Guidelines for Testing Protection Against Targeted Attacks

amtso™

Anti-Malware Testing Standards Organization

`

Notice and Disclaimer of Liability Concerning the Use of AMTSO Documents

This document is published with the understanding that AMTSO members are supplying this information for general educational purposes only. No professional engineering or any other professional services or advice is being offered hereby. Therefore, you must use your own skill and judgment when reviewing this document and not solely rely on the information provided herein.

AMTSO believes that the information in this document is accurate as of the date of publication although it has not verified its accuracy or determined if there are any errors. Further, such information is subject to change without notice and AMTSO is under no obligation to provide any updates or corrections.

You understand and agree that this document is provided to you exclusively on an as-is basis without any representations or warranties of any kind whether express, implied or statutory. Without limiting the foregoing, AMTSO expressly disclaims all warranties of merchantability, non-infringement, continuous operation, completeness, quality, accuracy and fitness for a particular purpose.

In no event shall AMTSO be liable for any damages or losses of any kind (including, without limitation, any lost profits, lost data or business interruption) arising directly or indirectly out of any use of this document including, without limitation, any direct, indirect, special, incidental, consequential, exemplary and punitive damages regardless of whether any person or entity was advised of the possibility of such damages.

This document is protected by AMTSO's intellectual property rights and may be additionally protected by the intellectual property rights of others.

`

Guidelines for Testing Protection Against Targeted Attacks

Introduction

This document describes best practices for the testing of solutions which claim to detect or protect against sophisticated targeted attacks. Such attacks represent a major threat to corporate and governmental systems, networks and data, and a wide range of new technologies has been developed to mitigate this risk. Testing the quality and accuracy of these new technologies is vital to ensuring that the right tools and solutions are deployed where they are most required. However, the complexity and diversity of "advanced" threats from targeted attacks, and the techniques used to target specific companies or institutions, render testing such solutions significantly more difficult than testing security software aimed at a more general market and protecting against more general threats. Comparative testing in particular is a significant challenge for testers.

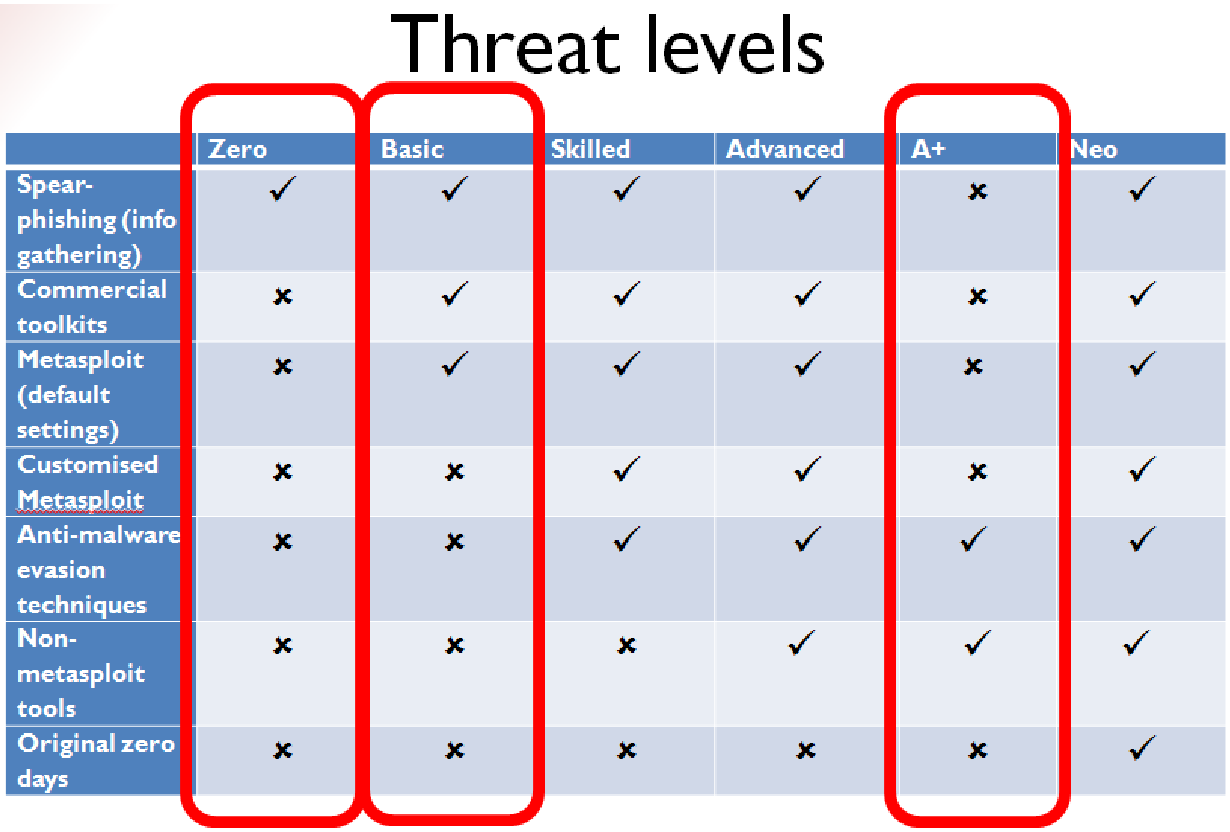
This document summarizes the main challenges faced in testing protection against targeted attacks, describes some of the important factors which should be taken into consideration when designing such tests, and proposes appropriate approaches to test design and implementation.

Like all AMTSO documents, this document should be read in conjunction with the AMTSO *Fundamental Principles of Testing* and other AMTSO testing guidelines, which can be found at www.amtso.org.

Defining the Threat

Targeted attacks come in a wide range of forms, from a single malicious actor honing a standard set of tools and techniques to defraud a single victim, to large and well-organized criminal gangs operating in concert and using a wide selection of bespoke software as well as hacking and social engineering techniques to penetrate the systems of a major corporation. This latter style of threat has come to be referred to as "advanced persistent threats" or "APTs", with many of the recent generation of solutions claiming to provide protection against just this type of danger. While this paper will focus on this more sophisticated type of threat, much of the information found herein may well be applicable to less high-level forms of targeted attacks.

The following "Zero-to-Neo Scale" shows a more detailed description of the various types of targeted attacks, and has been recopied here with permission from Virus Bulletin[i]:



Threat levels

	Zero	Basic	Skilled	Advanced	A+	Neo
Spear-phishing (info gathering)	✓	✓	✓	✓	✗	✓
Commercial toolkits	✗	✓	✓	✓	✗	✓
Metasploit (default settings)	✗	✓	✓	✓	✗	✓
Customised Metasploit	✗	✗	✓	✓	✗	✓
Anti-malware evasion techniques	✗	✗	✓	✓	✓	✓
Non-metasploit tools	✗	✗	✗	✓	✓	✓
Original zero days	✗	✗	✗	✗	✗	✓

Define the Threat

One of the first steps required when setting out to test solutions offering protection from targeted attacks is to define one's terms. The term "APT" has become a common buzzword in the security community, but its usage is varied with little agreement as to its exact application. In an appendix to this document is a section from a Virus Bulletin conference paper on "Testing APT Defenses", by a team of active AMTSO participants, looking at how the term has been applied. Their conclusion is that it is used with many different meanings in different contexts, and that in any objective or scientific setting it may be more appropriate to avoid the term entirely and instead to consider whatever any party describes as an "APT" to be a variety of targeted attack.

Whatever approach a tester chooses to take and whatever terminology they decide to use, it is vital that they fully understand and make clear to the consumers of their test data exactly what they mean when using such terms. For the purpose of the test to be clear and valid, it is necessary to define both the type(s) of solutions being tested, and the type(s) of threats those solutions will be pitted against.

`

Define the Approach

It is also important to decide on and describe the complexity of the test cases used, as more sophisticated targeted attacks will usually feature many stages. These may range from pre-attack reconnaissance and initial penetration via various activities within the compromised network to – in most cases – the final exfiltration, modification or destruction of sensitive data (and even, in some cases, physical hardware). Testers may choose to test individual attack stages separately, either as separate parts of a test suite or as standalone tests, or they may try to reproduce all stages of a full attack for the closest possible reflection of real-world conditions, but in any case they should make clear what they are doing and why.

Test Environments

When testing protection from targeted attacks, the test environment is of vital importance, perhaps more so than any other type of anti-malware test. The more complete and realistic the testing environment is, the more accurate the test results will be. The testing environments required to properly model a sophisticated targeted attack, and any potential solutions to detect, block or mitigate such attacks, are significantly more complex than those generally used for standard endpoint anti-malware tests.

A single machine is not enough to test protection from targeted attacks. Ideally the test environment should fully reproduce the type of environment threatened by the advanced attack; generally, that of a medium-to-large enterprise or institution. This environment should at the very least include a network of systems with standard gateways, storage, external services such as websites and email systems, and user endpoint systems, segregated into layers where appropriate.

A tester may create an appropriate environment to conduct a test using physical hardware or virtualization. However, where virtual machines are used in ways not commonly implemented in the real world, this should be noted in test reports, along with any effects of using virtualization that were taken into account when designing tests.

It may also be useful to include in the testing environment a range of known vulnerabilities which can be monitored as potential points of compromise. For example, some protection systems may use honeypot systems to trap attacks and allow them to be captured and analyzed without risk to genuine systems, again using known vulnerabilities to attract attackers.

In order to fully test the complete range of protections that products offer, or claim to offer, it is important that the environment contain features allowing a full spectrum of attack and defense types. For example, many solutions promise protection against attacks using previously unknown zero-day vulnerabilities. However, finding real examples of such flaws is likely to be resource-intensive and time-consuming, and keeping these examples private for testing purposes may risk endangering real-world users. There are ways to simulate zero-day threats, for example, by taking standard open-source software in widespread use and recompiling it with known flaws built in (for more information on this approach, see the paper proposing the concept presented at the 2014 Virus Bulletin conference[ii]).

`

For a test environment fully representative of the real world, it may also be useful to include human users, as social engineering is often a major part of a sophisticated targeted attack. Test design should include detailed plans as to how users, real or simulated, should respond to expected and unexpected inputs. It may also be of value to include live administrators, monitoring product outputs and responding to attacks in real-time; again, how such test components behave should be carefully planned and documented.

Of course, when building test environments connected to the internet in which real malware is expected to run, it is important to take all possible steps to minimize the risk of endangering other users and systems.

Test Attacks

Running test cases in such scenarios is more similar to penetration testing than to standard anti-malware testing. To accurately model the way advanced targeted threats operate, attacks should be designed to target the specific infrastructure used in the test, and possibly also the specific solutions being tested. A full test case should include the full path of an attack, from initial reconnaissance through compromise and network exploration to the final exfiltration or destruction of data.

It may be possible to divide a test in order to cover different stages separately, but care must be taken to ensure point-specific products are not given undue advantage over more holistic solutions which may rely on combinations of data sources from multiple points in the attack chain.

It may be useful for the attack to include a live human element which can rapidly respond to changes in the targeted environment, simulating real-life attackers. When using such an approach it is important to be aware of the risk of the test becoming a test of the penetrator's skills and knowledge. The particular attack methods that are used should be part of the test planning process, perhaps separating different techniques into categories and organizing them from least to most sophisticated.

False Positive Testing

As with all anti-malware testing, it is vital to include an element of false positive testing, building normal everyday activities into the test environment to ensure products do not block such activities, or produce unmanageably large amounts of logging data in which genuine attacks are hidden. The quantity and diversity of activities performed should scale in relation to the complexity of the test environment, with a more complete and complex environment expected to produce a greater quantity and diversity of everyday noise.

The Comparative Problem

In many types of testing it is desirable to compare multiple solutions. Comparative testing of products designed to detect or protect against sophisticated targeted attacks is rendered problematic by the very nature of those attacks.

For example, if Company A is running product X but suffers a breach, we can never say that product Y would have protected them better, even if it can be shown that product Y was capable of spotting and/or blocking all the techniques used in that breach. The simple reason for this is that had Company A been running product Y the attack would likely have been entirely different, designed to circumvent the protections of product Y rather than those of product X. Thus, it is fairly unrevealing to simply throw a range of attacks, of the sort thought to be commonly used in targeted attacks, against a range of products in order to count how many each manages to block.

One possible way to deal with this issue is to use a “Zero to Neo” test structure, as described above. The tester sets up their test environment, protected with the solutions they wish to test, and then attempts to compromise it using a range of simulated attack methods. Each method is categorized depending on the level of complexity and the skills, time or resources required to mount such an attack, and products can be rated comparatively depending on how they fare in each category. Solutions which require the most expertise or computing power to defeat would be considered superior.

It should be noted that some solutions may be designed only to spot the most advanced attacks and may, by design, be vulnerable to more basic attacks, perhaps leaving such issues to other solutions their users are expected to have in place. Products may also be sold as separate modules which can be run individually or in combination, and may provide different levels of protection depending on the modules selected.

Testers may want to consult with the developers of the products they wish to test before designing their system for categorizing products based on how they handle different classes of attack, to ensure they have a proper understanding of how each product is intended to be used. There are situations where it is quite acceptable to compare products with widely varying target markets and methods of operation, but in such cases it is important that testers are aware of how each solution works, and tests should be designed in a way which does not bias the results in favor of one particular approach or technology over another.

Interpreting Results

Measuring the success of different types of solution to complex threats is itself a complex business. Some solutions may block unwanted activities, while others may be designed only to detect and alert on unwanted events. Some test cases may succeed completely in compromising the test environment, while others may penetrate some areas but not others, or may be able to exfiltrate some files but not all the available data.

Test design should include a detailed breakdown of how different levels of “success” in detecting or blocking the test case should be measured and weighted, bearing in mind the potentially varying importance to the “victim” of various stages of the attack. It is important at this stage to fully understand the intended (or advertised) purpose of the solution(s) under test, and ensure products are not penalized for failing to provide features or functionality they did not intend to offer. It may also be valuable to consider the size and level of detail of a solution’s output, in order to judge the extent to which alerts can be swamped by large amounts of background noise.

`

In Summary:

Define your terms – make it clear what is being tested and how. This is particularly important if using terms with multiple conflicting applications, such as “APT”.

Select both solutions and “attack” test cases with care – be sure to understand what products promise to protect against, and how the test cases in use would operate in the real world. Include false positive testing too, to flag up issues with over-reporting.

Build the right environment – the systems and networks used in the test should reflect reality where possible, and should also offer a full range of potential penetration vectors.

Be aware of the difficulties of comparative testing – make it clear to your readers where comparisons may be inaccurate or biased towards specific approaches or technologies.

Interpret results with care – take into account different approaches and protections taking effect at different stages of an attack.

`

Appendix

What is an “APT”

The following excerpt is taken in whole from a section of “EFFECTIVELY TESTING APT DEFENCES: DEFINING THREATS, ADDRESSING OBJECTIONS TO TESTING, AND SUGGESTING SOME PRACTICAL APPROACHES”, Edwards, Ford, Szappanos, *Proceedings of the 25th Virus Bulletin International Conference (2015)*[iii]

What is an “APT”?

Although the term ‘APT’ is used commonly nowadays, there is no generally accepted definition for it, and this contributes greatly to the problem of testing. In part, the APT has become this year’s buzzword, but vendors, reviewers and users employ the term differently depending on circumstance and goal. Such definitional challenges only add to the confusion.

For example, *TechTarget* uses the following definition:

> ‘An advanced persistent threat (APT) is a network attack in which an unauthorized person gains access to a network and stays there undetected for a long period of time.’ [1]

According to this definition the sample must be undetected to be an APT. If a product detects a threat, then it is not an APT. This leads us to the inevitable outcome that the only valid outcome of a test of APT protection is that nothing is detected (otherwise the test sample is not an APT).

While this definition therefore significantly simplifies APT testing in general, it would make APT testing a very simple (non-existent) task so we should aim for a more practical one. Here are a few more definitions that are quite interesting:

Wikipedia:

> ‘APT is a set of stealthy and continuous computer hacking processes, often orchestrated by human(s) targeting a specific entity. APT usually targets organizations and/or nations for business or political motives. APT processes require a high degree of covertness over a long period of time.’ [2]

NSS Labs:

> NSS Labs adopts an alternative acronym for a targeted attack, referring to a Targeted Persistent Attack (TPA).

`

> 'Targeted: The attacker selected the organization, for a specific reason.
>
> Persistent: The attack is capable of using multiple command-and-control channels and attack vectors, and constantly increasing its penetration of your IT systems and resources. It is also stubborn, resisting remediation attempts.
>
> Attack: While the word 'threat' is somewhat nebulous when used in the context of APT, there is nothing unclear about it here. This is a true attack, and it may have several distinct stages.' [3]

Gartner:

> 'Advanced threat – any attack that gets past your existing defences.
>
> Persistent threat – any successful attack that goes undetected and continues to cause damage.
>
> Advanced persistent threat – any attack that gets past your existing defences, goes undetected and continues to cause damage.' [4]

The problem with these definitions is, once again, that they attribute being undetected to being a core feature of an APT. This definition renders APT defences and tests useless. Other definitions focus on other aspects:

RSA:

> 'An Advanced Persistent Threat (APT) is a targeted attack against a high-value asset or a physical system.' [5]

While this is a useful definition that makes it easy to determine if an attack belongs to this category, it does not explain the significance of the 'Advanced' and 'Persistent' attributes of an APT.

Damballa:

> 'Advanced Persistent Threats (APTs) are a cybercrime category directed at business and political targets. APTs require a high degree of stealithiness [sic] over a prolonged duration of operation in order to be successful…
>
> Advanced – Criminal operators behind the threat utilize the full spectrum of computer intrusion technologies and techniques…
>
> Persistent – Criminal operators give priority to a specific task, rather than opportunistically seeking immediate financial gain…

`

Threat – means that there is a level of coordinated human involvement in the attack, rather than a mindless and automated piece of code…' [6]

We have a lot of definitions that attempt to define APT on an abstract level, hardly helping testers to categorize test scenarios. Our best option at this point is to change scope and deal with a better defined and more practical definition of targeted attacks along the lines of the RSA definition.

The terms 'APT' and 'targeted attack' are often used synonymously by the press and the APT protection providers so it makes sense to stick to the easily definable 'targeted attack' cases in test scenarios.

For practical purposes of testing we will define targeted attacks as follows:

> ***A targeted attack is an infection scenario executed against a limited and pre-selected set of high-value assets or physical systems with the explicit purpose of data exfiltration or damage.'*** (emphasis added)

REFERENCES

[1] Advanced Persistent Threat (APT). TechTarget. November 2010. http://searchsecurity.techtarget.com/definition/advanced-persistent-threat-APT

[2] Advanced Persistent Threat. Wikipedia. http://en.wikipedia.org/wiki/Advanced_persistent_threat

[3] The Targeted Persistent Attack (TPA). NSS Labs. 19 August 2012. [https://www.nsslabs.com/blog/the-targeted-persistent-attack-tpa-when-the-thing-that-goes-bump-in-the-night-really-is-the-bogeyman/]

[4] Defining the "Advanced Persistent Threat". Gartner. 11 November 2010. [http://blogs.gartner.com/john_pescatore/2010/11/11/defining-the-advanced-persistent-threat/]

[5] Juels, A.; Yen, T.-F. Sherlock Holmes and the Case of the Advanced Persistent Threat. RSA, 2012. [https://www.usenix.org/system/files/conference/leet12/leet12-final29.pdf]

[6] Advanced Persistent Threats: A Brief Description. Damballa. [https://www.damballa.com/paper/advanced-persistent-threats-a-brief-description/]

Excerpt from "EFFECTIVELY TESTING APT DEFENCES: DEFINING THREATS, ADDRESSING OBJECTIONS TO TESTING, AND SUGGESTING SOME PRACTICAL APPROACHES", referenced above, is reproduced with kind permission of Virus Bulletin. Full paper, recording and slides are available on the Virus Bulletin website[iv]

`

This document was adopted by AMTSO on June 17, 2016

[i] https://www.virusbulletin.com/uploads/pdf/magazine/2016/vb201601-effectively-testing-APT-defences.pdf
[ii] https://www.virusbulletin.com/uploads/pdf/conference/vb2014/VB2014-FordCarvalho.pdf
[iii] https://www.virusbulletin.com/uploads/pdf/magazine/2016/vb201601-effectively-testing-APT-defences.pdf
[iv] https://www.virusbulletin.com/blog/2016/01/paper-effectively-testing-apt-defences/