360 Degree Assessment & Certification

Q4 2020

MRG Effitas
EFFICACY ASSESSMENT & ASSURANCE

MRG Effitas Ltd.
Efficacy Assessment & Assurance

MRG Effitas

Contents

MRG Effitas is a world-leading, independent IT security efficacy testing & assurance company. We are trusted by antimalware vendors across the world.

TEL:
+44 (0)20 3239 9289

EMAIL:
contact@mrg-effitas.com

TWITTER:
@mrgeffitas

Introduction

MRG Effitas is a world leader independent IT research company having a core focus on AV efficacy assessments both in the traditional “Real World” malware detection capabilities and in the financial fraud prevention area.

The methodology employed in this test maps closely to Real World practice representing the valid threads endangering anyone using Windows operating system. This evaluation is aimed to help users choosing the most suitable security application.

This Programme is called “360 Assessment & Certification” since it tests the security applications capabilities with a full spectrum of attack vectors. In the 360 Assessment, trojans, backdoors, spyware, financial malware, ransomware and “other” malicious applications are all used. Alongside the traditional In-The-Wild (ITW) file-based attacks, our evaluation also contains scenarios where fileless cases and exploitation techniques, live botnets and financial malware simulators are also applied.

Besides the malicious attacks, in order to evaluate the practical accuracy of AV products, they were exposed to potentially unwanted applications (PUA or Greyware) and clean files (FP) as well.

Additionally, besides security capabilities tests, our assessment measured the footprint each security software on a computer’s performance.

Executive Summary

This Certification Programme is designed to serve as a reflection of product efficacy based on what we have previously termed “metrics that matter”.

Based on decades of experience in IT security, our previous tests, and being one of the world’s largest supplier of early-life malicious binaries and URLs, we know that all endpoints can be infected, regardless of the security solutions employed. The question is not ‘if’, but ‘when’ a malicious binary hits the system.

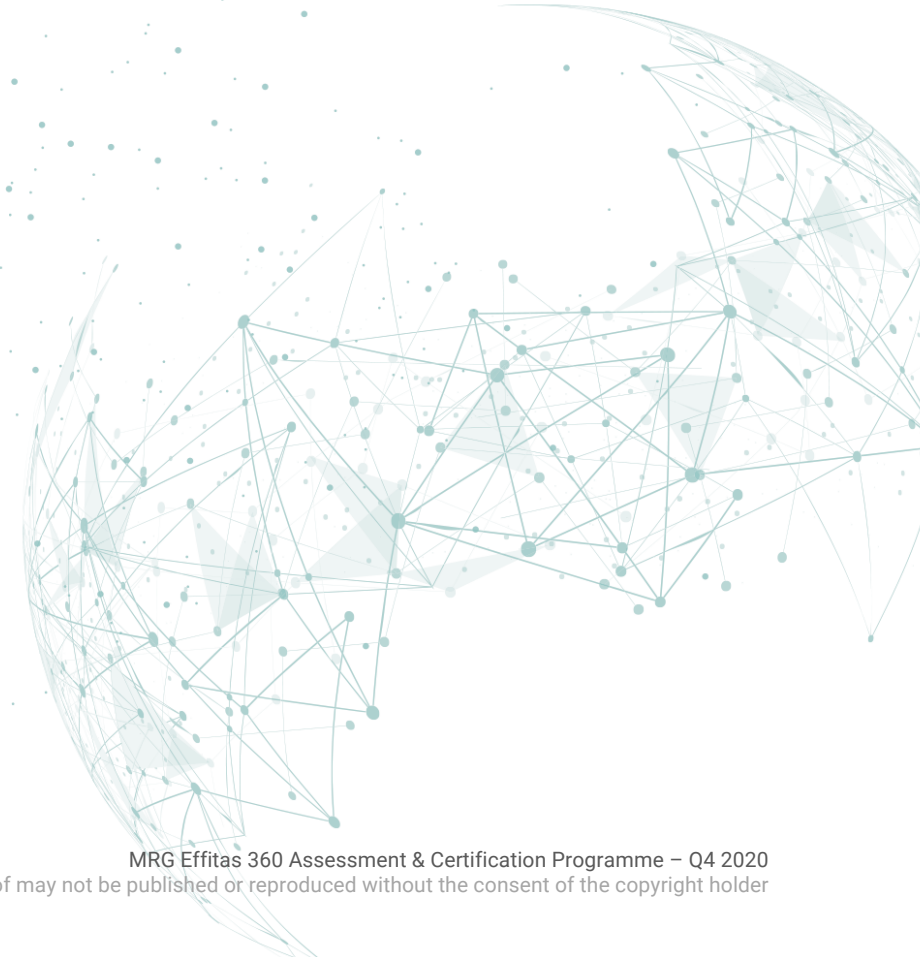
A security product’s ability to block initial infection (although critical in most cases) is not the only metric that matters. Measuring the time taken to detect malicious files or actions, is another metric that can be crucial in evaluation. An additional key factor is the point in time when the fact of the infection and any associated malicious behaviour are detected.

When conducting these tests, we tried to simulate normal user behaviour. We are aware that a “Real World” test cannot be conducted by a team of professionals inside a lab, because we understand how certain types of malware work, how malware attacks are conducted, and how such attacks could be prevented. Simulating normal user behaviour means that we paid special attention to all alerts given by security applications. A pass was given only when alerts were straightforward, and clearly suggested that the malicious action should be blocked.

With these in mind, it is very important to note that the best choice for an average user is to keep things as simple as possible and not to overwhelm the non-tech savvy with cryptic pop-ups, alerts or questions.

Out of 10 tested security products, the following six managed to meet the specification to attain our Q4 2020 360 Degree Certification.

- Bitdefender Endpoint Security
- ESET Endpoint Security
- F-Secure Computer Protection Premium
- Malwarebytes Endpoint Protection
- Microsoft Windows Defender
- Sophos Intercept X



Certification

In order to attain a quarterly MRG Effitas 360 Degree Level 1 certification, a security application must completely protect the system from initial infection either by automatically blocking every ITW sample, or by blocking them based on their behaviour, prior to any malicious actions and the product must pass the Live Botnet test. PUA, FP, Exploit/Fileless, Financial Malware Simulator, and performance tests are not part of the certification.

Level 2 certification is given if the application blocks or detects any initially missed malware in at least 98% of all cases on the 24-hour retest, while the initially missed test cases are less than 10%. If a ransomware/wiper successfully runs and the files are not available anymore, Level 2 certification is lost.

Under the MRG Effitas 360 Degree Assessment & Certification, the following products were certified for Q4 2020.

Certified (Level 1):

- Bitdefender Endpoint Security
- Malwarebytes Endpoint Protection
- Sophos Intercept X

Certified (Level 2):

- ESET Endpoint Security
- F-Secure Computer Protection Premium
- Microsoft Windows Defender

The Purpose of this Report

Since its inception in 2009, MRG Effitas has strived to differentiate itself from traditional testing houses by having its primary focus on providing “efficacy assessments” and not just performing “tests”.

Traditionally, testing of security software has been aimed at measuring a product’s ability to detect malware. Testing has evolved rapidly over the last couple of years, as most labs, under the direction of AMTSO (of which MRG Effitas is a member) has been striving to conduct “Real World” testing, based on standardised guidelines. More information about the compliance status of this test can be found on the AMTSO website.

https://www.amtso.org/amtso-ls1-tp031

Although there is no absolute definition of this kind of testing, loosely speaking, it involves the introduction of malware to an endpoint through a realistic entry point, such as downloading the sample using a browser or getting it from a USB memory stick. Real world testing mostly involves “dynamic testing” (i.e. the malware is executed and then the ability of the security product to block the malware is measured).

Whilst these types of tests are useful, yielding valid and meaningful data, MRG Effitas wanted to merge standalone tests and also go the extra mile by measuring the time security products take to detect infections and remediate the endpoint.

To make testing more akin to real world scenarios, no manual scanning was conducted. Instead, the system was retested exactly 24 hours after the system was compromised, thereby giving security applications the opportunity to detect infections on restart.

As we have stated in our previous test reports, most malware has one primary objective, and that is to make money for the cybercriminals.

Measuring initial detection rates and the time taken to detect active malware is important, particularly in today’s threat landscape with the mix of malware that is prevalent.

As we have repeated in our previous financial malware test reports, the longer a cybercriminal can run their malware on a system, the greater the opportunity is for them to be able to capture private user information, including banking passwords and social media credentials, etc.

For these types of malware, initial detection is of the utmost importance, since the vast majority of security solutions will be unable to remediate the problem of an encrypted system. In incident response scenarios, it is usually advised to purchase bitcoin right away, should the experts be unable to retrieve the encrypted files.

In providing these quarterly certifications, the MRG Effitas 360 Degree Assessment & Certification Programme is the de facto standard by which security vendors, financial institutions and other corporations can attain the most rigorous and accurate determination of a product’s efficacy against the full spectrum of malware that is prevalent during the period.

Tests Employed

In this assessment (Q4 2020), we ran the following tests:

In the Wild 360 / Full Spectrum Test

Most of the malicious URLs used in this test were compromised legitimate websites, serving malware. We believe that such URLs pose the greatest danger to users, as this is the place where they least expect to get infected, and any URL based protection fails on them. Some URLs originate from our honeypots, or in case of ransomware and financial malware in particular, we used URLs from newly discovered distribution sites.

Malware delivered by URLs used in this test can be considered as zero-day in the true meaning of the phrase. This posed a significant challenge to the participant products.

~10% of the threats used in this test were introduced to the system via internal webmail sites. We have witnessed many SMBs being infected through internal webmails and lack of spam filtering. Downloading malware attachments from internal webmail sites bypass the URL blocking features of the products, and this happens in-the-wild.

During the In the Wild 360 / Full Spectrum test, 360 live ITW samples were used. The stimulus load comprised the following: 52 trojans, 42 backdoors, 56 financial malware samples, 41 ransomware samples, 45 spyware, 39 malicious documents, 37 malicious emails, 48 malicious script files.

PUA / Adware Test

The PUA samples used in this test are deceptive, or potentially unwanted applications (PUA), that are not malicious, but are generally considered unsuitable for most home or business networks. They usually contain adware, installs toolbars or have other unclear objectives. They may also contribute to consuming computing resources or network bandwidth. PUAs can be deceptive, harmful, hoax, show aggressive popups and misleading or scaring the user. They may provide some unconventional ways of uninstalling the application, maybe retain some of their components on the device without the user's consent. We mainly use a filtered version of AppEsteem's feed, as they developed deceptor requirements as part of a cross-industry effort of many of the world's leading security companies and represent a minimum bar that all apps and services must meet to avoid being titled deceptive.

AppEsteem, as a member of the AMTSO group is dedicated to help protecting consumers from harassing and objectionable material, and to enable security companies to restrict access to such actions. MRG Effitas, as a member of the AMTSO group, is also dedicated to protecting these thoughts.

In the PUA/Adware part we tested the products against 20 PUAs.

Exploit/Fileless Test

The main purpose of this test is to see how security products protect against a specific exploitation technique. In order to measure this, we developed test cases that simulate the corresponding exploit and post-exploitation techniques only.

Drive-by download exploits are the biggest threats for an enterprise environment, since no user interaction is needed to start the chain of infection on a victim machine. Outdated browsers and Office environments are widespread in enterprise environments, due to compatibility issues or the lack of proper patch management process.

We were testing the products' abilities to avoid any exposure to adversaries, to interrupt malicious payload delivery before performing malicious actions. We focus explicitly on each product's ability to mitigate each attack technique. The results are not intended to evaluate the complete efficacy of the products, but rather the products' anti-exploit and anti-post-exploit features in isolation.

During this test we used 9 different exploitation techniques. The detailed description can be found in the 'Appendix'.

Botnet Test - TinyNuke

TinyNuke (aka Nuclear Bot, NukeBot) is a modular Zeus-style banking trojan. It was released via GitHub in 2016 by a Russian-speaking member who was the actor. The botnet has some built-in features, including HTML code injection but typically used to steal web services credentials. It has three major components: C&C server, Portable Executable file (the bot) and a DLL loaded into memory.

Simulator Test - Obfuscated ZombieBrowserPack

The obfuscated version of ZombieBrowserPack was developed for educational and testing purposes. This is a fully functional credential stealer browser extension for Firefox, Chrome and Safari.

False Positive Test

Perfect blocking of malicious content is only part of the story from a practical point of view for any decent AV product. In many cases all malware blocking is a result of a very aggressive filter which can block non-malicious legitimate applications as well prohibiting everyday work by blocking legitimate, perhaps newly developed in-house software.

In order to test this feature, we tested the security applications against completely clean, recently created applications.

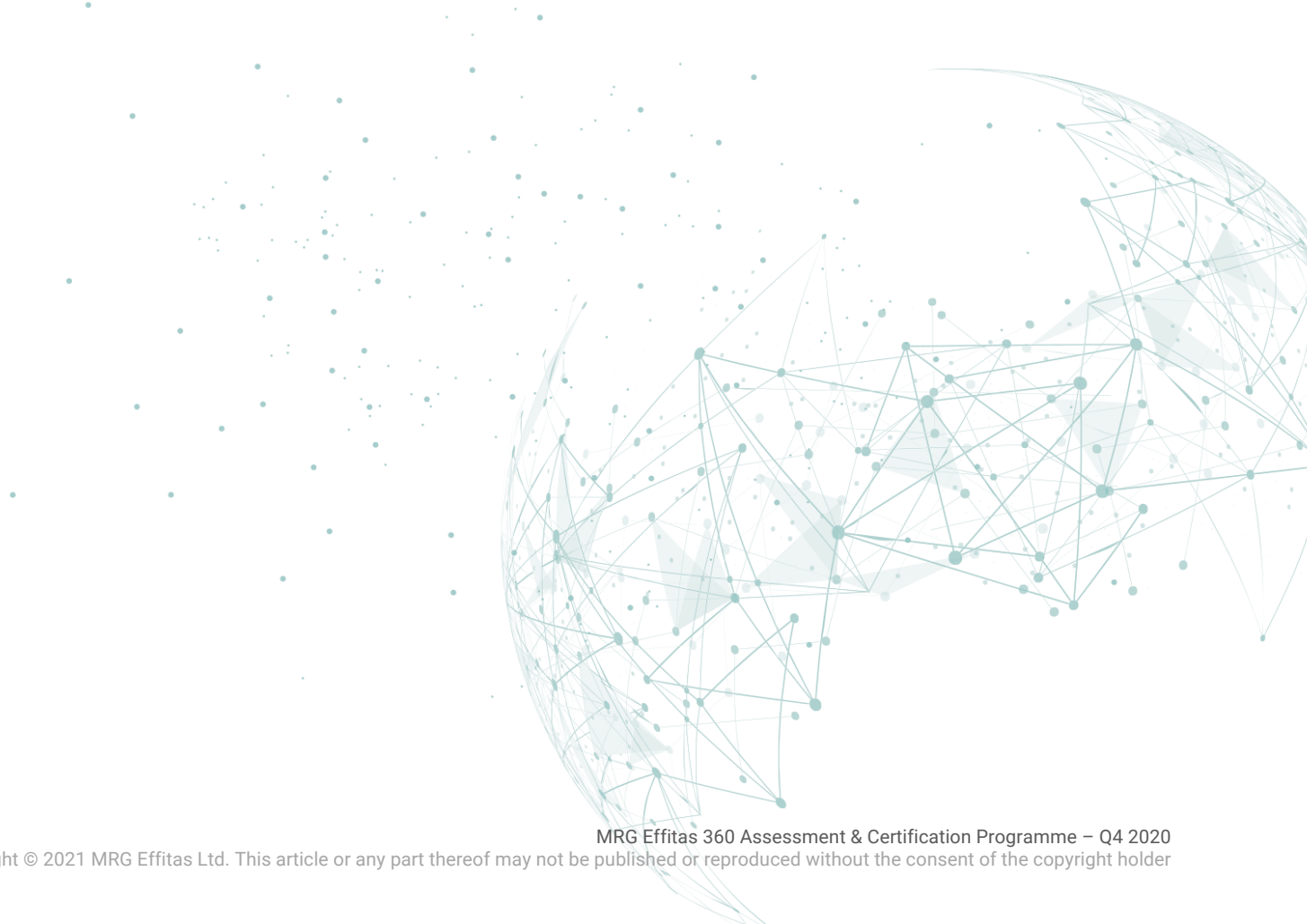
False positive assessment consisted of 1000 clean and legitimate application samples. The selection has been focused on applications, frequently found in enterprise environments (drivers, media editors, developer tools, etc.)

Performance Test

A security product's usefulness does not depend on protection level solely, but also on its resource footprint and its effect of the overall operating system performance.

In order to assess the products' influence on the operating system, we tested several performance factors on a physical machine and combined the results, based on a scoring approach. Detailed information can be found in the 'Appendix'.

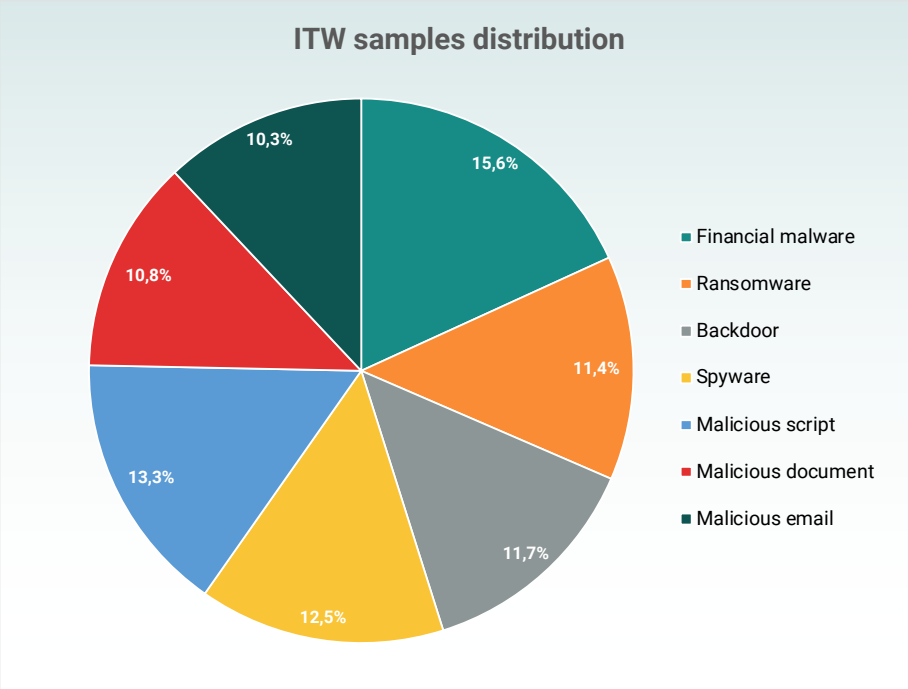
In every test case, (except for the performance test) our testing environment supports the execution of VM-aware malware, this is the reason why we were able to use more sophisticated threats which normally would not run on Virtual Machines.



Security Applications Tested

- Avast Business Antivirus 20.10.2625
- Avira Antivirus Pro 15.0.2101.2069
- Bitdefender Endpoint Security 6.6.23.329
- ESET Endpoint Security 7.3.2036.0
- F-Secure Computer Protection Premium 20.1 (4.12.13.68.0/PSB)
- Malwarebytes Endpoint Protection 1.2.0.844
- Microsoft Windows Defender 4.18.1911.3
- Sophos Intercept X 2.0.18
- Symantec Endpoint Protection 14.3.3384.1000
- Trend Micro Security 6.7.1478/14.2.1243

Malware sample types used to conduct the tests

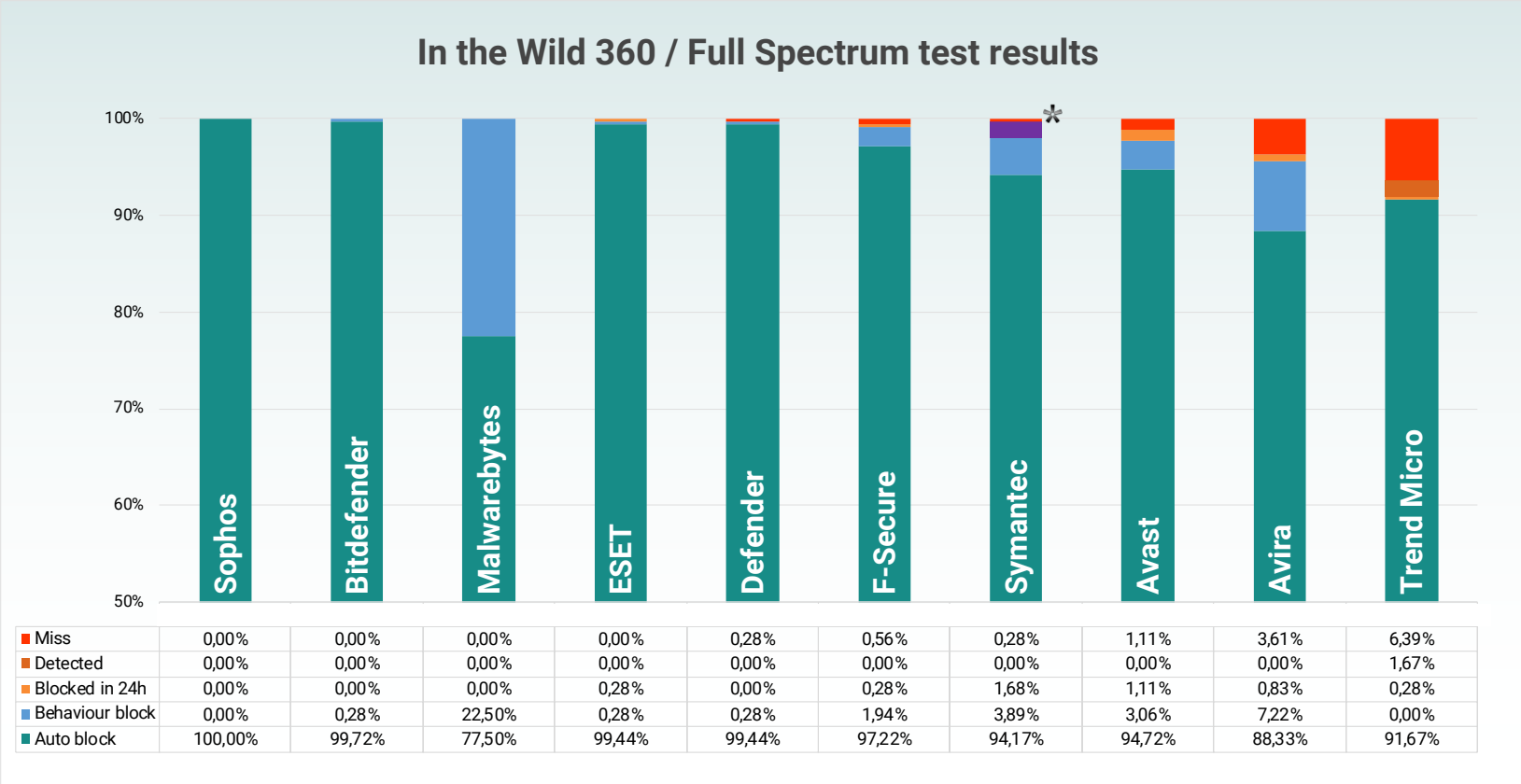


Test Results

The tables below show the results of testing under the MRG Effitas 360 Q4 2020 Assessment Programme.

Q4 2020 In the Wild 360 / Full Spectrum test results

The table below shows the detection rates of the security products for 360 ITW samples. This table is sorted by smallest number of missed samples.

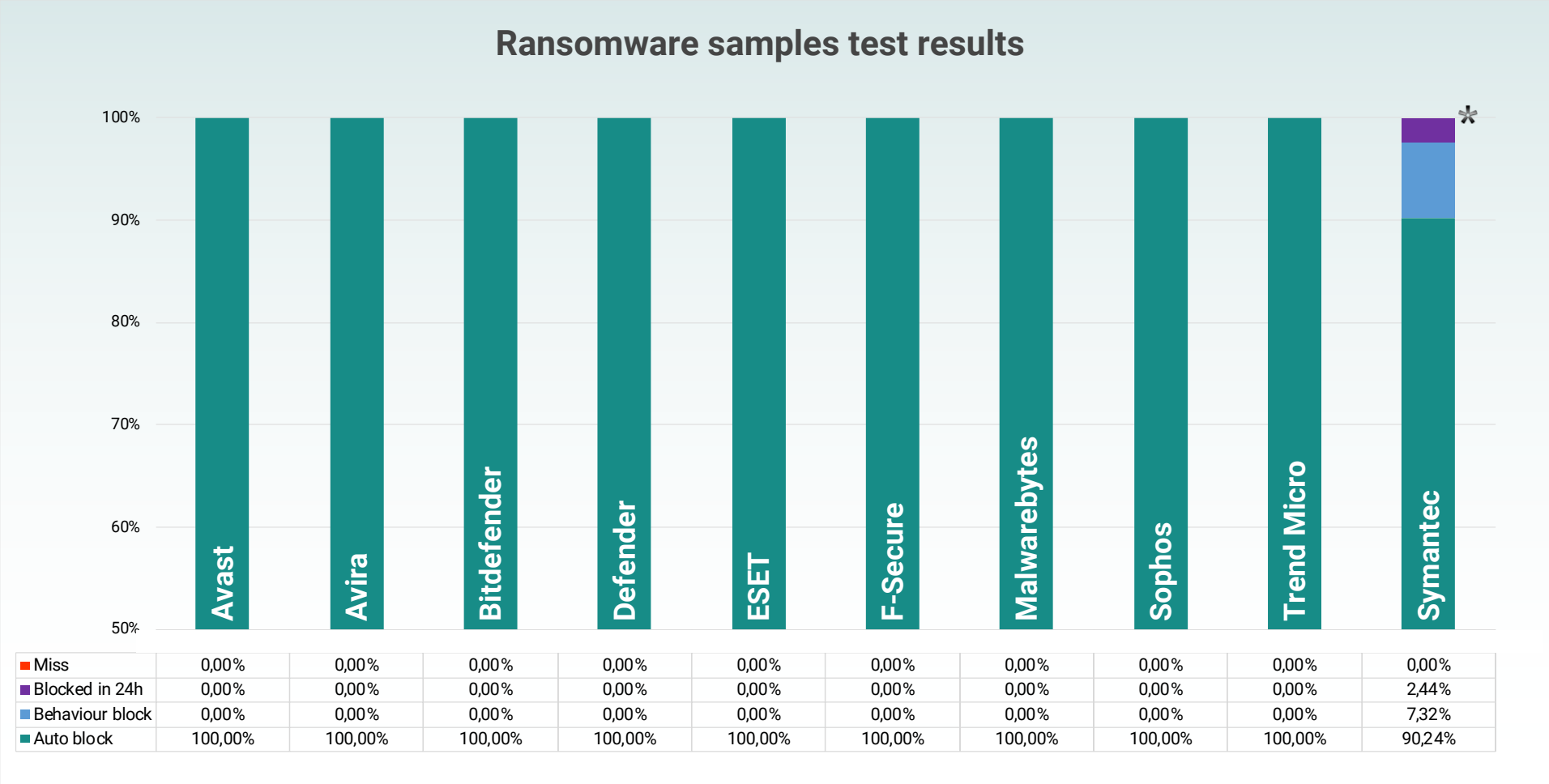


	Sophos	Bitdefender	Malwarebytes	ESET	Defender	F-Secure	Symantec	Avast	Avira	Trend Micro
Miss	0,00%	0,00%	0,00%	0,00%	0,28%	0,56%	0,28%	1,11%	3,61%	6,39%
Detected	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	1,67%
Blocked in 24h	0,00%	0,00%	0,00%	0,28%	0,00%	0,28%	1,68%	1,11%	0,83%	0,28%
Behaviour block	0,00%	0,28%	22,50%	0,28%	0,28%	1,94%	3,89%	3,06%	7,22%	0,00%
Auto block	100,00%	99,72%	77,50%	99,44%	99,44%	97,22%	94,17%	94,72%	88,33%	91,67%

*Blocking of samples indicated with purple colour were affected by a network configuration mismatch with Symantec Endpoint Protection.

Ransomware samples test results

The table below shows the detection rates of the security products for 41 ransomware samples. This table is sorted by smallest number of missed samples.

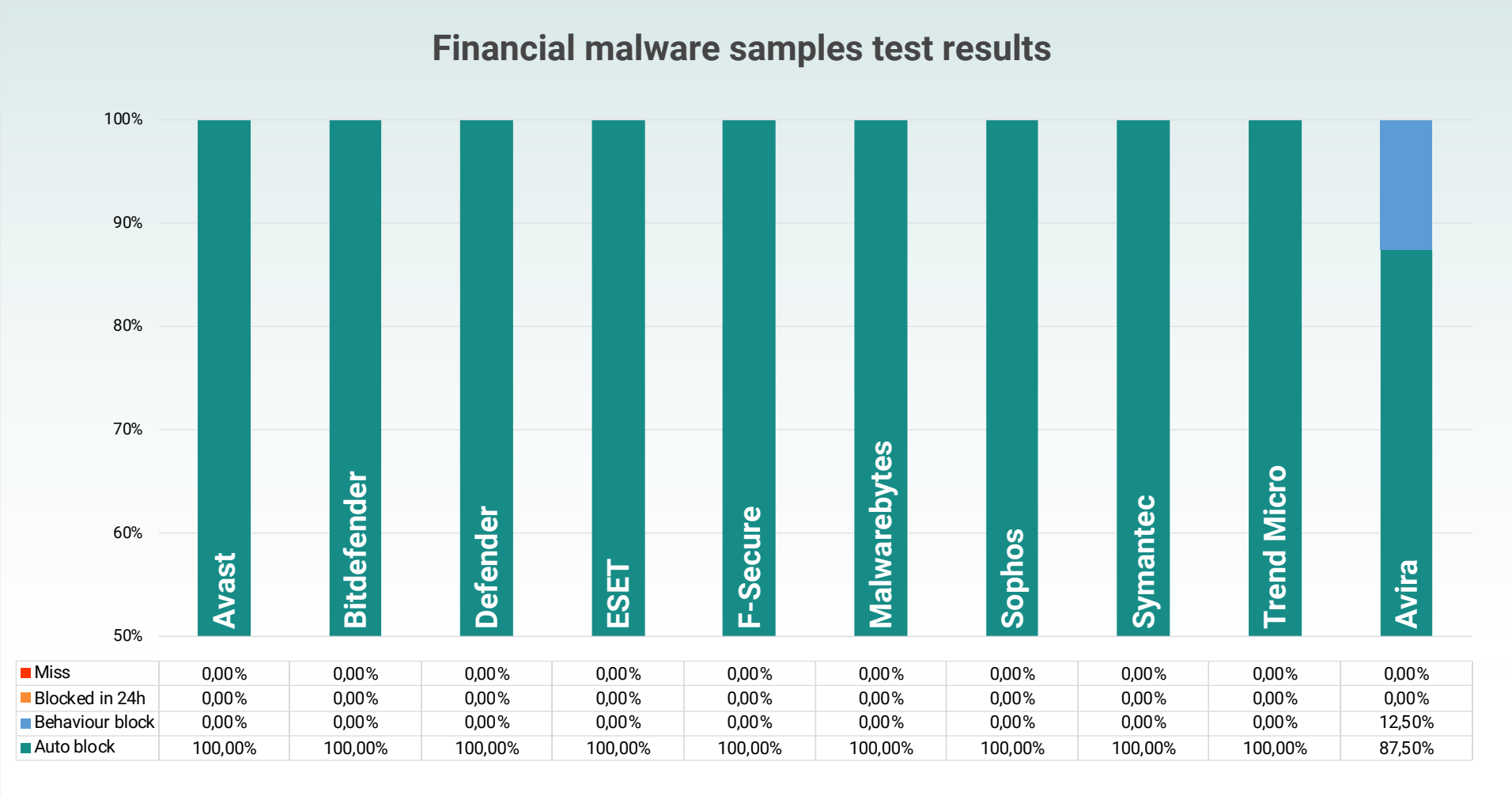


	Avast	Avira	Bitdefender	Defender	ESET	F-Secure	Malwarebytes	Sophos	Trend Micro	Symantec
Miss	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Blocked in 24h	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	2,44%
Behaviour block	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	7,32%
Auto block	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	90,24%

*Blocking of samples indicated with purple colour were affected by a network configuration mismatch with Symantec Endpoint Protection.

Financial malware samples test results

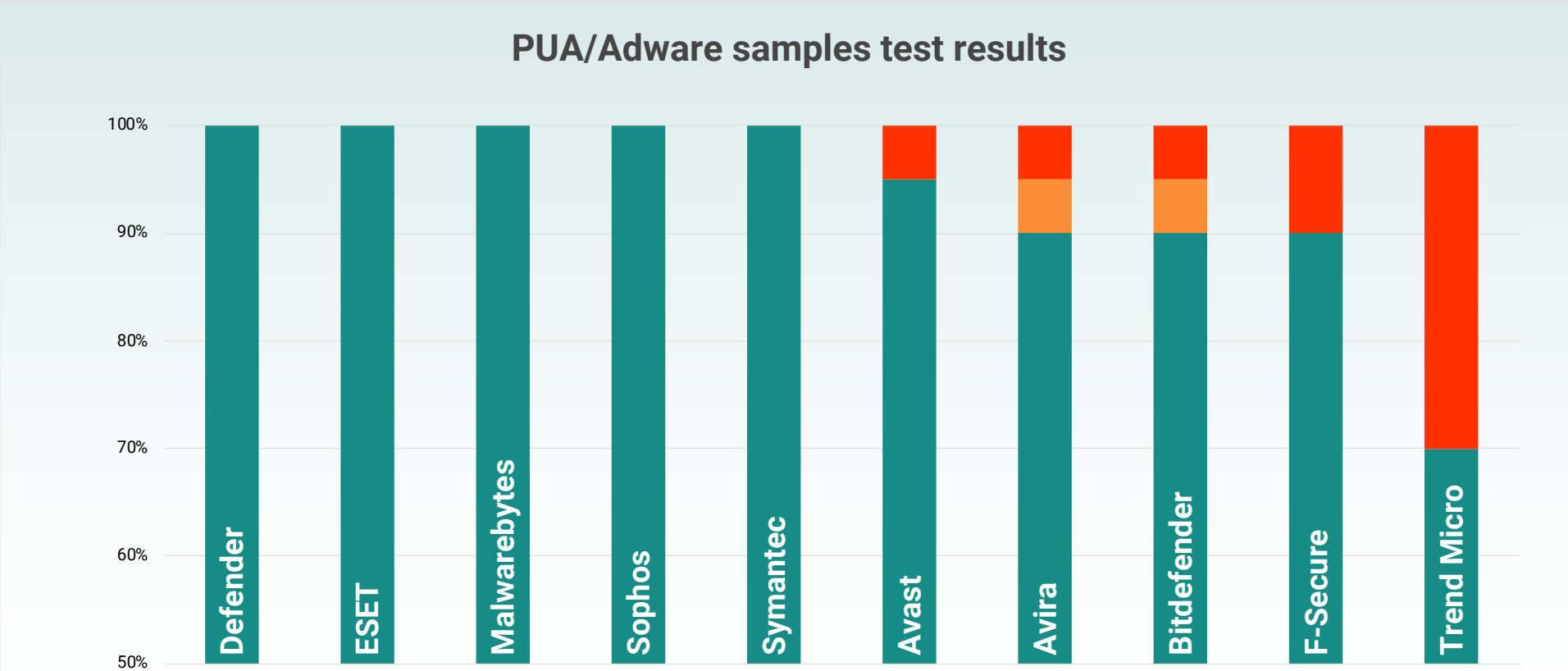
The table below shows the detection rates of the security products for 56 financial malware samples. This table is sorted by smallest number of missed samples.



	Avast	Bitdefender	Defender	ESET	F-Secure	Malwarebytes	Sophos	Symantec	Trend Micro	Avira
Miss	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Blocked in 24h	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Behaviour block	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	12,50%
Auto block	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	87,50%

PUA/adware samples test results

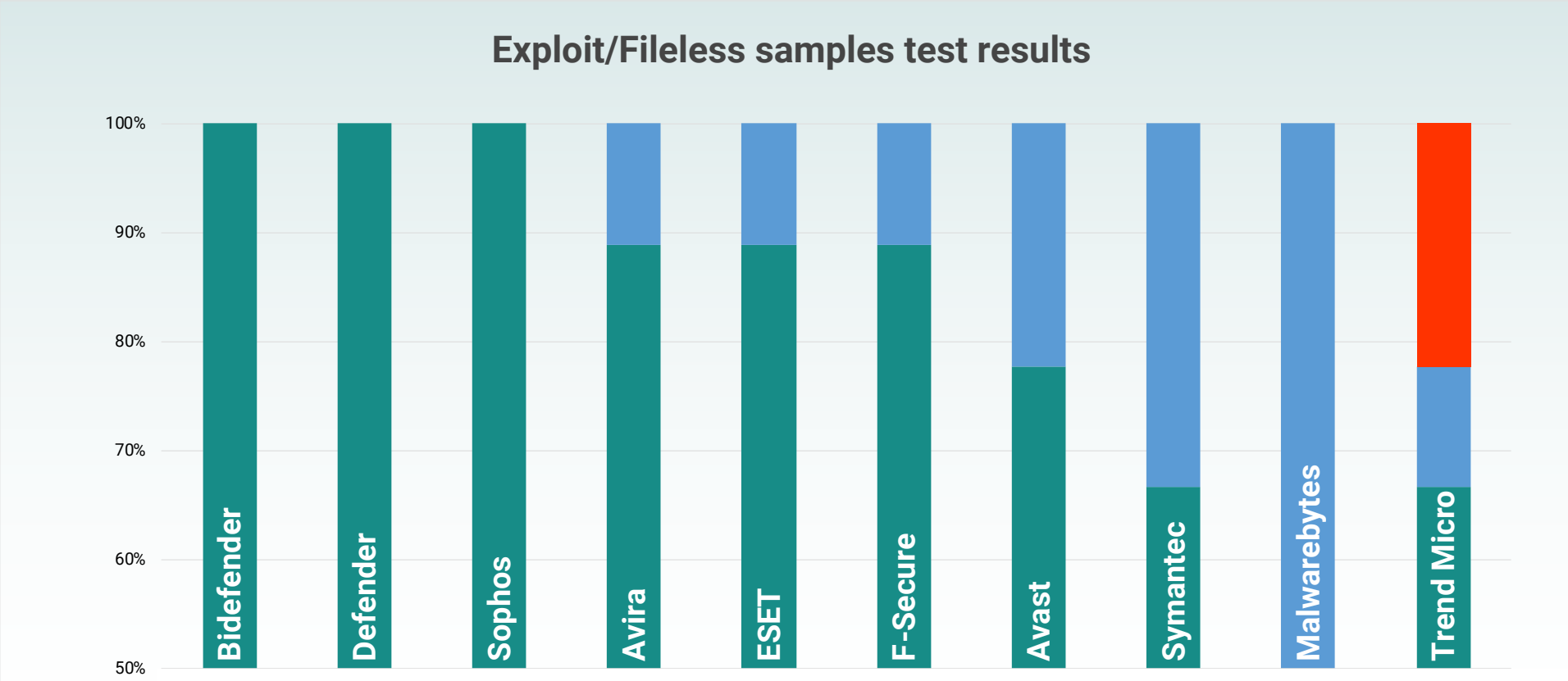
The table below shows the detection rates of the security products for 20 PUA/Adware samples. This table is sorted by smallest number of missed samples.



	Defender	ESET	Malwarebytes	Sophos	Symantec	Avast	Avira	Bitdefender	F-Secure	Trend Micro
Miss	0,00%	0,00%	0,00%	0,00%	0,00%	5,00%	5,00%	5,00%	10,00%	30,00%
Blocked in 24h	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	5,00%	5,00%	0,00%	0,00%
Behaviour block	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%
Auto block	100,00%	100,00%	100,00%	100,00%	100,00%	95,00%	90,00%	90,00%	90,00%	70,00%

Exploit/fileless samples test results

The table below shows the initial detection rates of the security products for 9 exploit/fileless test. This table is sorted by smallest number of missed attack vectors.



	Bidefender	Defender	Sophos	Avira	ESET	F-Secure	Avast	Symantec	Malwarebytes	Trend Micro
Miss	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	22,22%
Signature block	0,00%	0,00%	0,00%	11,11%	11,11%	11,11%	22,22%	33,33%	100,00%	11,11%
Auto block	100,00%	100,00%	100,00%	88,89%	88,89%	88,89%	77,78%	66,67%	0,00%	66,67%

Real Botnet test results

The table below shows the results of live financial malware test.

Botnet test	
Vendor	**TinyNuke**
Avast Business Antivirus	✓
Avira Antivirus Pro	✓
Bitdefender Endpoint Security	✓
ESET Endpoint Security	✓
F-Secure Computer Protection Premium	✓
Malwarebytes Endpoint Protection	✓
Microsoft Windows Defender	✓
Sophos Intercept X	✓
Symantec Endpoint Protection	✓
Trend Micro Security	✓

✓ The application prevented the malware from capturing login data

✗ The application failed to prevent the malware from capturing login data

Financial malware simulator test results

The table shows the results of financial malware simulator test.

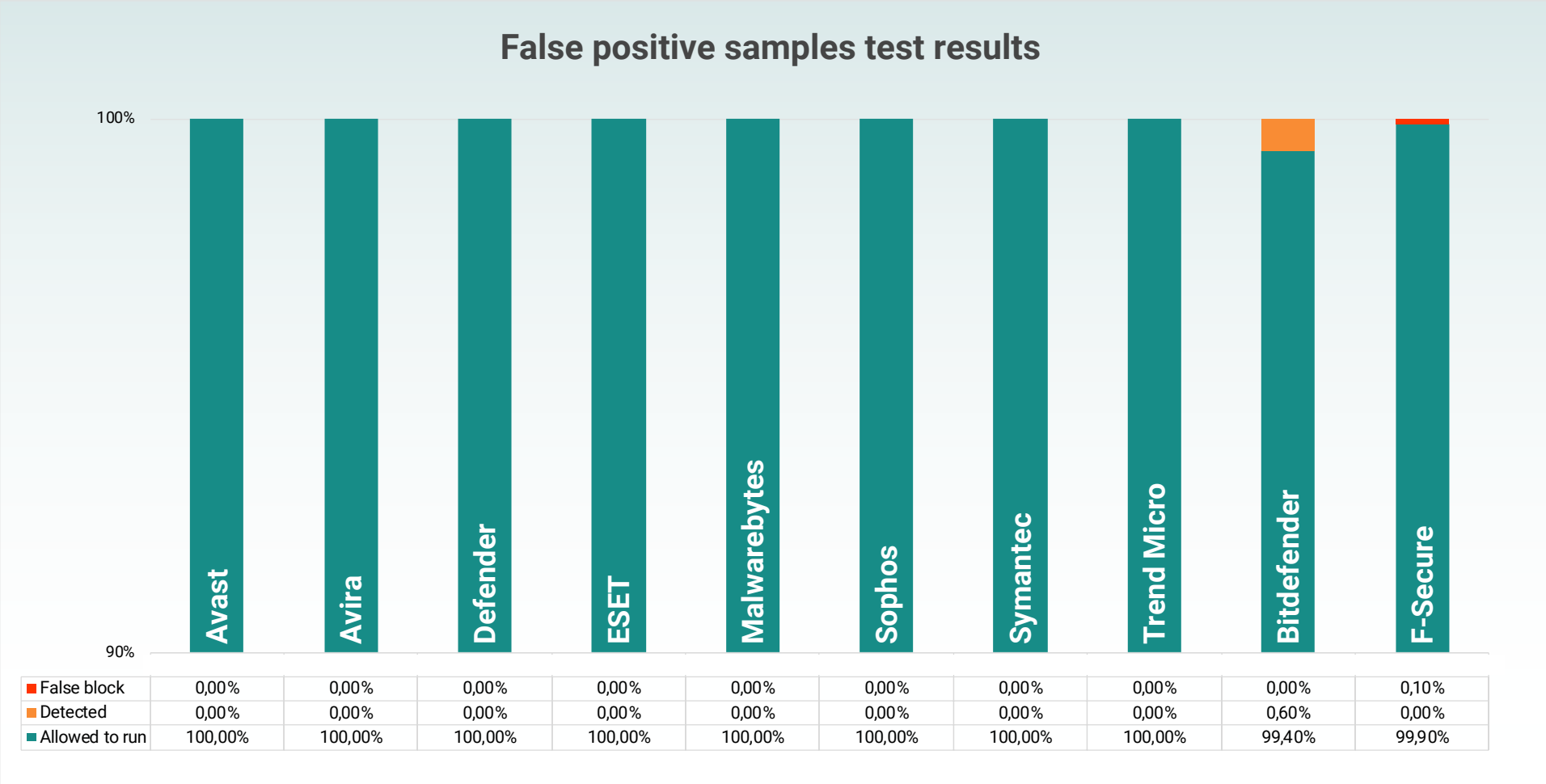
Financial malware simulator test	
Vendor	**ZombieBrowserPack**
Avast Business Antivirus	✓
Avira Antivirus Pro	✓
Bitdefender Endpoint Security	✓
ESET Endpoint Security	✗
F-Secure Computer Protection Premium	✓
Malwarebytes Endpoint Protection	✓
Microsoft Windows Defender	✗
Sophos Intercept X	✗
Symantec Endpoint Protection	✓
Trend Micro Security	✗

✓ The application blocked the simulator

✗ The application failed to block the simulator

False positive samples test results

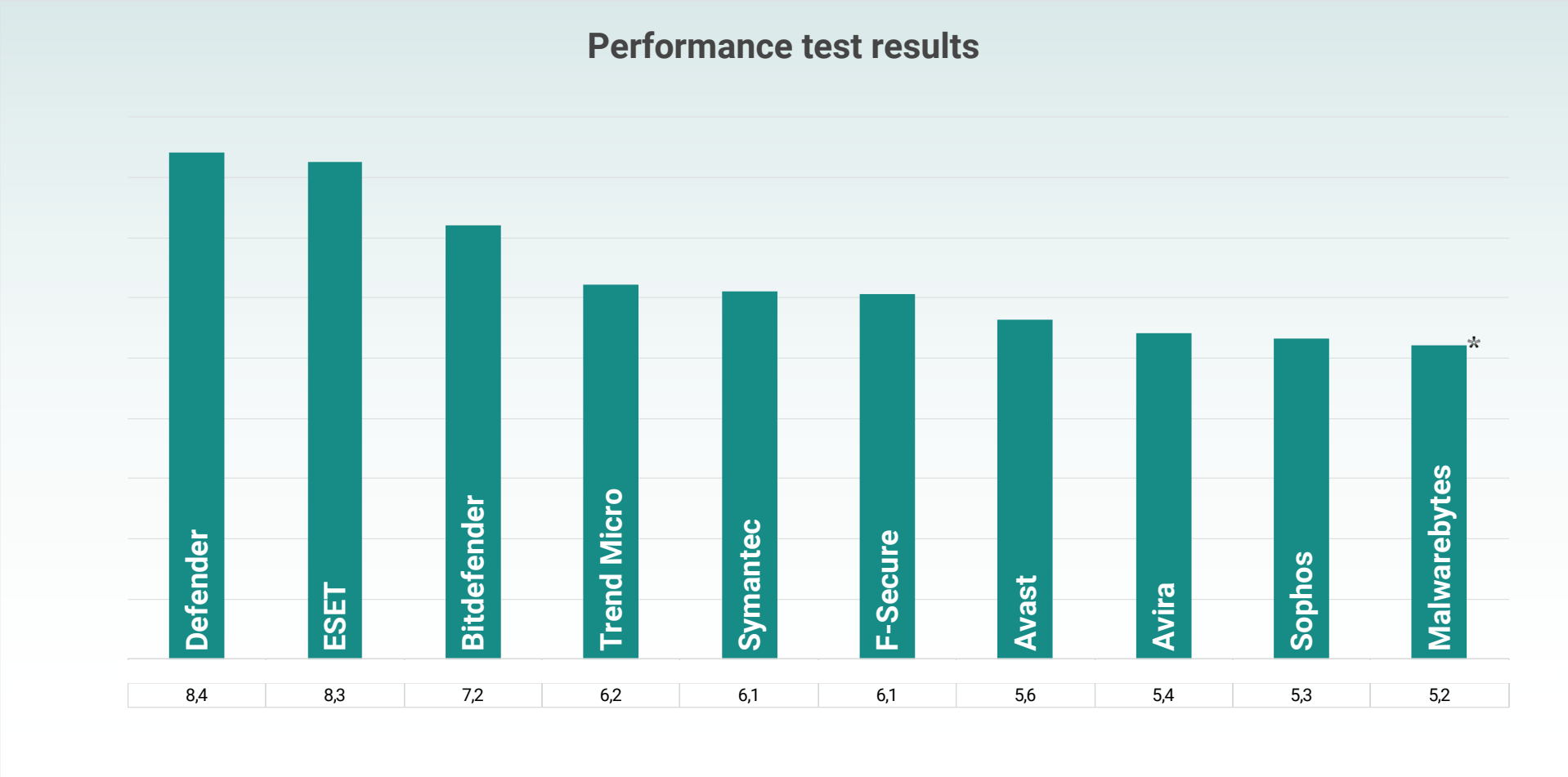
The table below shows the initial detection rates of the security products for 1108 false positive samples. This table is sorted by smallest number of false positive sample blocks.



	Avast	Avira	Defender	ESET	Malwarebytes	Sophos	Symantec	Trend Micro	Bitdefender	F-Secure
False block	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,10%
Detected	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	0,60%	0,00%
Allowed to run	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	99,40%	99,90%

Performance test results

This table is sorted from highest to lowest score where the highest score denotes the lowest impact on the system.



* Malwarebytes Endpoint Protection's performance was impacted due to unnecessary components were enabled.

Scoring details can be found in the 'Appendix'.

Detailed results of the performance test

The table below shows the detailed results of the performance test of the security products. This table is sorted alphabetically.

	Windows 10 Base	Avast	Avira	Bitdefender	Defender	ESET	F-Secure	Malwarebytes	Sophos	Symantec	Trend Micro
Bootup time (s)	29,6	53,5	58,2	38,5	31,0	34,7	34,5	44,5	49,8	34,5	41,7
Security software size on disk (Mb)	n/a	1242,6	971,9	1026,6	419,5	736,1	1089,5	343,8	1950,6	1259,0	753,2
Browser Operations (s)											
Website Open	2,7	3,5	3,2	3,2	2,5	3,7	3,3	4,5	3,6	3,2	4,1
File Download	10,1	15,6	12,5	12,4	10,6	12,2	11,1	15,0	18,1	13,4	12,3
File Operations (s)											
File Copy	2,1	2,0	1,9	2,2	2,2	2,2	2,3	1,9	1,8	2,0	2,4
File Compression	37,3	38,2	37,9	38,4	37,5	37,4	37,6	48,6	38,5	38,6	42,4
Archive Extraction	6,0	6,4	6,9	6,5	8,2	5,6	9,8	7,4	17,8	9,4	28,4
Office File Opening (s)											
Excel	6,2	6,7	7,4	7,0	6,4	6,4	6,6	8,6	8,1	6,4	7,2
Word	2,4	2,7	3,7	2,6	2,5	2,5	2,7	5,1	3,3	4,8	2,9
Security software update											
Time (s)	n/a	37,3	49,7	159,7	23,0	31,7	26,0	n/a	67,3	n/a	35,0
CPU usage (%)	n/a	43,3	38,2	26,0	37,7	36,9	37,5	n/a	26,8	n/a	25,0
Memory usage (Mb)	n/a	710,1	770,1	725,8	495,8	312,4	786,6	n/a	938,7	n/a	601,6
Physical disk usage (%)	n/a	41,7	14,4	9,2	13,0	62,5	89,5	n/a	8,2	n/a	17,5
Network interface usage (B/s)	n/a	69633,1	207215,8	32100,4	178351,5	74494,3	104954,7	n/a	7998,5	n/a	160228,1
Security software scanning - C:\											
Time (s)	n/a	407,7	608,3	112,0	361,7	206,0	491,3	398,7	618,0	n/a	450,3
CPU usage (%)	n/a	22,8	22,7	38,4	67,1	27,8	81,0	26,5	29,1	n/a	32,7
Memory usage (Mb)	n/a	1088,9	885,9	865,5	914,2	415,8	902,4	1082,4	1223,9	n/a	821,7
Physical disk usage (%)	n/a	42,7	20,8	24,8	26,6	50,8	22,5	38,3	8,6	n/a	15,3
Network interface usage (B/s)	n/a	1872,2	15157,5	1955,9	621,0	1363,2	1008,3	30907,1	1103,2	n/a	929,6

Understanding Grade of Pass

Level 1 certified

All threats detected on first exposure or via behaviour protection and Live Botnet test is passed.

- Bitdefender Endpoint Security
- Malwarebytes Endpoint Protection
- Sophos Intercept X

Level 2 certified

At least 98% of the threats detected and neutralised / system remediated before or on the first rescan while the initially missed test cases are less than 10%.

- ESET Endpoint Security
- F-Secure Computer Protection Premium
- Microsoft Windows Defender

Not certified

Security product failed to detect at least 98% of the infections and remediate the system during the test procedure.

- Avast Business Antivirus
- Avira Antivirus Pro
- Symantec Endpoint Protection
- Trend Micro Security

Appendix 1

Methodology used in the “In the Wild 360 / Full Spectrum” test

1. Windows 10 Enterprise 64-bit operating system is installed on a hardened virtual machine, all updates are applied, and third-party applications installed and updated.
2. An image of the operating system is created.
3. A clone of the imaged systems is made for each of the security applications used in the test.
4. An individual security application is installed using default settings on each of the systems created in (3) and then, where applicable, updated. If the vendor provided a non-default setting, this setting is checked whether it is realistic. If yes, the changes are documented, applied, and added to the appendix section of the report.
5. A clone of the system as at the end of (4) is created.
6. Downloading a single binary executable (or document, script, etc.) from its native URL using Chrome to the Downloads folder and then executing the binary in the clean, unprotected system. If the sample works, the sample is saved in a replay proxy to provide the same binary throughout the test.

Live URL test is conducted by the following procedure.

6.1. The sample is selected for the test and tested in the systems where a security product is installed.

6.2. The test case is retested 24 hours after the initial test if the security application failed to block the malicious binary.

Spam e-mail attachment test is conducted by the following procedure.

6.3. Microsoft Office Outlook client downloading a single email from its server to the victim system created in (4).

6.4. Opening the e-mail, saving the attachment to the Downloads folder and then executing the binary.

6.5. The test case is retested 24 hours after the initial test if the security application failed to block the malicious binary.

- **The test case is marked as “Blocked”** by either the security application blocks the URL where the malicious binary was located. Or the security application blocks the malicious binary whilst it was being downloaded to the desktop.
- **The test case is marked as “Behaviour Blocked”** if the security application blocks the malicious binary when it is executed and either automatically blocks it or postpones its execution and warns the user that the file is malicious and awaiting user input.

- **The test case is marked as "Detected"** if the security application detects the threat and sends an alert to the central console or notifies the user, but the sample is allowed to run.
- **The test case is marked as "Blocked in 24h"** if the security application fails to block or behaviour block the malicious sample but blocks it during the retest.
- **The test case is marked as "Missed"** if the security application fails to block or behaviour block the malicious sample during both tests.

7. Tests are conducted with all systems having internet access.
8. As no user-initiated scans is involved in this test, applications rely on various technologies to detect, block and remediate threats. Some of these technologies are URL blacklisting, reputation, signature, machine learning, heuristics, behaviour etc.

Methodology used in the "In-The-Wild PUA/Adware" test

1. Windows 10 Enterprise 64-bit operating system is installed on a hardened virtual machine, all updates are applied, and third-party applications installed and updated.
2. An image of the operating system is created.
3. A clone of the imaged systems is made for each of the security applications used in the test.
4. An individual security application is installed using default settings on each of the systems created in (3) and then, where applicable, updated. If the vendor provided a non-default setting, this setting is checked whether it is realistic. If yes, the changes are documented, applied, and added to the appendix section of the report.
5. A clone of the system as at the end of (4) is created.
6. Downloading a single binary executable (or document, script, etc.) from its native URL using Chrome to the Downloads folder and then executing the binary in the clean, unprotected system. If the sample works, the sample is saved in a replay proxy to provide the same binary throughout the test.
7. The sample is selected for the test and tested in the systems where a security product is installed.
8. The test case is retested 24 hours after the initial test if the security application failed to block the malicious binary.

- **The test case is marked as "Blocked"** by either the security application blocks the URL where the malicious binary was located. Or the security application blocks the malicious binary whilst it was being downloaded to the desktop.

- **The test case is marked as "Behaviour Blocked"** if the security application blocks the malicious binary when it is executed and either automatically blocks it or postpones its execution and warns the user that the file is malicious and awaiting user input.
- **The test case is marked as "Detected"** if the security application detects the threat and sends an alert to the central console or notifies the user, but the sample is allowed to run.
- **The test case is marked as "Blocked in 24h"** if the security application fails to block or behaviour block the malicious sample but blocks it during the retest.
- **The test case is marked as "Missed"** if the security application fails to block or behaviour block the malicious sample during both tests.

9. Tests are conducted with all systems having internet access.

As no user-initiated scans is involved in this test, applications rely on various technologies to detect, block and remediate threats. Some of these technologies are URL blacklisting, reputation, signature, machine learning, heuristics, behaviour etc.

Methodology used in the false positive test

1. Windows 10 Enterprise 64-bit operating system is installed on a hardened virtual machine, all updates are applied, and third-party applications installed and updated.
2. An image of the operating system is created.
3. A clone of the imaged systems is made for each of the security applications used in the test.
4. An individual security application is installed using default settings on each of the systems created in (3) and then, where applicable, updated. If the vendor provided a non-default setting, this setting is checked whether it is realistic. If yes, the changes are documented, applied, and added to the appendix section of the report.
5. A clone of the system as at the end of (4) is created.
6. Introducing the binary executables (or documents, scripts, etc.) to the clean, unprotected system via disk image or network share. If the sample works, the sample is saved to a different disk image or network share.

False Positive test is conducted by the following procedure.

6.1. Scanning the binary executables (or documents, scripts, etc.) on the disk image or on the network share.

6.2. Executing the test samples.

6.3. The sample is retested 24 hours after the initial test if the security application failed to permit the harmless file.

- **The test case is marked as "False block"** if the security application falsely identifies and blocks the binary at any stage during the test and retest.
- **The test case is marked as "Detected"** if the security application falsely identifies and the binary at any stage during the test and retest but allows it to run.
- **The test case is marked as "Allowed to run in 24h"** if the security application falsely identifies and blocks the binary at any stage during the test but allows it to run upon the retest.
- **The test case is marked as "Allowed to run"** if the security application correctly identifies the binary as harmless and allows it to run.

7. Tests are conducted with all systems having internet access.

Methodology used in the exploit/fileless test

1. Windows 10 Enterprise 64-bit operating system is installed on a hardened virtual machine, all updates are applied, and third-party applications installed and updated.
2. An image of the operating system is created.
3. A clone of the imaged systems is made for each of the security applications used in the test.
4. An individual security application is installed using default settings on each of the systems created in (3) and then, where applicable, updated. If the vendor provided a non-default setting, this setting is checked whether it is realistic. If yes, the changes are documented, applied, and added in the report in an appendix.
5. A clone of the system as at the end of (4) is created.

Exploit / Fileless test is conducted by the following procedure.

6. Our payloads use an exploit for the one of an installed vulnerable application. In order to simulate a realistic attack scenario, a payload is constructed to include at least one of the common CnC frameworks.
7. The opening stage of the exploit is introduced to the system and we monitor if the vulnerable application starts the initial stage payload, the exploit is being executed and if a session is established to our CnC server.
8. After navigating to the exploit site, the system is supervised if there are any new processes, loaded DLLs or CnC traffic emerge. If the exploitation is successful, the following actions are executed.

8.1. Upload a file to the victim.

8.2. Download a file from the victim.

8.3. Create a process remotely.

8.4. Read the contents of a file on the victim.

9. When user interaction is needed from the endpoint protection (e.g. site visit not recommended, etc.) the default action is chosen. When user interaction is needed from the operating system, we chose the run/allow options.
10. Throughout the test, the Process Monitor from the Sysinternals Suite and Wireshark are running (both installed to non-default directories and modified not to be detected by default anti-debugging tools).

- **The test case is marked as "Signature Block"** if the security application blocks the URL (infected URL, exploit kit URL, redirection URL, malware URL) by the URL database (local or cloud).
- **The test case is marked as "Blocked"** if the security application blocks the page containing a malicious HTML code, JavaScript (redirects, iframes, obfuscated JavaScript, etc.) or Flash files. Or if the security application blocks the downloaded payload by analysing the malware before it can be started. (reputation-based block or heuristic based block).
- **The test case is marked as "Behaviour Blocked"** if the security application blocks the downloaded payload after it has been started.
- **The test case is marked as "Detected"** if the security application detects the threat and sends an alert to the central console or notifies the user, but the attack is allowed to run.
- **The test case is marked as "Missed"** if the security application fails to detect, block or behaviour block the attack and the it can be carried out.

11. Tests are conducted with all systems having internet access.
12. As no user-initiated scans is involved in this test, applications rely on various technologies to detect, block and remediate threats. Some of these technologies are URL blacklisting, reputation, signature, machine learning, heuristics, behaviour etc.

Detailed description of the Exploit / Fileless cases.

Test case 001

Koadic/wmic

Koadic is a framework using VBScript stagers for increased stealth and limited footprint. In this test case, a Koadic connectback payload is instantiated using a wmic command.

In case the exploitation was successful, as a proof of that working session has been establised, the following actions has been carried out through the connection.

- A directory list is queried
- A file is uploaded to the victim
- A file is downloaded
- A shell command is executed

The test case is flagged as MISSED if exploitation was successful and test machine had been successfully controlled via the new session.
References:
https://github.com/zerosum0x0/koadic

Test case 002

Koadic/mshta

Koadic is a framework using VBScript stagers for increased stealth and limited footprint. In this test case, a Koadic connectback payload is instantiated using a malicious Windows help .hta document.
In case the exploitation was successful, as a proof of that working session has been establised, the following actions has been carried out through the connection.

- A directory list is queried
- A file is uploaded to the victim
- A file is downloaded
- A shell command is executed

The test case is flagged as MISSED if exploitation was successful and test machine had been successfully controlled via the new session.
References:
https://github.com/zerosum0x0/koadic

Test case 003

Koadic/regsvr32

Koadic is a framework using VBScript stagers for increased stealth and limited footprint. In this test case, a Koadic connectback payload is instantiated using a regsvr32 remote object load call.
In case the exploitation was successful, as a proof of that working session has been establised, the following actions has been carried out through the connection.

- A directory list is queried
- A file is uploaded to the victim
- A file is downloaded
- A shell command is executed

The test case is flagged as MISSED if exploitation was successful and test machine had been successfully controlled via the new session.
References:
https://github.com/zerosum0x0/koadic

Test case 004

EMPIRE/.net compilation job

In this test case, we use the Empire PowerShell framework to create a crafted .net build job XML to spawn an Empire connectback shell upon a compilation process.
In case the exploitation was successful, as a proof of a working session, the following steps are taken.

- A screenshot has been made
- A file has been downloaded
- A file has been uploaded

The test case is flagged as MISSED if exploitation was successful and the test machine had been successfully controlled via the new session.
References:
https://github.com/EmpireProject/Empire/

Test case 005

EMPIRE/MSHTA

In this test case, we use the Empire PowerShell framework to create malicious Windows help document to spawn an Empire connectback shell upon a compilation process.
In case the exploitation was successful, as a proof of a working session, the following steps are taken.

- A screenshot has been made
- A file has been downloaded
- A file has been uploaded

The test case is flagged as MISSED if exploitation was successful and the test machine had been successfully controlled via the new session.
References:
https://github.com/EmpireProject/Empire/

Test case 006

Foxit reader Use After Free + Empire

In this test case, we use the Foxit Reader v9.0.1.1049 exploit (foxit_reader_uaf) to start the exploit chain. After successfully exploiting the vulnerability an Empire (PowerShell) stager is executed.

In case the exploitation was successful, as a proof of a working session, the following steps are taken.

- A screenshot has been made
- A file has been downloaded
- A file has been uploaded

The test case is flagged as MISSED if exploitation was successful and the test machine had been successfully controlled via the new session.

Exploited application: Foxit Reader v9.0.1.1049 OS version: Windows 7

CVE:

CVE-2018-9948
CVE-2018-9958

The exploit

Foxit Reader v9.0.1.1049 and earlier are affected by use-after-free and uninitialized memory vulnerabilities that can be used to gain code execution. This module uses Uint32Array uninitialized memory and text annotation use-after-free vulnerabilities to call WinExec with a share file path to download and execute the specified exe. The module has been tested against Foxit Reader v9.0.1.1049 running on Windows 7 x64 and Windows 10 Pro x64 Build 17134. Windows 10 Enterprise needs to have insecure logons enabled for the exploit to work as expected.

References:

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-9948
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-9958
https://www.rapid7.com/db/modules/exploit/windows/fileformat/foxit_reader_uaf
https://www.powershellempire.com/
https://github.com/EmpireProject/Empire.

Test case 007

Firefox version 31.0 exploit with Empire

In this test case, we target Firefox 31.0 with an exploit (CVE-2014-8636, CVE-2015-0802) starting the exploit chain. After successfully exploiting the vulnerability an Empire (PowerShell) stager is executed.

In case the exploitation was successful, as a proof of a working session, the following steps are taken.

- A screenshot has been made
- A file has been downloaded

- A file has been uploaded

The test case is flagged as MISSED if exploitation was successful and the test machine had been successfully controlled via the new session.

The exploit

This exploit gains remote code execution on Firefox 31-34 by abusing a bug in the XPConnect component and gaining a reference to the privileged chrome:// window. This exploit requires the user to click anywhere on the page to trigger the vulnerability.

CVE:

CVE-2014-8636

CVE-2015-0802

References:

https://www.rapid7.com/db/modules/exploit/multi/browser/firefox_proxy_prototype

https://www.powershellempire.com/

https://github.com/EmpireProject/Empire

Test case 008

MSBuild + Metasploit Meterpreter

In this test case, we target MSBuild starting the exploit chain. Assuming that MSBuild.exe is allowed since this tool is part of the Microsoft .NET Framework, we can invoke it to execute a .xml file as a Visual Studio .NET C# Project descriptor. The well-composed file contains a CSharp code which starts a Meterpreter stager. If code execution is not blocked, as a result, a new Meterpreter session back to MRG-Effitas CnC server will be created.

In case the exploitation was successful, as a proof of a working session, the following steps are taken.

- A screenshot has been made
- A file has been downloaded
- A file has been uploaded

The test case is flagged as MISSED if exploitation was successful and the test machine had been successfully controlled via the new session.

References:

https://ired.team/offensive-security/code-execution/using-msbuild-to-execute-shellcode-in-c

Test case 009

Code Injection via NtCreateSection (shellcode: bind shell)

In this test, we used a code injection technique that leverages Native APIs NtCreateSection, NtMapViewOfSection, and RtlCreateUserThread to inject code to a trusted process.

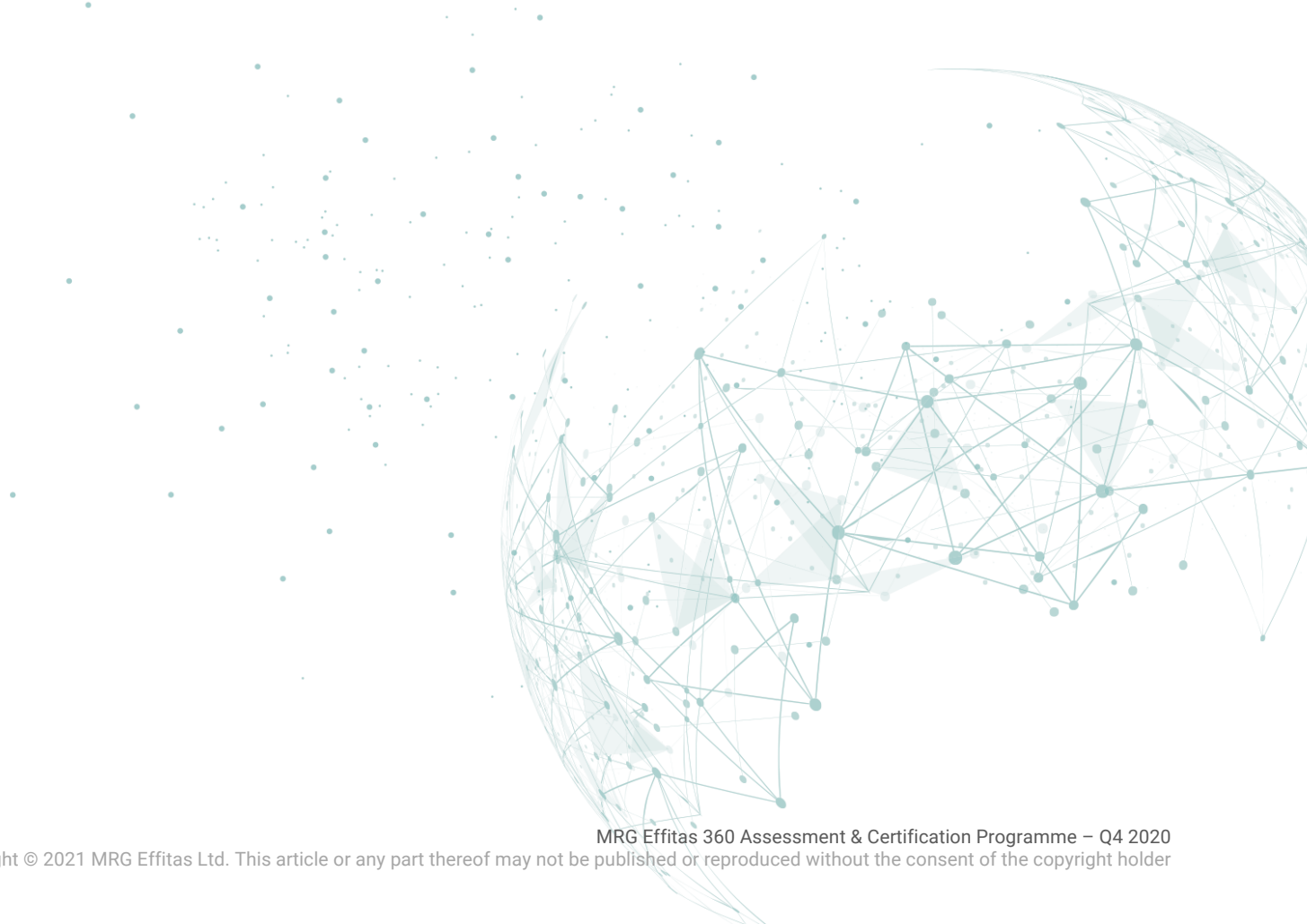
If the code successfully executed, bind shell shellcode is injected to the C:\Windows\System32\exprorer.exe. This payload accepts remote TCP connection and serve them by cmd.exe. Doing this, targeted machine can be controlled from local network.

The test case is flagged as MISSED if exploitation was successful and the test machine had been successfully controlled via the new session.
In case the exploitation was successful, as a proof of a working session, the following steps are taken.

- A screenshot has been made
- A file has been downloaded
- A file has been uploaded

The test case is flagged as MISSED if exploitation was successful and the test machine had been successfully controlled via the new session.
References:

https://ired.team/offensive-security/code-injection-process-injection/ntcreatesection-+-ntmapviewofsection-code-injection



Methodology used in the Real Botnet Test

1. Windows 10 Enterprise 64-bit operating system is installed on a hardened virtual machine, all updates are applied, and third-party applications installed and updated.
2. An image of the operating system is created.
3. A Real botnet dropper is run on the clean, unprotected system, thus simulating a pre-infected state.
4. A clone of the imaged system is made for each of the security applications to be used in the test.
5. An individual security application is installed using default settings on each of the systems created in (4) and then, where applicable, updated. If the vendor provided a non-default setting, this setting is checked whether it is realistic. If yes, the changes are documented, applied, and added in the report in an appendix.
6. A clone of the system as at the end of (5) is created.

Real botnet test is conducted by the following procedure.

6.1. Starting a new instance of Firefox (or the Safe Browser) and navigating to a financial website. Where the security application offers a secured or dedicated banking browser, this is used. If the security application is designed to protect Internet Explorer, only that component is tested.

6.2. Text is entered into the Account login page of the financial website using the keyboard or using a virtual keyboard if the application under test provides such functionality, and then the "log in" button is pressed.

- **The test case is marked as passed – a green checkmark** if the security application detects the financial malware when the security application is installed, and a mandatory scan is made. Or the security application detects the real financial malware when it is executed according to the following criteria:

  - It identifies the real financial malware as being malicious and either automatically blocks it or postpones its execution, warns the user that the file is malicious and awaits user input.
  - It identifies the real financial malware as suspicious or unknown and gives the option to run in a sandbox or safe restricted mode, which prevents the real financial malware from capturing and sending the logon data to the MRG CnC, whilst giving no alerts or giving informational alerts only. Or The security application intercepts the action of the real financial malware and displays warnings and user action input requests that are clearly different from those displayed in response to legitimate applications.

    a. **The test case is marked as missed – a red cross** if the security application fails to detect the real financial malware according to the following criteria:

- The security application fails to prevent the real financial malware from capturing and sending the logon data to the MRG CnC and gives no alert or provides informational alerts only.
- The security application intercepts the action of the real financial malware but displays warnings and user action input requests that are indistinguishable in meaning from those displayed in response to legitimate applications.
- The security application identifies the malware and gives the option to run in a sandbox or safe restricted mode which fails to prevent the real financial malware from capturing and sending the logon data to the MRG CnC and gives no alert or provides informational alerts only.

7. Testing is conducted with all systems having internet access.

Because we did not use 0-day malware in this test, but 1-2 years old or even older malware versions, when a security application provided both traditional AV engines and safe browser solutions, the security application was tested in two modes. In the first mode, all protections were turned on and the safe browser was used. In the second mode, all protections were turned on and the safe browser was not used. Thus, the second test simulated that if the user forgot to use the safe browser, but the AV engine is still on.

Methodology Used in the Financial Malware Simulator Test

1. Windows 10 Enterprise 64-bit operating system is installed on a hardened virtual machine, all updates are applied, and third-party applications installed and updated.
2. An image of the operating system is created.
3. A clone of the imaged systems is made for each of the security applications used in the test.
4. An individual security application is installed using default settings on each of the systems created in (3) and then, where applicable, updated. If the vendor provided a non-default setting, this setting is checked whether it is realistic. If yes, the changes are documented, applied, and added to the appendix section of the report.
5. A clone of the system as at the end of (4) is created.

Financial malware simulator test is conducted by the following procedure.

6. Where the security application offers a secured or dedicated banking browser, this is used. If the security application is designed to protect IE, only that component is tested.

   6.1. The simulator specific process is started.

- **The test case is marked as passed – a green checkmark** if the security application identifies the simulator as being malicious and either automatically blocks it or postpones its execution, warns the user that the file is malicious and awaits user input. Or, it identifies the simulator as suspicious or unknown and gives the option to run in a sandbox or safe restricted mode which does not allow the hooking/redirection, or even with successful hooking, the personal data cannot be captured from the browser.
- **The test case is marked as missed – a red cross** if the security application fails to identify the simulator based on the following criteria:

  - The security application allows the hooking/redirection of the event, and the personal data can be captured from the browser. Or, it fails to prevent the simulator from injecting itself into the browser process and gives no alert or provides informational alerts only.
  - The security application identifies the simulator as malware or unknown and gives the option to run in a sandbox or safe restricted mode which fails to prevent the simulator from injecting itself into the browser process and gives no alert or provides informational alerts only. Or, the security application allows the hooking/redirection of the event, and the personal data can be captured from the browser.

7. Testing is conducted with all systems having internet access.

Methodology used in performance test

1. Windows 10 Enterprise 64-bit operating system is installed on a physical machine, all updates are applied, and third-party applications installed and updated.
2. A backup image of the operating system is created.
3. The security application is installed, with the same configuration it is used in the other tests.
4. The following performance metrics are measured:

- Operating system boot time
- Size of the files installed and created by the security application. The size is measured at least one week after the installation, after virus definition updates, scans, and time passed with normal computer usage.
- Copy time of files
- Archive operation time
- Opening time for (clean) files in Office applications
- Downloading files through browser

- Website loading time in browser. The browser should fully load a popular, complex website, from a local network URL or replay proxy to eliminate network latency.
- AV product update time
- System disk scan time

Every performance result is a calculated average of at least three measurements.

Performance chart was calculated based on:

- The security product reaching the best result in the category was rewarded with 9 points, the second received 8 points and so on. Once every performance category was measured, the points were summed, and the final calculation was made by dividing the summarized points by the number of tests the product's result could have been measured.

Physical machine specification

- OS: Windows 10 x64
- CPU: Intel Core i5
- Memory: 8GB
- Storage: 100GB SSD

Hardened virtual machine specification

- OS: Windows 10 x64
- CPU: 2 core processor
- Memory: 4GB
- Storage: 100GB SSD

Appendix 2

Non-default endpoint protection configurations

Endpoint protection software was running on custom configuration if suggested by the vendor.

- **Avast Business Antivirus**
  Detailed logging was enabled via configuration file

- **Avira Antivirus Pro**
  Log level was set to 'Complete' instead of 'Default' in 'System Scanner' and in 'Real-Time Protection'

- **ESET Endpoint Security**
  Detection of 'Potentially unwanted applications' and 'Potentially unsafe applications' were turned on among with 'SSL/TLS protocol filtering'.

- **Microsoft Windows Defender**
  Microsoft Defender ATP endpoint detection and response capabilities were turned on including ASR rules.

- **Sophos Intercept X**
  Tamper protection was turned off

Default endpoint protection configurations

- **Bitdefender Endpoint Security**
- **F-Secure Computer Protection Premium**
- **Malwarebytes Endpoint Protection**
- **Symantec Endpoint Protection**
- **Trend Micro Security**

Version History

Nr.	Modify date	Comment
1.0	01.03.2021	Report published
1.1	09.03.2021	PUA chart and ‘Appendix 2’ updated