Keywords: anti-malware; compliance; assessment; testing; test plan; MRG Effitas; 360 Degree Assessment

AMTSO 2018
August 2, 2018

Version 1.2

MRG Effitas Test Plan for Q3 2018 360 Degree Assessment and Certification

Sponsored and Authored by:
MRG Effitas (Zoltan Balazs), AMTSO (John Hawes, Scott Jeffreys)

Abstract:
This Test Plan has been prepared jointly by MRG Effitas and AMTSO as part of the AMTSO Standards V1.0. The Plan details the MRG Effitas testing activities in 360 Degree Assessment and Certification for the period July through September 2018 with reporting taking place in October 2018. This document has been developed using AMTSO Test Plan Template Version 2.1 from June 2018. Wherever conflicts might exist between this Template and the Live Standards Version 1.0, the Testing Protocol Standards will provide the prevailing rule.

www.amtso.org

Table of Contents

MRG Effitas 360 Degree Assessment and Certification Test Plan – Q2'2018

1. Introduction

A first-of-its-kind test that covers all angles, our pioneering 360 Degree Protection Test targets the key threats faced by internet users. In each test case we employ the full spectrum of Early Life Malware. We use a Time-To-Detect metric to measure how long it takes each application to detect and neutralize missed threats.

MRG Effitas has a core focus on efficacy assessments in the anti-financial fraud space, but we also publish more traditional "Real World" detection tests. Our "Time to Detect Assessment Q4 2013" measured the ability of security Test Subjects to protect an endpoint from a live infection, and, in the event of a system being compromised, the time taken to detect the infection and remediate the system. The time-to-detect-and-remediate component relied on each security product being manually forced to conduct a scan every thirty minutes over a 24-hour period. For 2014, it was decided that a new approach was needed as the methodology applied in previous tests did not reflect how a security product would be used on an endpoint in the Real World.

In practice, many security applications will only detect an infection during a reboot/startup or if a scheduled scan has been set by default. For this assessment, time-to-detect will employ a methodology based on the infected endpoint being re-scanned once during a 24-hour period. The methodology employed in this test maps more closely to Real World use, and although it may not be a 100% accurate model of how an "average" system is used, it gives a more realistic assessment of a security product's ability to detect and remediate an infected endpoint.

This Programme is called a "360 Assessment" since it deals with the full spectrum of malware instead of just financial malware. In the 360 Assessments, trojans, backdoors, ransomware, financial malware and "other" malware are used.

2. Scope

This Certification Programme is designed to serve as a reflection of product efficacy based on what we have previously termed "metrics that matter". In many of our previous tests, particularly those that have focused on financial malware, we started with the assumption that the endpoint has already been compromised. Being the world's largest supplier of early-life malicious binaries and malicious URLs, and from our own simulator development, we know that all endpoints can be infected, regardless of the security solution employed.

For us, a product's ability to block initial infection (although critical in most cases) is not the only metric that matters. One also needs to measure the time taken for the security product to detect malware on a system and remediate it. 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 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 suites and complementary security applications. With these, 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 pop-up alerts or questions.

In our next execution of this Test Plan, eleven Test Subjects were considered. Companies expected to be represented and their target Test Subjects include the following. Specific Test Subject Vendors and Participants will be determined after the Public Test Notification has been issued.

- avast! Business Antivirus
- Avira Antivirus Pro - Business edition
- BitDefender Gravityzone Advanced Business Security - cloud
- ESET Endpoint Security
- F-Secure Protection Service for Business
- Kaspersky Endpoint Security Cloud
- Microsoft Windows Defender
- McAfee Endpoint Security
- Symantec Endpoint Protection
- Trend Micro Worry-Free™ Services with XGEN
- Webroot SecureAnywhere Business Endpoint Protection

3. Methodology and Strategy

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 centred on measuring a product’s ability to detect malware.

Testing has evolved rapidly over the last two to three years as most labs, under the guidance of AMTSO (of which MRG Effitas is a member) strived to conduct “Real World” testing. 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 vector, such as a browser or 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). Several testing labs also conduct “System Rescue” tests. These assess a security product’s ability to remediate a preinfected endpoint.

Whilst both types of tests are useful and yield valid and meaningful data, MRG Effitas wanted to merge these tests and also go one step further by measuring the time security Test Subjects 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 re-scanned once a day (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, all 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 have their malware on a system, the greater the opportunity for them to be able to capture private user information including banking passwords and social media credentials, etc.

There has been an increase in the prevalence of ransomware, which, once active on the system, holds the user at ransom to decrypt system data or unlock the system in some other way. For these types of malware, it is initial detection that is of the greatest importance, since the vast majority of security solutions will be unable to rescue an encrypted or locked system.

In providing these quarterly certifications, the MRG Effitas 360 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.

The detailed Methodology to be used in the Real World Protection assessment adheres to the following procedures.

1. Windows 10 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 will be 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 will be 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 will be checked whether it is realistic. If yes, the changes will be documented, applied, and added in the report in an appendix.

5. A clone of the system as at the end of (4) will be created.

6. Each live URL test is conducted by the following procedure.

    a. Downloading a single binary executable (or document, script, etc.) from its native URL using Microsoft Edge to the Downloads folder and then executing the binary.

    b. Either the security application blocked the URL where the malicious binary was located.

        i. Or the security application detected and blocked the malicious binary whilst it was being downloaded to the desktop.

        ii. Or the security application detected the malicious binary when it was executed according to the following criteria: It identified the binary as being malicious and either automatically blocked it or postponed its execution and warned the user that the file was malicious and awaited user input.

7. The system under test is deemed to have been infected if the security application fails to detect or block the binary at any stage in (6) and allowed it to be executed.

8. The test case was retested 24 hours after the initial test if the security application failed to detect or block the malicious binary.

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

10. As no user-initiated scans will be involved in this test, applications relies on various technologies to detect, block and remediate threats. Some of these technologies were: URL blacklist, reputation, signature, machine learning, heuristics, behavior etc.

The same methodology is used for fileless attacks, but instead of the malicious binary, some other form of infection vector is used.

4. Exploit test methodology

The test will be conducted as follows:

1. One default install Windows 10 hardened virtual machine endpoint is created. The default HTTP/HTTPS proxy is configured to point to a proxy running on a different machine. SSL/TLS traffic is intercepted on the proxy, and AV's have been either configured to skip the proxy, or the SSL decryption is disabled for the AV's update/cloud connections.
2. The security of the OS is weakened by the following actions:
    a. Microsoft Defender is disabled
    b. Internet Explorer SmartScreen is disabled
3. The following vulnerable software is installed:
    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 11
    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 is disabled.
5. From this point, XXX different snapshots are created from the virtual machine, each with different endpoint protection products and one with none. This procedure ensures that the base system is exactly the same in all test systems. The following endpoint security suites, with the following configuration, are defined for this test:
    a. No additional protection, this snapshot is used to infect the OS and to verify the exploit replay.

b. Vendor A
c. Vendor B
d. …

The endpoint systems are installed with default configuration, potentially unwanted software removal is enabled, and if it was an option during install, cloud/community participation is enabled. The management servers (if needed) are 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 are used during the test. One in-the-wild exploit kits (e.g. RIG), and one from publicly available open-source exploit frameworks (e.g. Metasploit). In spite of other "real world protection tests", no binary downloads (e.g. exe) were tested. ActiveX, VBscript based downloaders and Office macro documents are out of scope. Microsoft Office documents with exploits inside are in scope of the test.
7. The virtual machine is reverted to a clean state and traffic was replayed by the proxy server. The replay meant that the browser is used as before, but instead of the original webservers, the proxy server answers the requests based on the recorded traffic. In this replay, no other traffic is allowed, which means that unmatched requests (previously not recorded) were answered with HTTP 404 codes. When the "replayed exploit" is able to infect the OS, the exploit traffic is 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 is contacted, without replaying.
8. After new exploit traffic is approved, the endpoint protection systems are tested, in a random order. Before the exploit site is tested, it is verified that the endpoint protection had been updated to the latest version with the latest signatures and that every cloud connection is working. If there is a need to restart the system, it is restarted. In the proxy setup, unmatched requests are allowed to pass through. No VPN is used during the test. When user interaction is needed from the endpoint protection (e.g. site visit not recommended, etc.), the block/deny action is chosen. When user interaction is needed from Windows, we chose the run/allow options, except for UAC. No other processes are running on the system, except the Process Monitor from Sysinternals and Wireshark (both installed to non-default directories and modified not to be detected by default tools).
9. After navigating to the exploit site, the system is monitored to check for new processes, loaded DLLs or C&C traffic.
10. After an endpoint protection suite is tested, a new endpoint protection is randomly selected for the test until all endpoint protection products had been tested.
11. The process goes back to step 7. until all exploit site test cases are reached.

4.1. Analysis of the exploit results

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”, 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.

5. False positive test methodology

1. Windows 10 64-bit operating system is installed on a virtual machine, all updates are applied and third party applications installed and updated.

2. An image of the operating system will be 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 will be installed using same configuration as in the other tests on each of the systems created in (3) and then, where applicable, updated.

5. A clone of the system as at the end of (4) will be created.

6. Each FP test is conducted by the following procedure.

    a. Downloading a single binary executable (or document, script, etc.) from an internal URL to the Downloads folder

    b. Executing the binary.

7. The test case is marked as a False Positive sample if the security application detects or blocks the binary at any stage in (6).

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

6. Performance test methodology

1. Windows 10 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 will be created.

3. A security application is installed into the OS. Same configuration is used as in the other tests.

4. The following performance metrics are measured:

    a. Install time, starting from downloading the installer binary, finished when the security application is installed, started, and the GUI is working.

    b. Size of the files installed and created by the security application. The size is measured both after the installation, and after some time passed with normal computer usage.

    c. CPU overhead of the processes and services belonging to the security applications are summed.

d. Memory footprint (private and shared working set) of the processes and services belonging to the security applications are summed.

e. The performance impact on the browser load time is measured. The browser should fully load a complex website, from a local network URL or replay proxy.

7. Participation

AMTSO's goal with having Participants is that in exchange for cooperating (engaging with Testers and following disclosure requirements), Participants have additional rights to audit their configuration and provide commentary on Test results. There must be no additional cost to a Test Subject Vendor to be a Participant. If a Tester charges to participate in a Public Test or any related services, and a Test Subject Vendor chooses to not pay the fee, that Vendor must be able to choose to be a Participant and follow this AMTSO standard.

Opt-Out Policy : Vendors can opt out if Vendor can prove that the test system or the Product was misconfigured in a way which greatly changes the test results. E.g. important modules were unnecessarily turned off compared to default configuration, or during the test the system could never reach the cloud.

Conflict of Interest Disclosure : There is no known conflict of interest.

Funding : Funding of this project is achieved by vendors subscribing to participate in this project, this gives them more in-depth information on how their product(s) performed and, if there are any issues discovered in the product during testing, our technical team provides all the data necessary to help improve the product.

Part of the funding comes from directly licensing reports so they can be used for marketing purposes.

Vendors often silently enter testing, sometimes of their newly developed product or a product in BETA or Pre-Release phase.

Finally, part of our funding comes from third parties, they commission us to include certain product into testing, both public and private.

8. Environment

Test hardware and configuration details follow.

Physical Configuration : Our Virtual Machine hardware specification calls for 4GB RAM and a dual core processor. AES includes Adobe Flash, Reader, Java, Microsoft Office 2010 or newer, Edge & VLC Player. All Microsoft components were fully updated and all third-party components were out of date by three months. During installation of the security application, if an option to detect PUAs was given, it was selected.

Sample Relevance : In the Wild 360 / Full Spectrum Test, approximately 50% of the malicious URLs used in this test were compromised legitimate websites which served malware. We believe that such URLs pose the greatest danger to users as this is the place where they least expect to

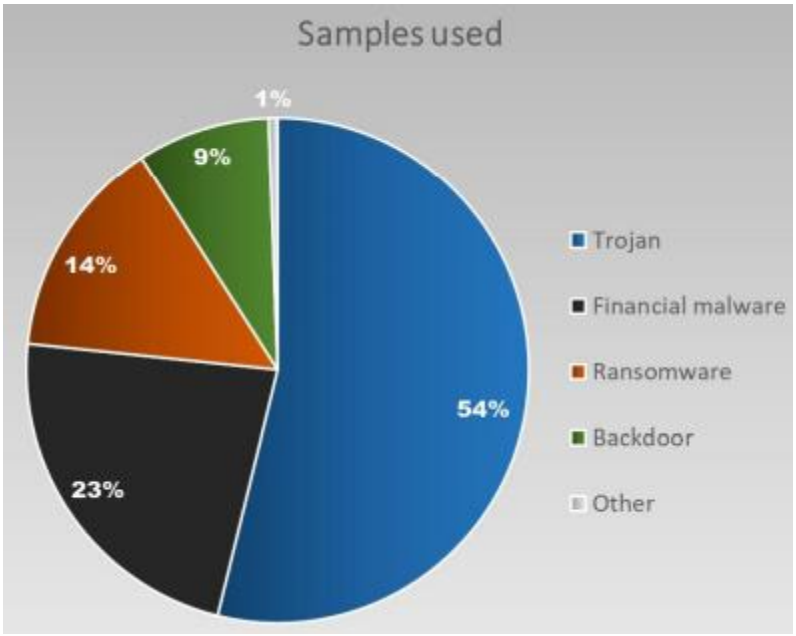
get infected. 10% of the URLs pose as fake porn websites serving visitors with various types of malware. The remaining 40% of the URLs come from our regular honeypots or, in case of ransomware and financial malware in particular, we used URLs from newly-discovered distribution sites.

Geographic Limitations : There are no geographic limitations in terms of samples.

Curation Process : Voluntary Participants are given equal opportunities to participate in such Curation and feedback processes for all their respective Test Subjects.

Malware delivered by URLs used in this test can be considered as Zero Day in the true meaning of that phrase. It is our opinion that Ransomware currently poses the greatest threat to users, for this reason we choose to use more URLs serving this threat than before. Because of the wide spectrum of malware used in this project and the freshness of the samples, we used a smaller set than usual.

Applications that didn't protect the system from file encrypting ransomware cannot be certified because they could not remediate the threat as files usually cannot be decrypted. Our testing environment supports the use of VM aware malware, this is the reason why we were able to use more sophisticated threats which wouldn't run on Virtual Machines. 10% of the threats used in this test were introduced to the system via USB flash memory sticks. These samples came originally from live URLs, but inside archives. Testing was conducted as per the methodology detailed in Appendix 1. In total, 351 live ITW samples were used. The stimulus load comprised the following: 189 trojans, 30 backdoors, 80 financial malware samples, 50 ransomware samples, and 2 others.



Distribution of Test Data : We send all failed samples to all participants, along with detailed test logs.

9. Schedule

Start Date Range : The test commencement date is August 10, 2018.

Test Duration and Calculated End Date : The test is expected to require approximately seven weeks and is forecast to conclude on October 15, 2018.

Milestones : Delivery milestones appear in the following chart.

MRG-Effitas 360 Degree Test Project Schedule Milestones

Index	Test Activity	Start Date Range	Dependencies
1	*Test Commencement*	*August 10, 2018, Duration until October 1, 2018*	
2	*Confirm Vendor Configuration Feedback*	*August 16, 2018*	
3	*Milestone 1 – Preliminary Results*	*September 17, 2018*	*(1), (2)*
4	*Milestone 2 – Test Report First Edition – End of Testing Period*	*September 24, 2018*	*(3)*
5	*Feedback and Dispute Resolution Time – Retests as Needed*	*September 24, 2018 - October 1, 2018*	*(4)*
6	*Milestone 3 – Issue Final Report – End Date for Test*	*October 15, 2018*	*(5)*

Communications : Whenever there are significant deviations from this schedule (more than 5 days), we will notify all affected vendors within 3 business days.

Risks and Risk Management : We could not identify any risks with the test.

10. Control Procedures

Connectivity Validation : Tests are conducted with all systems having internet access. Each individual test for each security application is conducted from a unique IP address. All security applications are fully-functional unregistered versions or versions registered anonymously with no connection to MRG Effitas. Each vendor can supply a utility or another in-product feature to validate proper cloud-connectivity functionality.

Logging : In the first initial email to the vendors, we will ask if they require any special logging during the test. We expect these methods and tools to be working, otherwise we cannot guarantee the result of these logs.

Updates : Each individual security application will be installed using default settings on each of the test systems defined in the Environment section of this Test Plans and then, where applicable, updated.

11. Dependencies

Participant Actions : As this test scope are consumer Test Subjects, we do not require any Vendor specific actions.

12. Scoring Process

In order to attain a quarterly MRG Effitas 360 Degree certification award, a security application must either protect the system from initial infection (autoblock or behaviour protection) (a level 1 pass) or in at least 98% of all cases detect any missed malware and fully remediate the system before or on the first user initiated rescan (a level 2 pass).

Applications that meet this specification will be given certification for that quarter. Under the MRG Effitas 360 Degree Assessment & Certification, Test Subjects are certified for Level 1 or Level 2 Compliance.

13. Dispute Process

After the vendors receive the test results and the logs, Vendors can dispute individual test cases if they can prove that they do not agree with the result.

14. Attestations

I understand and agree that I am submitting this Test Plan, and the following Attestations, on behalf of the entity listed below, and I represent and warrant that I have authority to bind such entity to these Attestations. All references to “I” or “me” or similar language refer to such entity. I represent and warrant that the following Attestations are true, to the best of my knowledge and belief, and each of the following commitments will be upheld to the best of my ability.

1. I will provide public notification on the AMTSO website covering my obligation for notification of a Public Test, regardless of whether a potential Participant is in actual receipt of such notification prior to the Commencement Date of a Test. (Section 1, Section 4, Section 6)
2. All Test Subjects included in this Test will be analyzed fairly and equally. (Section 2, Section 3, Section 5)
3. I will disclose any anticipated or known imbalance or inequity in the Test design to all Participants in the Test. (Section 2, Section 3)
4. Although I may charge for participation in a Test, I will not charge any additional fees for a Test participant to be “Voluntary” under the Standards. (Section 4)

5. I will disclose any material conflicts of interest or other information that could materially impact the reliability of the Test. (Section 4)
6. I will disclose how the Test was funded. (Section 4)

I hereby affirm, to the best of my knowledge and belief that this Test Plan complies with the AMTSO Testing Standards, as of the date hereof.

Signature: /s/ Zoltan Balazs

Name: Zoltan Balazs

Test Lab: MRG Effitas

AMTSO Test ID: [AMTSO-LS1-TP003]