Online Banking / Browser Security Certification

Q1 2019

MRG Effitas
EFFICACY ASSESSMENT & ASSURANCE

+44 (0)20 3239 9289
www.mrg-effitas.com
@mrgeffitas

MRG Effitas Ltd.

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

MANAGEMENT TEAM:

- Chris Pickard, Chief Executive Officer
- Zoltan Balazs, Chief Technical Officer
- Rehab Frimpong, Chief Finance Officer

TESTING TEAM:

- Norbert Biro, Senior Threat Researcher

WEBSITE:
www.mrg-effitas.com

TEL:
+44 (0)20 3239 9289

EMAIL:
contact@mrg-effitas.com

TWITTER:
@mrgeffitas

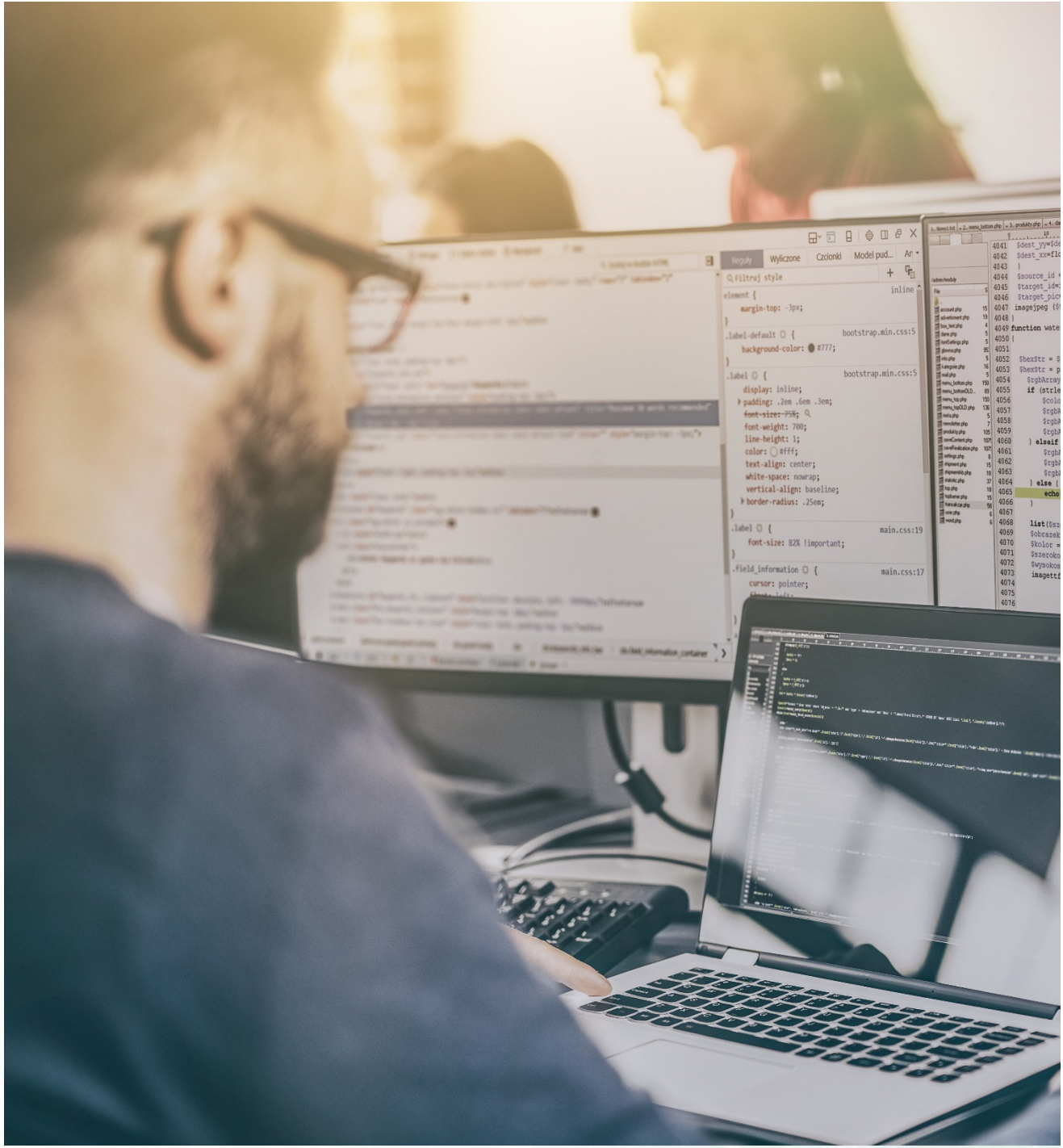
Contents

Introduction

MRG Effitas is a world-leader in independent IT security efficacy testing, research and expertise.

In the drive to protect businesses and home users from ever more advanced malicious threats, malware and viruses, our innovative research and testing helps IT security vendors to be the best they can be.

Our technical competence and insight into future trends and challenges is trusted by IT security vendors across the world.



About our Online Banking test

MRG Effitas publishes an Online Banking Browser Security Report every year. Since 2013, a single report has been replaced by quarterly assessments. This report is the assessment for Q1 2019.

While similar to our previous reports, it employs more sophisticated assessments that result in an extremely accurate level of efficacy assessments, so much so that we now award quarterly certifications to products that meet specific assessment criteria.

We provide two levels of testing: Level 1, where we test a vendor's product and provide a report for that quarter's assessment, and Level 2 (which incorporates Level 1), where we liaise with vendors during testing, alerting them to any issues found with their technology and providing the engineering and technical support required for them to counter these issues. Level 2 participation serves as an external QA service for vendors, helping them improve the efficacy of their product. Level 1 and 2 reports are published separately. This is a Level 1 report.

Early in our online banking work we recognised that although many vendors protect their clients' browsers from data exfiltration, the techniques employed were not effective against financial malware. Since then we have been at the forefront of online banking testing, and are the only testing house in the world whose tests map 100% against in-the-wild threats.

In this test we focus on in-the-wild financial malware, using cloud-based testing systems to create botnets that map identically to those we find in the real world. It is a criminal offence to test in-the-wild botnets in the UK, so we use IBM technology to host malware in a safe environment. We can create from scratch our own financial malware, reverse-engineer existing threats and modify them slightly.

Our tests comprise existing malware and real botnets. We can assess whether protective software detects existing malware and whether data exfiltration occurs against the browsers. We can anticipate future threats and advise our clients accordingly.

In 2010 we began reverse engineering financial malware to create simulators that employ the same "Man in the Browser" attacks as the in-the-wild code, and for the first time were able to determine whether secure browsers were capable of preventing data exfiltration.

Simulators are used in industries including aerospace, automotive, law enforcement, the military and finance. There are two major types of simulators: those used to teach students (e.g. pilots) and those used to simulate attacks (e.g. military). This is why we decided to start creating simulators: by developing test tools, we simulate attacks that may not be prevalent now, but could become more so in the future.

Simulators can point out potential weaknesses in products and even use new types of attacks that can be useful for developers as they can learn about these from a testing lab, rather than from their users when an attack of this type occurs in the wild.

Executive Summary

This Certification Programme can also serve as educational material for average users as it raises awareness about financial malware and all the dangers that face users when they do online banking, use online payment services, or use any form of online shopping.

It should be noted that financial malware earned its name because, in most cases, it attempts to grab the user name and password from places that are used for online transactions. Another thing financial malware can do is steal login credentials from popular social networking websites such as Facebook, Twitter, LinkedIn, etc.

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 financial malware works, how it attacks 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 malicious action should be blocked.

We tested a group of internet security suits and anti-financial fraud applications. With internet security suites it is very important to note that the best choice for an average user is to keep things very simple and for the product not to present many popup alerts or questions. Out of nine products we tested, four managed to pass all two stages of the test (simulators are not part of the certification). From these four products, none of them passed all tests, including the simulator tests.

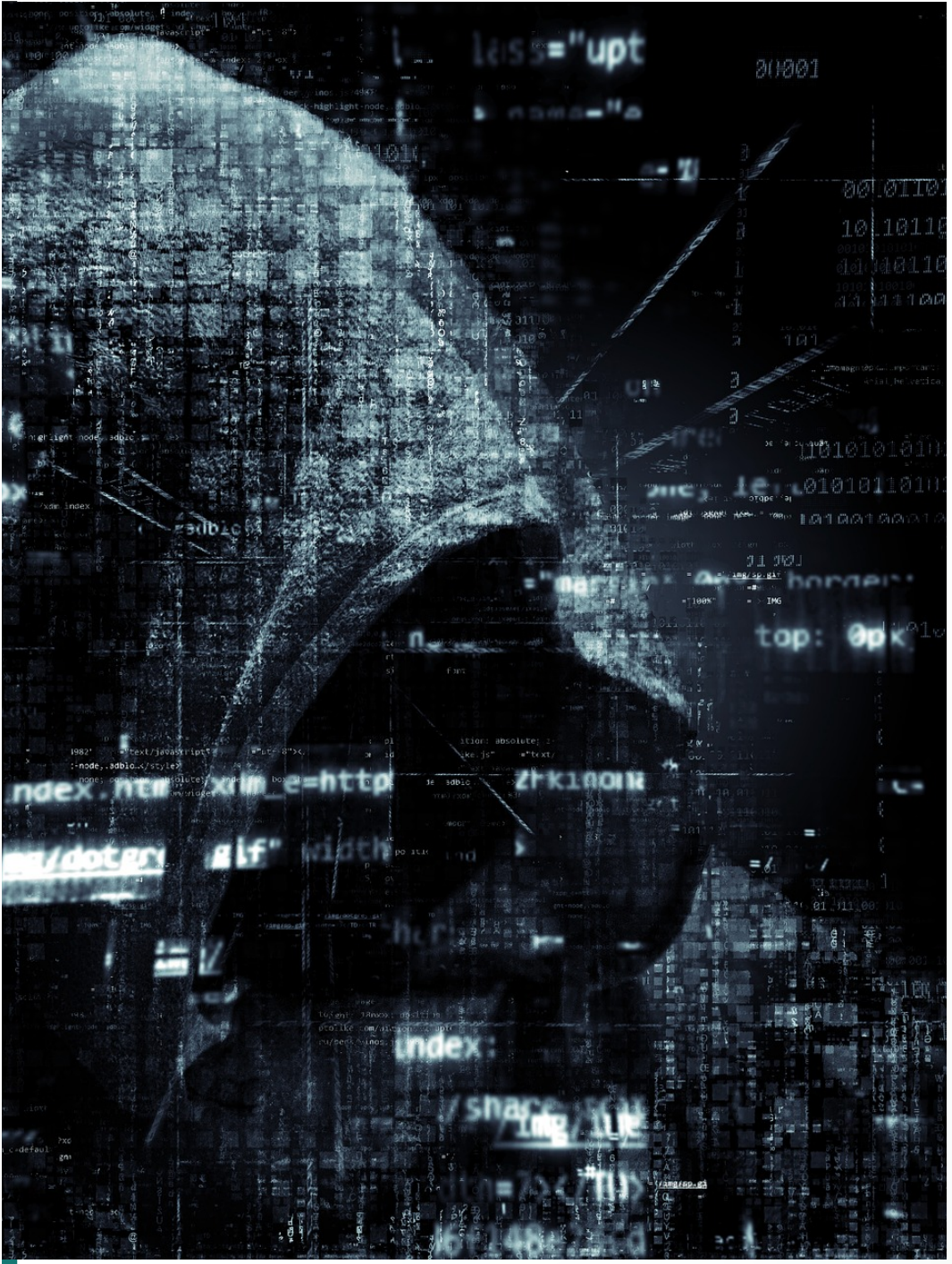
Certification

In order to attain MRG Online Banking / Browser Security Certification, a product must pass the ITW and Botnet test during the quarter. Applications that meet this specification will be given certification for that quarter.

Certified

The MRG Effitas Online Banking Browser Security Certification for Q1 2019 is awarded to the following products:

- **Avira Antivirus Pro**
- **Bitdefender Internet Security**
- **ESET Internet Security**
- **Kaspersky Internet Security**



The purpose of this report

In providing these quarterly certifications, the MRG Effitas Online Banking / Browser Security 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 current financial malware attacks.

We test over twelve months beginning in Quarter 2 and ending in Quarter 1, at which point (or shortly after) we publish our results. As with all of our certification testing, we work with vendors, offering feedback and helping them to improve their product as we go.

Products that pass the botnet and the in-the-wild real financial malware tests during a quarter will receive the MRG Effitas certification for secure online banking.

Tests Employed

In this assessment (Q1 2019) we ran the following tests:

In-the-Wild Real Financial Malware Test

In total, 15 live ITW samples were used. The tests were performed using financial malware only, including, inter alia, the following: ZeuSv1/v2 clones, Azorult, Emotet, Trickbot, IceDid, Ursnif, Gootkit, etc.

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 Magecart credit card-skimming attack

Magecart is the name of the collection of groups who are targeting some eCommerce sites and ticketing companies such as Ticketmaster, British Airways, Newegg, Infowars etc. They used a small JavaScript code on the online store's checkout pages. These scripts pulled personal and/or credit card data and sent it to the attackers' servers. Magecart groups use distinct approaches: directly compromise the eCommerce website's server or a third-party tool that they were using. The JavaScript code is usually obfuscated by some obfuscator tool. That's why we obfuscated our test sample for this test case.

Security Applications Tested

- avast! Internet Security 19.2.2364
- Avira Antivirus Pro 15.0.44.142
- BitDefender Internet Security 2019 23.0.19.85
- ESET Internet Security 12.1.31.0
- Kaspersky Internet Security 2018 19.0.0.1088 (d)
- McAfee Total Protection 16.0 R18
- Microsoft Windows Defender with SmartScreen 4.18.1902.2
- Symantec Norton Security 22.16.4.15
- Trend Micro Maximum Security 15.0.1212

Samples used in the In-the-Wild real financial malware test

Sample selection is of fundamental importance to this and all similar tests. In the case of the Online Banking / Browser Security Certification – In-the-Wild Test, all samples used are “live” and “in the wild”, by which we mean they are residing at the URLs selected or created by the cybercriminals and they are not from a time lagged ITW list. As these are live ITW samples, they represent current zero day threats that can present an issue with sample verification. There is no effective and reliable way to verify samples before testing that does not introduce possible artificial sample submission or delay, so all verification is conducted after testing. Tests performed using samples that are later proven to be invalid are excluded from the results. The type of samples used is decided by MRG Effitas on the basis of a mixture of criteria, cantering about key relevancies:

1. Prevalence – they are widespread and so represent the most common threats.
2. Growth – they may be few now, but our research shows they are rapidly expanding.
3. Innovation – they employ innovative techniques to counter security measures.
4. It is malware having financial motives, by either stealing login credentials, initiating transactions, or doing web injects.

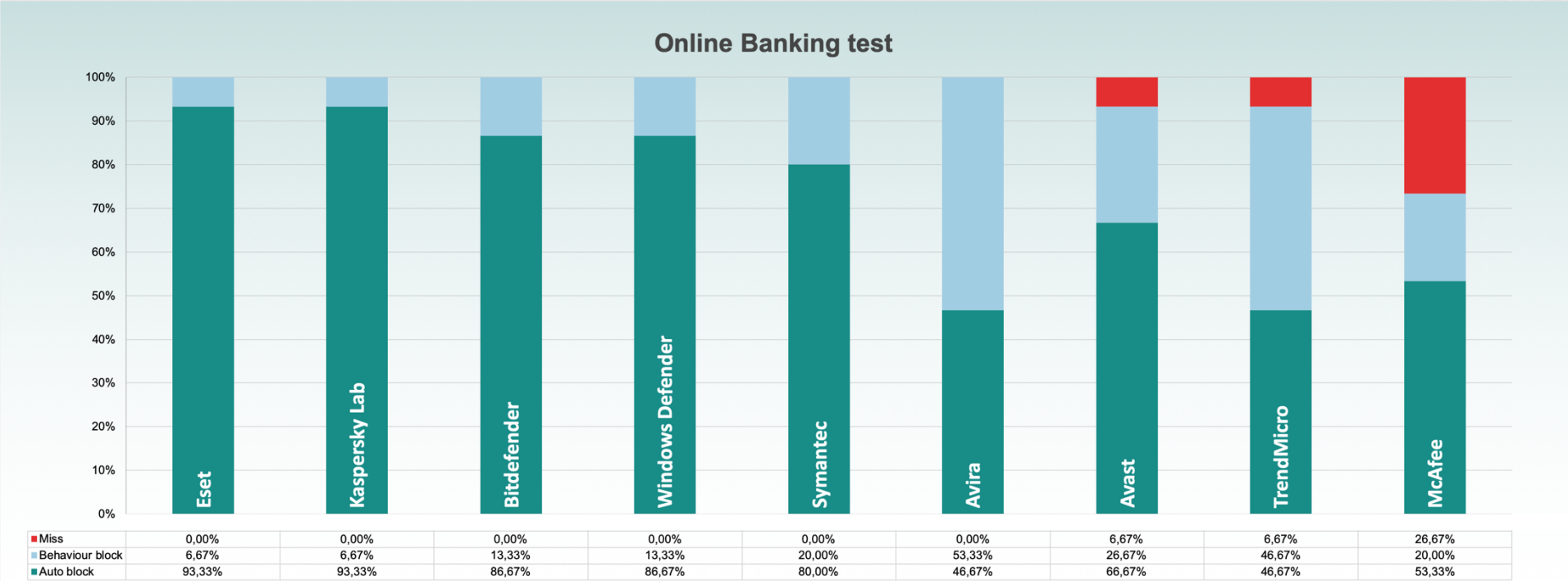
In total, 15 live ITW samples were used. The tests were conducted using financial malware only, including Azorult, Emotet, Trickbot, IceDid, Ursnif, Gootkit clones.

Test Results

The tables below show the results of testing in the Online Banking / Browser Security Certification Programme.

Q1 2019 In-the-Wild real financial malware test results

The table below shows the results of testing using In-the-Wild real financial malware.



	Eset	Kaspersky Lab	Bitdefender	Windows Defender	Symantec	Avira	Avast	TrendMicro	McAfee
Miss	0,00%	0,00%	0,00%	0,00%	0,00%	0,00%	6,67%	6,67%	26,67%
Behaviour block	6,67%	6,67%	13,33%	13,33%	20,00%	53,33%	26,67%	46,67%	20,00%
Auto block	93,33%	93,33%	86,67%	86,67%	80,00%	46,67%	66,67%	46,67%	53,33%

Q1 2019 Real Botnet test results

The table shows the results of testing using real financial malware.

Detailed description of the botnet test

For this test, a modified version of a plain TinyNuke sample was used. The dropping process was not simulated. A clean Windows 7 system was created, and the modified botnet was executed on this clean system which is simulated the pre-infected state of the machine. Whenever the OS starts the botnet was loading itself into the memory. We configured the C&C servers in the safe SoftLayer environment. Because this test used real financial malware, where data exfiltration can be tested as it happens in the wild, the test efficiently maps the real-world threats users face today. This builder and dropper are available to everyone for free, thus the threats provide an entry level for criminals and are common threats in the wild. The advantage of this test is that there is no TinyNuke malware on disc for a regular AV to catch, but a simple process memory scan can find and kill the malicious code.

Botnet test	
Vendor	TinyNuke
avast! Internet Security with Secure Browser	✓
avast! Internet Security without Secure Browser	✗
Avira Antivirus Pro	✓
BitDefender Internet Security 2019	✓
ESET Internet Security with banking protection	✓
ESET Internet Security without banking protection	✓
Kaspersky Internet Security with SafeMoney	✓
Kaspersky Internet Security without SafeMoney	✓*
McAfee TotalProtection	✗
Microsoft Windows Defender	✗
Symantec Norton Security	✗
Trend Micro Maximum Security with PayGuard	✓
Trend Micro Maximum Security without PayGuard	✗

✓ The application prevented the malware from capturing login data

✗ The application failed to prevent the malware from capturing login

* The application prevented the malware from capturing login data after the system restart

During the tests we witnessed many problems with endpoint protection systems. Following is a non-exhaustive list of problems:

- With some sample, we witnessed multiple times that although the AV was able to detect the malware at boot time, the quick scan after installation failed to detect the sample and because the AV did not enforce or recommend a reboot, the sample remained active.
- Inconsistent behaviour/block: Some vendors failed to protect the user in the first test but protected the user after the first test. During the first test, the protected browser usually crashed and was restarted automatically. If the user was protected 4 times from 5 attempts, we marked these as transient failures and the products were marked as having passed.
- Missing alert: Some vendors detected the threat during the security product installation but failed to warn the user about the detected and removed threat. However, the detailed AV log revealed the threat detection and removal.
- Missing log and alert: Some vendors detected the threat during the security product installation but failed to warn the user about the detected and removed threat, and even the detailed AV log was empty.
- Some vendors would have failed the test without the mandatory restart in the test methodology. These vendors had not suggested or enforced any restart after product installation or threat removal.
- Some vendors detected the threat and removed the malware from the file system, but the threat was not removed from the memory. After threat removal, the security product did not suggest any restart to the user. This was marked as a fail, as users tend to use the OS without restarting for weeks.
- Some safe browsers are using browser types that are not targeted by financial malware. As a result, even if the malware was running in the background and without any active protection, these browsers passed the test.
- A vendor detected some malware samples and gave the option to block the threat. Still, it did not prevent the malware from stealing login credentials.

Q1 2019 Simulator test results

The table shows the results of testing using the malware simulators.

Detailed description of the simulator test

The methodology behind these attacks was simple and similar: Injecting a malicious obfuscated JavaScript code into the website's checkout pages and listening for an event for example when the user clicks on the "pay" or "place order now" or similar button (event hijacking). When this event happens, the malicious code sends the credit card data to the attackers' servers. In our test we simulated this attack. We implemented our obfuscated malicious JavaScript code based on the Newegg and British Airways cases and injecting it into a test webstore which was built by us. The code behavior and the obfuscation technique are exactly same as in the real-world examples: when the user fills out the credit card (cc) data and press the "place order now" button the cc data is sent to our servers.

Simulator test	
Vendor	Obfuscated Magecart
avast! Internet Security with Secure Browser	✗
Avira Antivirus Pro	✗
BitDefender Internet Security 2019 with SafePay	✗
ESET Internet Security with banking protection	✗
Kaspersky Internet Security with SafeMoney	✗
McAfee TotalProtection	✗
Microsoft Windows Defender	✗
Symantec Norton Security	✗
Trend Micro Maximum Security with PayGuard	✗

✓ The application blocked the simulator

✗ The application failed to block the simulator

Appendices

A: Methodology Used in the Q1 2019 Online Banking Certification – In-the-Wild Test

1. Windows 10 PRO 64-bit operating system is installed on a virtual machine, all updates are applied, and third-party applications installed and updated according to our “Average Endpoint Specification”.
2. An image of the operating system is created.
3. A snapshot of the imaged systems is made for each of the security applications to be used in the test.
4. An individual security application is installed using default settings on each of the systems created in 2 and then, where applicable, it is updated and shut down. If the installer has the option to participate in cloud protection, or PUA protection, all of these are enabled.
5. Testing is conducted by:
    a. Downloading the sample using Chrome, the browser is kept running, conducting a context menu scan or, where unavailable, a system scan, and then executing the sample.
6. A test is deemed to have been passed based on the following criteria:
    a. The security application blocks the URL where the sample is located, thus preventing its download.
    b. The security application detects the sample whilst it is being downloaded.
    c. The security application detects the sample during the context or system scan.
    d. The security application detects the sample when it is executed according to the following criteria:
        i. It identifies the sample as being malicious and either automatically blocks it or pauses its execution, advises the user not to execute it and awaits user input.
    e. The security application blocks the malware from stealing the password from the browser.
7. A test is deemed to have been failed based on the following criterion:
    a. The security application fails to detect the sample under condition 6a, 6b, 6c, 6d or 6e, and the malware is able to steal passwords from the browser.
8. Testing is conducted with all systems having internet access.

B: Methodology Used in the Q1 2019 Online Banking Certification – Real Botnet Test

1. Windows 7 Ultimate 64-bit operating system is installed on a virtual machine, all updates are applied, and third-party applications installed and updated according to our “Average Endpoint Specification”.
2. An image of the operating system is created.
3. Real TinyNuke dropper is installed onto clean systems without protection, thus simulating a pre-infected state.
4. A clone of the imaged systems 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 2 and then, where applicable, it is updated and shut down if the security application asked it form the user. If the installer has the option to participate in cloud protection, or PUA protection, all of these are enabled.
6. A snapshot of the system as it is at the end of 5 is created, and the system is started.
7. Each real financial malware test is conducted by:
    a. 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 will be tested.
    b. 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.
8. A test is deemed to have been passed (marked as a green checkbox) based on the following criteria:
    a. The security application detects the real financial malware when the security application is installed, and a mandatory scan is made.
    b. The security application detects the real financial malware when it is executed according to the following criteria:
        i. 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.
        ii. It identifies the real financial malware as suspicious or unknown and gives the option to run in a sandbox or safe restricted mode, and, when run in this mode, it meets the criterion c or d below.
    c. The security application prevents the real financial malware from capturing and sending the logon data to the MRG results page, whilst giving no alerts or giving informational alerts only.
    d. 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, when they are executed or installed on that system.
9. A test is deemed to have been failed (marked as a red cross) based on the following criteria:
    a. a. The security application fails to detect the real financial malware and then:
        i. The security application fails to prevent the real financial malware from capturing and sending the logon data to the MRG results page location (malware C&C server) and gives no alert or provides informational alerts only.

   ii. 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, when they are executed or installed on that system.
b. The security application identifies the malware and gives the option to run in a sandbox or safe restricted mode, and, when run in this mode, it:
   i. Fails to prevent the real financial malware from capturing and sending the logon data to the MRG results page or local store and gives no alert or provides informational alerts only.
   ii. Displays warnings and user action input requests that are indistinguishable in meaning from those displayed in response to legitimate applications, when they are executed or installed on that system.

10. 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 engines is still on.

C: Methodology Used in the Q1 2019 Online Banking Certification – Simulator Test

1. Windows 10 PRO 64-bit operating system is installed on a virtual machine, all updates are applied, and third-party applications installed and updated according to our “Average Endpoint Specification”.
2. An image of the operating system is created.
3. A snapshot of the imaged systems is made for each of the security applications to be 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, it is updated. If restart is recommended by the application (visible to the user), the system is restarted. If the installer has the option to participate in cloud protection, or PUA protection, all of these are enabled.
5. A snapshot of the system as it is at the end of 4 is created, and the system is started.
6. The simulator is started onto the clean systems with protection installed.
7. Each simulator test is conducted by:
   a. Starting a new instance of Chrome (or the safe browser) and navigating to our compromised webstore. 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 going to be tested.
   b. Text is entered into the credit card data boxes (owner name, credit card number, card type, expiration date and cvc number) using the keyboard, or using a virtual keyboard if the application under test provides such functionality, and then the “Place Order Now” button was pressed.
8. A test is deemed to have been passed (marked as a green checkbox) based on the following criteria:
   a. The security application detects the malware simulator when it is executed according to the following criteria:

i. It 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.
ii. It identifies the simulator as suspicious or unknown and gives the option to run in a sandbox or safe restricted mode, and, when run in this mode, it meets the criterion c below.

b. The security application does not allow the hooking/redirection of the “Place Order Now” button, or even with successful hooking, the credit card data cannot be captured from the browser.

9. A test is deemed to have been failed (marked as a red x) based on the following criteria:
   a. The security application fails to detect the simulator and then:
      i. The security application fails to prevent the simulator from injecting itself into the browser process and gives no alert or provides informational alerts only.
      ii. The security application allows the hooking/redirection of the event, and the credit card data can be captured from the browser.
   b. The security application identifies the simulator as malware or unknown and gives the option to run in a sandbox or safe restricted mode, and, when run in this mode, it:
      i. Fails to prevent the simulator from injecting itself into the browser process and gives no alert or provides informational alerts only.
      ii. The security application allows the hooking/redirection of the event, and the credit card data can be captured from the browser.
10. Testing is conducted with all systems having internet access.

Disclaimer

1. The information contained in this report is subject to change and revision by MRG Effitas without notice. Furthermore, MRG Effitas is under no obligation to update this report at any time.
2. MRG Effitas has made every reasonable effort to ensure the information contained within this report is accurate at the time of its publication but cannot guarantee this.
3. Use of any information contained within this report is at your own risk. MRG Effitas shall not be liable or responsible for any loss of profit, good¬will, reputation or data, or for any damages arising from this report in any way whatsoever.
4. This report represents the results of tests carried out in good faith under professional and unbiased conditions. It does not constitute a guarantee (or otherwise) of any of the products listed, mentioned or tested.
5. Our testing and subsequent reporting does not guarantee that there are no errors in the products featured, nor that featured products will meet your expectations or requirements.
6. Any vendor trademarks/names/logo/images used in this report belong to their respective vendors.