HEURISTIC SCANNING AND SANDBOX APPROACH IN MALWARE DETECTION

Petar Čisar, PhD[1]
University of Criminal Investigation and Police Studies, Belgrade, Serbia

Dušan Joksimović, PhD[2]
University of Criminal Investigation and Police Studies, Belgrade, Serbia

Abstract: A heuristic approach in malware detection is similar to the method of detecting anomalies applied to the intrusion detection system (IDS). It speeds up the process of finding sufficiently good solution in situations where the implementation of detailed research is not practical or is very time-consuming - for example, using various general rules, informed speculation, intuition and common sense. Instead of looking for matches (like in static signature-based detection), heuristic intrusion detection looks for behavior that is out of ordinary, with regards to a baseline of the normal network traffic and activity. Heuristic scanning uses rules and/or algorithms to look for commands which may indicate malicious intent without needing a signature. Analysis of static signatures will fail to catch new types of attacks but have usually less false positives. Heuristics might catch more new malware but this usually comes with higher false positive rate. Because of that, most modern and efficient IDS software uses both signature and heuristic-based methods in combination, with the goal of increasing the chance to detect and remove malware. In parallel with the heuristic and signature-based method, sandboxing approach is also used in detection of network anomalies. This is a software management technique that isolates examined applications from critical system resources and other programs. Without sandboxing, an application may have unrestricted access to all system resources and user data on a computer. Similar to heuristics, this method also has its benefits and limitations. The general conclusion is that the best network security can be achieved utilizing more methods simultaneously - by multi-scanning (scanning with multiple anti-malware engines).

Keywords: heuristics, scanning, malware, signature, sandboxing, detection.

[1] petar.cisar@kpu.edu.rs
[2] dusan.joksimovic@kpu.edu.rs

INTRODUCTION

Intrusion detection is an act of identification such activities, which are aimed at compromising confidentiality, integrity and availability (security objectives) of computer resources. The goal of intrusion detection is to identify malicious entities.

Depending on what is being detected (analysis strategy), intrusion detection systems can detect misuses (known attacks and attacks from inside) or anomalies (concentrates on unusual network activity (anomaly) that may indicate the intrusion).

Misuse detection is based on "signatures" (sequences of known attacks) and rules of known attacks. The analysis of static signatures will fail to catch new types of attacks but usually has less false positives. Network anomalies appear in different forms such as:

- protocol anomalies (refers to protocol format and behavior - usually TCP/IP),
- anomalies of the content of application (requires knowledge of application semantics) and
- statistical anomalies (characterization of traffic through statistical measurements - monitoring the behavior of a user or system by measuring statistical variables over a certain period of time).

Applied statistical algorithm must recognize the difference between long-term and short-term changes, to avoid generating false alarms during normal traffic variations. The advantage of statistical anomaly detection is that it can detect attacks that have not been recognized by other detection mechanisms and there is no need for large databases of attack signatures.

Sandboxing is a type of approach in the prevention of a network intrusion where the input code (ActiveX, Java applets and scripts in different languages) is placed in quarantine (sandbox), i.e. restricted area. The system executes the code there and monitors its behavior. If the code violates a predefined policy, it will be stopped and the system will be protected from its further execution and possible attack.

HEURISTIC APPROACH

Heuristics is, in general, a kind of informal decision theory that strives to combine elements of rational (controlled, effortful, slow and often serial, may be abstract, rule-based) and intuitive decision theory (automatic, effortless, rapid and parallel, concrete, associative) because very often the decision should be made quickly, even at the risk of making the wrong decision.

In the context of malware scanning, heuristics can be considered as the opposite of signature-based scanning, which looks for match signatures found in analyzed files with that of a database of known malware. Heuristic scanning uses rules and/or algorithms to look for commands which may indicate malicious entity. Using this approach, some heuristic scanning methods are able to detect malware without needing a signature.

A lot of modern viruses are only slightly changed versions of conceptions developed years ago. Specific detection methods like signature scanning became very efficient ways of detecting known threats. Finding specific signature in code allows the scanner to recognize every virus whose signature has been stored in the built-in database. Problem occurs when source code of virus is modified by a programmer or mutation program. Signature is being modified due to even minor changes. A virus can behave in an exactly the same way and be undetected due to new signature. The key question is how to recognize a virus with no information about its structure. The answer is in examining its behavior and characteristics.

Heuristic scanning in its basic form incorporates three procedures (called metaheuristics):

- pattern matching (methods of comparison samples)
- automatic learning (self-adjusting system (machine), whose control algorithm changes in accordance with the control results obtained so that with the passing of time the machine improves its characteristics and quality of performance)
- environment emulation (imitation of complete hardware - allows to run software written for another computer or device)

The main idea of heuristic scanning is to examine program sequences and qualify them by their potential destructiveness. If there are sequences behaving suspiciously, the analyzed program can be qualified as a virus.

In anti-virus (AV) software, heuristic scanning is implemented to recognize threats by following built-in rules. Single suspicion is not a sufficient reason to trigger the alarm. But if a program tries to stay resident and contains instructions to search for executable files, it probably is a real virus. AV software very often classifies sequences by their behavior granting them a flag (importance). Every flag has its weight and if total values for one program exceeds a predefined threshold, the scanner regards it as virus.

Heuristic based anti-malware software uses different scanning methods, including[3]:

- File analysis - During this analysis, the scanning software will precisely examine a file to determine its purpose, destination and intent. For example, if the purpose of a file is deletion of some other files, it could be flagged as a virus.
- File emulation - Otherwise called dynamic scanning or sandbox testing, file emulation tests a file in a virtual condition to perceive what occurs. If the file behaves like a virus, it is presumably a virus.
- Generic signature detection - Intended to find different forms of a malware, this method utilizes earlier virus definitions to locate viruses inside the same family.

[3] Forcepoint, https://www.forcepoint.com/cyber-edu/heuristic-analysis

This type of scanning has its advantages and disadvantages.

Advantages of heuristic scanning:

- The process of heuristic scanning is significantly quicker than sandboxing because it does not execute the examined file and then wait to record and analyze its behavior (except for some emulation-based techniques).
- The rules in heuristic engines based on the latest threat vectors can be updated without the malicious person finding out the details.
- It does not provide details on how malware is flagged, so malware creators do not know what they have to change so as to avoid detection.
- Heuristic scanning can identify malware that can avoid sandbox detection.

Disadvantages of heuristic scanning:

- As a result of scanning, the information found is generally connected with the name of threat.
- Because the scanning engines are searching for specific parts of code which show a malicious activity, two limitations are possible:
    - If the creator has not implemented the detection for a specific activity, at that point the malware will avoid detection.
    - If the malicious code is camouflaged (for example, inside an encoded file), it will dodge detection.
- Some of the earlier strategies for heuristic-based scanning have a higher tendency for reporting false positives since they are searching for a wide scope of activities that could identify a potentially malicious file. But, newer strategies for heuristic scanning, for example, generic detection, produce false positives less often. Generic detection is based on searching for features or behaviors that are generally observed for known threats.

Multi-layered in-depth detection is a newer and more efficient detection concept which consists of different phases (or layers): reputation matching, heuristic detection, virtualized execution and threat analysis. The structural scheme of this concept is presented in the following figure.

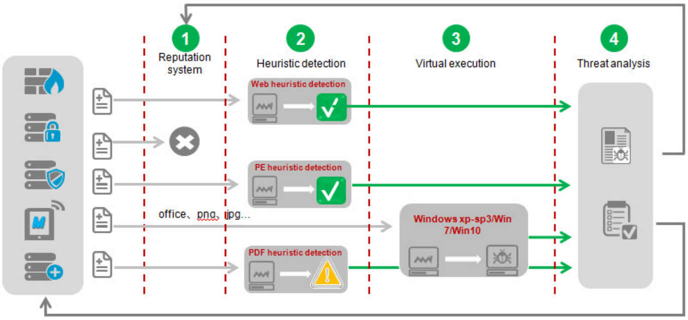


Figure 1: *Layered defense system*[4]

[4] Huawei, https://e.huawei.com/en/related-page/products/enterprise-network/security/apt/firehunter6300/brochure/security_firehunter6300_en

The signature scanning methods provide precise scanning by comparing viruses with their signatures. But, these methods fail to detect new and unknown forms of viruses. At the same time the generic techniques can detect new viruses without using signatures. But, these methods are more likely to generate false positives. Based on the above, it can be concluded that there is a positive correlation between the capability to detect new viruses and false positive rate. For example, the heuristic scanning can detect new and unknown viruses but it is more sensitive to false positives.

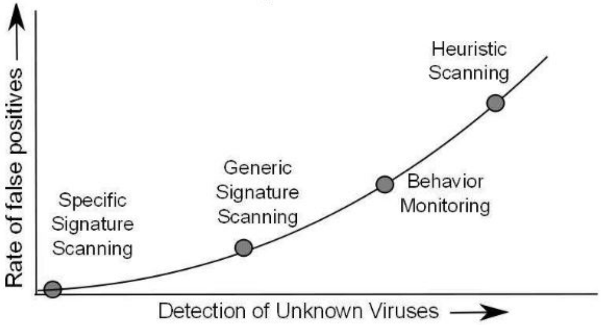


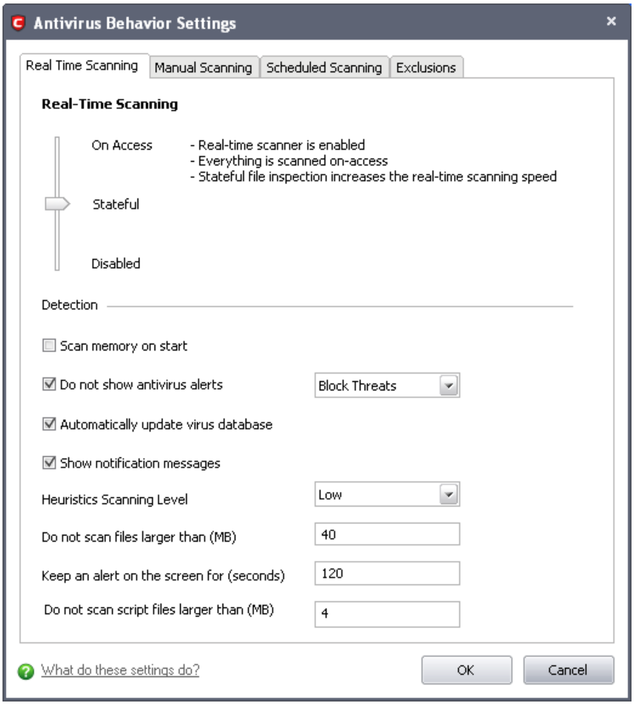
Figure 2: *Correlation between new viruses and false positives*[5]

AV software employs various heuristic techniques to identify previously unknown viruses. Heuristics is about detecting virus-like behavior or attributes rather than looking for a precise virus signature that matches a signature on the virus blacklist. This is a significant progress in finding malicious scripts and programs as it allows the engine to 'predict' the existence of new viruses - even if it is not contained in the actual virus database.

Anti-malware software may have built-in heuristic capabilities. One of the ways to activate it is to adjust the scanning level, within the real-time scanning options. The appropriate behavior settings allow selecting the level of heuristic scanning (Figure 3.):

- Off - Disables heuristic scanning - the software uses only the signature database to determine whether a file is malicious or not.
- Low (often recommended) - Allows lowest sensitivity to detecting unknown threats but will also generate the fewest false positives. This setting combines a high level of security with a low rate of false positives.
- Medium - Detects unknown threats with greater sensitivity than the 'Low' option but with increase of the possibility of false positives.
- High - Highest sensitivity to detecting unknown threats but this also increases the possibility of more false positives.

[5] Umakant Mishra: Finding and Solving Contradictions of False Positives in Virus Scanning, Semantic Scholar, https://pdfs.semanticscholar.org/6666/30ad1ec7fdf70d1441c1c883264b3bdee20f.pdf



Real-Time Scanning

On Access - Real-time scanner is enabled
- Everything is scanned on-access
- Stateful file inspection increases the real-time scanning speed

Stateful

Disabled

Detection

☐ Scan memory on start

☑ Do not show antivirus alerts: Block Threats

☑ Automatically update virus database

☑ Show notification messages

Heuristics Scanning Level: Low

Do not scan files larger than (MB): 40

Keep an alert on the screen for (seconds): 120

Do not scan script files larger than (MB): 4

Figure 3: *Heuristics scanning adjustment*[6]

SANDBOXING

Sandbox consists of special built environment, generally virtualized (in some cases physical), where the possibly malicious files are executed and their behavior is recorded. The recorded behavior is then analyzed automatically through a weight system in the sandbox and/or manually by a malware expert. The objective of this analysis is to decide if the file is malicious and if it is, what exactly the file does.

This security mechanism is designed to run untrusted (or exploitable) code, in a manner that prevents the encapsulated code from damaging the rest of the system. The aim of a sandbox is to isolate threats - it provides protection by isolation, not detection.

Sandbox approach in layered configuration can also be successfully used for web and email security as well as for filtering network traffic (figures below).

[6] Comodo, https://help.comodo.com/topic-72-1-284-3011-.html

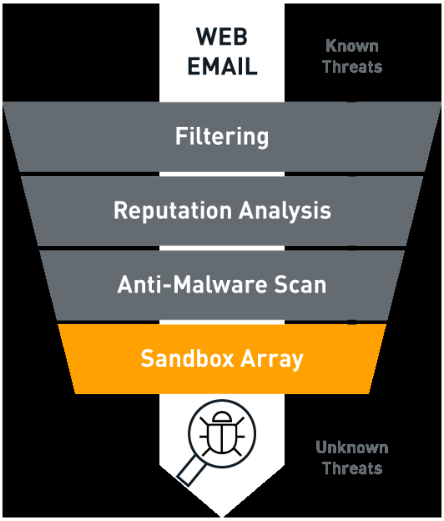


Figure 4: *Web and email security - sandboxing solution*[7]

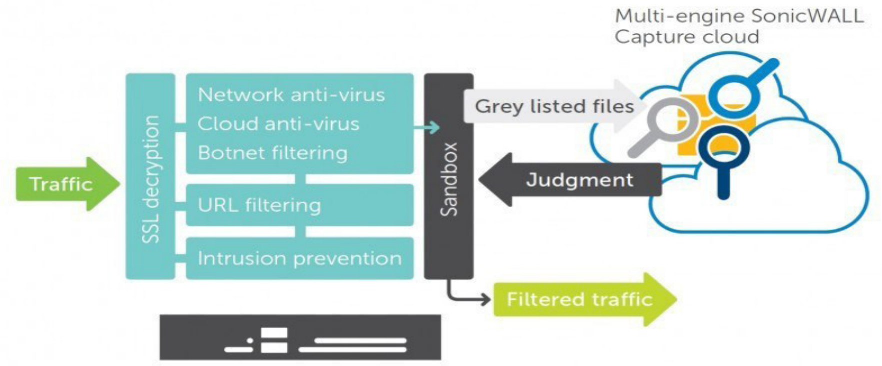


Figure 5: *Traffic filtering using sandbox approach*[8]

Similar to heuristic scanning, sandbox approach also has its benefits and limitations.[9]

Benefits of sandboxing:

- Considering that sandboxing practically opens the file being examined, it can find precisely what that file will do in specific environment.
- Instead of binary form of data and threat name, the great part of sandboxes offers reporting with details on threat behavior. Besides giving additional information on the best way to characterize the file, this technique can be especially helpful in an incident response environment so as to distinguish precisely what the aim of the file has been, in order to explain what the results are.

[7] Cyren, https://www.cyren.com/products/cloud-sandboxing

- Many products offer the capacity to provide a very customized actions. For instance, a part of malware that is intended to just completely execute on a concrete client's machine can be replicated.

Limitations of sandboxing:

- Due to the possibility of identifying the applied methodology and customization that is accessible in business sandboxes, malware makers can create adequate practices to avoid detection. This incorporates two classes:
    - Malware intentionally created to be executed in a sandbox and which will act differently so as to not be flagged as malicious. This might be as easy as not running on any virtual machine, or more progressive searching for signs characteristic to a sandbox.
    - Malware makers have made solutions which cannot be detected by the sensors of a specific sandbox.
- It is necessary to provide conditions for sample execution and the time to generate full reports, especially if trying to execute stalled code takes a lot of time and hardware resources to analyze a given sample, causing lower throughput.
- Although the security trend is towards automated sandboxes, a large number of them only provide the raw data on behavior of the malware, so it is necessary to either build a custom application or have an expert to interpret the results.
- Due to the considerable processing time, numerous newer sandboxes are cloud-based, which makes sensitive files hard to use.

The effectiveness of heuristics and sandboxing dominantly depends on the setting of rules. These rules should be forward-looking enough to identify unknown threats and also capable to avoid the appearance of false alarms. One of the possibilities for testing the effectiveness of applied rules is to perceive how well the rules respond to zero-day attacks. Zero-day attacks are malware utilizing unpatched vulnerabilities but with similar exploitation methods. If heuristic rules are well-designed and implemented, they will be able to catch them.

For example, rules developed in the Trend Micro Advanced Threat Scan Engine have been able to detect the following zero-day attacks:[10]

1. CVE (Common Vulnerabilities and Exposures)-2014-0515 in May 2014 was detected by a rule developed in 2014 (HEUR_SWFJIT.B)
2. CVE-2014-1761 in April 2014 was detected by a rule developed in 2012 (HEUR_RTFEXP.A/HEUR_RTFMALFORM)
3. CVE-2014-0496 in February 2014 was detected by a rule developed in 2010 (HEUR_PDFEXP.A)
4. CVE-2013-3346 in November 2013 was detected by a rule developed in 2010 (HEUR_PDFEXP.A).

Testing the performance of a virus scanner has always been a sensitive topic, bearing in mind that small number of testers and organizations are recognized as sufficiently competent by other members of this field. The testing organizations

[10] https://blog.trendmicro.com/trendlabs-security-intelligence/heuristic-scanning-and-sandbox-protection-best-of-both-worlds/

generally considered to be competent in this area include: AV Comparatives, AV-Test.org, ICSA Labs, SC Magazine/West Coast Labs, Virus Bulletin, AMTSO (Anti-Malware Testing Standards Organization) and others. Without generally accepted competent reference test set, there is no assurance that scanners are really being tested against actual, working malware. Practically, the majority of the AV vendors provide daily (or more often) adequate updates, so testing a scanner when it is a few months outdated does not say much regarding its present detection abilities.

CONCLUSION

Both heuristic-based scanning and sandboxing present specific advantages and disadvantages and for various situations one scanning technique might be more proper than the other. More reliable security results are achieved by using both methods at the same time in order to minimize the quantity of samples which might probably sidestep detection. Multi-scanning (scanning with multiple engines) is an advantage of the differing heuristic algorithms of many other scan engines.

REFERENCES

1. Joxean Koret, Elias Bachaalany: The Antivirus Hacker's Handbook, John Wiley & Sons, Inc., 2015.
2. Dragan Pleskonjić, Borislav Đorđević, Nemanja Maček, Marko Carić: Sigurnost računarskih mreža, Beograd, Viša elektrotehnička škola, 2006.
3. Aleksandar Lazarevic, Vipin Kumar, Jaideep Srivastava: Managing Cyber Threats: Issues, Approaches and Challenges, Kluwer Academic Publishers, Boston, doi: 10.1007/b104908, 2005.
4. Mohaddeseh Zakeri, Fatemeh Faraji Daneshgar, Maghsoud Abbaspour: A Static Heuristic Approach to Detecting Malware Targets, Wiley Online Library, 2015, https://onlinelibrary.wiley.com/doi/full/10.1002/sec.1228
5. Muhammad Ali, Stavros Shiaeles, Maria Papadaki, Bogdan Ghita: Agent-based vs Agent-less Sandbox for Dynamic Behavioral Analysis, Global Information Infrastructure and Networking Symposium (GIIS), 2018.
6. Anna Bryk: Sandbox-Evading Malware: Techniques, Principles, and Examples, 2018, https://www.apriorit.com/dev-blog/545-sandbox-evading-malware
7. Konrad Rieck, Philipp Trinius, Carsten Willems, Thorsten Holz: Automatic Analysis of Malware behavior Using Machine Learning, Journal of Computer Security, 19(4), pp. 639–668, 2011
8. Joris Kinable, Orestis Kostakis: Malware Classification Based on Call Graph Clustering, Journal in Computer Virology, 7(4), pp. 233–245, 2011

9. Zahra Bazrafshan, Hashem Hashemi, Seyed Mehdi Hazrati Fard, Ali Hamzeh: A Survey on Heuristic Malware Detection Techniques, 5th Conference on Information and Knowledge Technology (IKT), pp. 113-120, 2013
10. David Harley, Andrew Lee: Heuristic Analysis - Detecting Unknown Viruses, White Paper, Eset, 2009, https://www.welivesecurity.com/media_files/white-papers/Heuristic_Analysis.pdf
11. Umakant Mishra: Finding and Solving Contradictions of False Positives in Virus Scanning, Semantic Scholar, https://pdfs.semanticscholar.org/6666/30ad1ec7fdf70d1441c1c883264b3bdee20f.pdf
12. Comodo, https://help.comodo.com/topic-72-1-284-3011-.html
13. Huawei FireHunter, https://e.huawei.com/en/related-page/products/enterprise-network/security/apt/firehunter6300/brochure/security_firehunter6300_en
14. Datasharp - integrated communications, https://datasharp-ic.co.uk/blog/dell-security-multi-engine-approach-advances-sandboxing-beyond-threat-detection-to-complete-prevention-with-new-sonicwall-capture-advanced-threat-protection-atp-service
15. SANS Institute, Protection from the Inside, http://www.sans.org/reading-room/whitepapers/analyst/protection-inside-application-security-methodologies-compared-35917
16. OPSWAT, https://www.opswat.com/blog/understanding-heuristic-based-scanning-vs-sandboxing