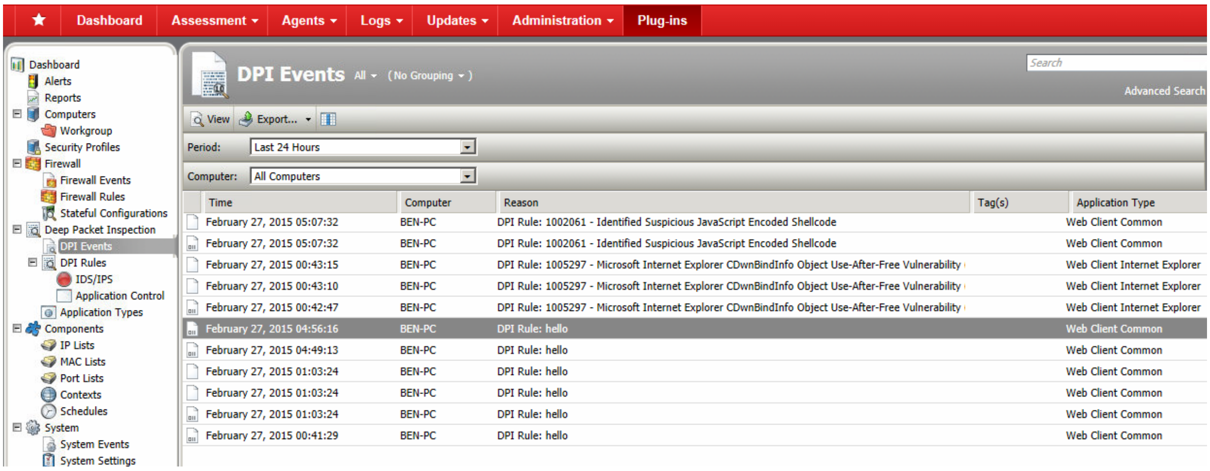
Name: Web Client Port

Description:

Port(s): (One port or port range per line)

```
8080
80
8888
```

Figure 7 - Proxy on port 8888 configured



Period: Last 24 Hours

Computer: All Computers

Time	Computer	Reason	Tag(s)	Application Type
February 27, 2015 05:07:32	BEN-PC	DPI Rule: 1002061 - Identified Suspicious JavaScript Encoded Shellcode		Web Client Common
February 27, 2015 05:07:32	BEN-PC	DPI Rule: 1002061 - Identified Suspicious JavaScript Encoded Shellcode		Web Client Common
February 27, 2015 00:43:15	BEN-PC	DPI Rule: 1005297 - Microsoft Internet Explorer CDwnBindInfo Object Use-After-Free Vulnerability		Web Client Internet Explorer
February 27, 2015 00:43:10	BEN-PC	DPI Rule: 1005297 - Microsoft Internet Explorer CDwnBindInfo Object Use-After-Free Vulnerability		Web Client Internet Explorer
February 27, 2015 00:42:47	BEN-PC	DPI Rule: 1005297 - Microsoft Internet Explorer CDwnBindInfo Object Use-After-Free Vulnerability		Web Client Internet Explorer
February 27, 2015 04:56:16	BEN-PC	DPI Rule: hello		Web Client Common
February 27, 2015 04:49:13	BEN-PC	DPI Rule: hello		Web Client Common
February 27, 2015 01:03:24	BEN-PC	DPI Rule: hello		Web Client Common
February 27, 2015 01:03:24	BEN-PC	DPI Rule: hello		Web Client Common
February 27, 2015 01:03:24	BEN-PC	DPI Rule: hello		Web Client Common
February 27, 2015 00:41:29	BEN-PC	DPI Rule: hello		Web Client Common

Figure 8 - Sample and Metasploit exploits blocked