Methods for Detecting Malware Using Static, Dynamic and Hybrid Analysis

Alexandru-Radu BELEA
Faculty of Electronics, Telecommunications, and Information Technology,
University POLITEHNICA of Bucharest, Romania
alexandru.belea@stud.etti.upb.ro

Abstract

Malware analysis is the process of locating and examining malicious software or code with the aim of comprehending its operation and developing countermeasures. Malware can take many forms, such as viruses, worms, Trojans, and ransomware, and can cause significant harm to individuals, organizations, and even entire countries. To determine a piece of malware's purpose, potential effects, and capabilities, malware analysis entails examining the behavior, structure, and functionalities of the malware. Malware analysts are essential to the cybersecurity sector because they strive to spot dangers, eliminate them, and defend against online attacks. By using the knowledge gleaned from malware analysis, security solutions can be created that will better protect businesses from dangerous software. Malware analysis is a crucial part of any successful cybersecurity strategy in the continually changing threat landscape of today. In this article, we will explore the key concepts of malware analysis, including its purpose, techniques, and tools and we will contrast methods for detecting malware using static, dynamic, and hybrid analysis.

Index terms: dynamic analysis, hybrid analysis, malware, PE file, static analysis

1. Introduction

Malware analysis is the process of dissecting malicious software to understand its behavior and capabilities. With the increasing prevalence and sophistication of cyberattacks, the need for effective malware analysis has become more critical than ever. Malware analysts use a variety of techniques to analyze malware, including dynamic and static analysis, to identify its purpose and potential harm. The insights gained through malware analysis can be used to develop effective countermeasures and strengthen cybersecurity defenses. In this article, we will explore the importance of malware analysis, the different types of analysis techniques, and best practices for conducting effective analysis. We will also examine real-world examples of malware and how they were analyzed to provide valuable insights into the nature of cyber threats [1] [2].

Malware analysis is a crucial task in cybersecurity because it provides insights into the tactics, techniques, and procedures (TTPs) used by cybercriminals to create and distribute malware. By analyzing malware, cybersecurity experts can develop effective countermeasures to protect against malware attacks and improve the overall security posture of organizations and individuals. There are several types of malware analysis techniques, including static analysis, dynamic analysis, and hybrid analysis.

In addition to analyzing malware samples, cybersecurity experts also use malware analysis techniques to develop and improve security solutions, such as antivirus software, intrusion detection systems, and firewalls. By understanding how malware works and how it can be detected and

prevented, cybersecurity experts can design more effective security solutions that protect against both known and unknown threats [2] [3].

2. Malware Analysis Techniques

Malware analysis techniques are used to understand the behavior and characteristics of malicious software or malware. Malware can be designed to evade detection and analysis, so various techniques are used to overcome these challenges. One of the most widely used and effective techniques is static, dynamic and hybrid analysis, which we will look at in more detail (depicted in Figure 1). Other common malware analysis techniques are:

- Reverse Engineering: This technique involves decompiling the malware's code to understand how it works.
- Sandboxing: This technique involves running the malware in a controlled environment that isolates it from the rest of the system. Sandboxing can help identify the malware's behavior without the risk of infecting the host system.
- Memory Analysis: This technique involves analyzing the contents of a computer's memory while the malware is running. Memory analysis can help identify the malware's activities, such as code injection and network communication.
- Network Analysis: This technique involves analyzing the network traffic generated by the malware. Network analysis can help identify the malware's command-and-control servers and the data it is exfiltrating.
- Behavioral Analysis: This technique involves analyzing the malware's behavior to understand its intent. Behavioral analysis can help identify the malware's target and its objectives [3] [4].

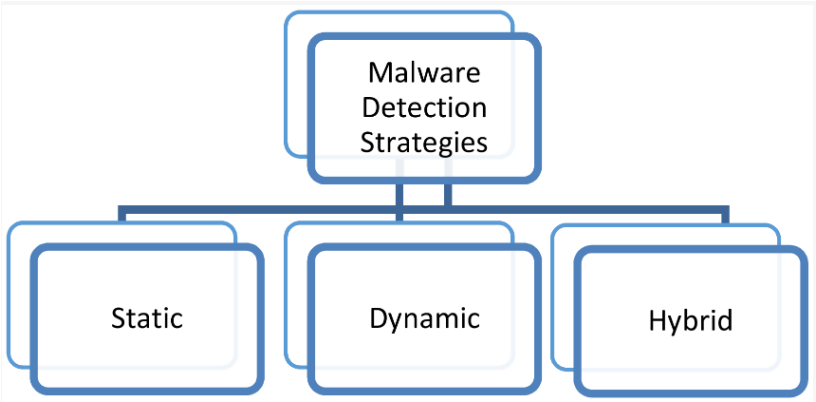

Fig. 1. Malware Detection Strategies [5]

2.1. Static Analysis

Static analysis involves examining the code and structure of the malware without executing it. This technique provides valuable insights into the behavior and potential impact of the malware on a system. Some common static analysis techniques include:

- Disassembling: Converting the binary code of the malware into assembly language to understand the functionality and logic of the code.
- Decompiling: Reversing the compiled code into high-level programming language code to understand the purpose of the malware.
- Debugging: Analyzing the code in a debugging environment to identify vulnerabilities and potential attack vectors [6].

2.2. Dynamic Analysis

Dynamic malware analysis is a technique used to analyze malware by observing its behavior in a controlled environment. This method involves running the malware in a secure and isolated environment, such as a virtual machine, and monitoring its activity to understand its behavior, capabilities, and potential impact. During dynamic malware analysis, the malware is executed and observed as it interacts with the system and network. This allows analysts to identify the malware's functions, including how it spreads, what it communicates with, and what it tries to achieve [6].

2.3. Hybrid Analysis

Hybrid analysis combines the strengths of both static and dynamic analysis. It involves first performing a static analysis to gather as much information as possible about the malware. This can include extracting any embedded files or configuration data, identifying any code obfuscation techniques, and looking for any signs of anti-analysis or anti-debugging techniques.

Once the initial static analysis is complete, the malware is then run in a controlled environment for dynamic analysis. This can include running the malware in a sandbox, using a debugger to step through the code, or running the malware on a virtual machine. The goal of the dynamic analysis is to observe the malware's behavior in a controlled environment and identify any malicious activity that may not have been apparent during the static analysis [3].

3. Types of Malware

Malware comes in various forms, each with its unique characteristics, effects, and methods of delivery. We will discuss some of the most common types of malware:

- Virus: A virus is a type of malware that infects executable files or system boot sectors, making them behave differently from their intended purpose. A virus replicates itself by attaching its code to other executable files, spreading its infection to other systems when the infected file is shared or transferred.
- Worm: Worms are self-replicating malware that spread over networks, exploiting system vulnerabilities to propagate from one computer to another. Unlike viruses, worms do not require a host program to spread, as they can self-replicate and spread autonomously. Worms can cause network congestion and slow down computer systems by consuming system resources, and they can also be used to steal sensitive information from infected systems.
- Trojan: A Trojan, also known as a Trojan horse, is a type of malware that disguises itself as a legitimate program to trick users into downloading or installing it. Once installed, the Trojan can give the attacker remote access to the infected system, allowing them to steal sensitive data, install other malware, or use the infected system as a part of a botnet.
- Ransomware: Ransomware is a type of malware that encrypts the victim's files, making them inaccessible until a ransom is paid to the attacker. Ransomware can be delivered through email attachments, infected websites, or social engineering attacks.
- Adware: Adware is a type of malware that displays unwanted advertisements on the victim's computer, usually in the form of pop-ups or banners. Adware can slow down the victim's computer, consume bandwidth, and track the user's internet activity.
- Spyware: Spyware is a type of malware that secretly monitors the victim's computer activity, collecting sensitive information such as login credentials, banking information, and personal data. Spyware can be used for identity theft, financial fraud, and espionage. Spyware can be delivered through infected websites, email attachments, or social engineering attacks [7].

4. The Advantages of Techniques

4.1. Benefits of Static Malware Analysis

There are several advantages of static malware analysis, including:

- Safe and controlled environment: Static malware analysis provides a safe and controlled environment for examining malware without the risk of infecting a real system.
- Ability to analyze large volumes of malware.
- Detection of hidden or obfuscated code: Malware authors often use obfuscation techniques to hide the true nature of their code. Static analysis can reveal the hidden code and make it easier to identify the malware.
- Identification of malware functionality: Static analysis can reveal the functionality of malware, including its ability to communicate with command-and-control servers.
- Can be used to create signatures that can be used by antivirus software to detect and block malware.
- Static analysis can be used for malware reverse engineering to understand how the malware operates and to develop countermeasures [2] [8].

4.2. Benefits of Dynamic Malware Analysis

There are several benefits to using dynamic malware analysis, including:

- Detection: Dynamic analysis can detect malware that may not be detected by traditional signature-based antivirus solutions.
- Identification: Dynamic analysis can identify the capabilities of malware, such as whether it has the ability to steal data or control a system.
- Rapid Response: Dynamic analysis provides security teams with the ability to rapidly respond to emerging threats.
- Flexibility: Dynamic analysis can be used to analyze a wide range of malware types, including new and unknown threats.
- Forensic Analysis: Dynamic analysis provides forensic investigators with a wealth of information about the behavior of malware, including network activity, system changes, and other actions [2] [8].

4.3. Benefits of Hybrid Malware Analysis

Some of the benefits of hybrid malware analysis include:

- Improved accuracy: Hybrid malware analysis can improve the accuracy of malware detection and analysis by combining multiple approaches, which increases the likelihood of identifying malicious behavior that may have been missed by a single technique.
- Faster analysis: By using multiple approaches simultaneously, hybrid analysis can speed up the malware analysis process.
- Better understanding of malware behavior: Hybrid malware analysis can provide a more complete picture of the behavior and capabilities of malware. For example, static analysis can reveal information about the structure and code of the malware, while dynamic analysis can provide insights into its runtime behavior.
- Effective response to sophisticated threats: Hybrid malware analysis is useful for detecting and responding to sophisticated threats that use advanced evasion techniques, such as polymorphism or obfuscation.
- Enhanced threat intelligence: Hybrid malware analysis can help build a more robust threat intelligence database by identifying commonalities and patterns among different malware samples [2] [4].

5. Malware Analysis on PE Files

PE (Portable Executable) format is a file format used for executables, DLLs (Dynamic Link Libraries), and other Windows operating system components. It was introduced in Windows NT 3.1 and has been used in all subsequent versions of Windows. The PE format is designed to be portable across different Windows systems, with the ability to handle different memory layouts and processor architectures. It contains information about the binary file, such as headers, sections, and resources [9] [10].

The main components of a PE file are: 1.DOS header (includes a small DOS executable to support older versions of Windows); 2.PE header (contains information about the file, such as the number of sections, entry point address, and the file checksum); 3.Section headers (describe the sections in the file, including their size, attributes, and virtual addresses); 4.Import and export tables (contain information about the functions and symbols that are imported and exported); 5. Resource section; 6.Debug information (contains debugging symbols and other information used by debugging tools) [9] [10].

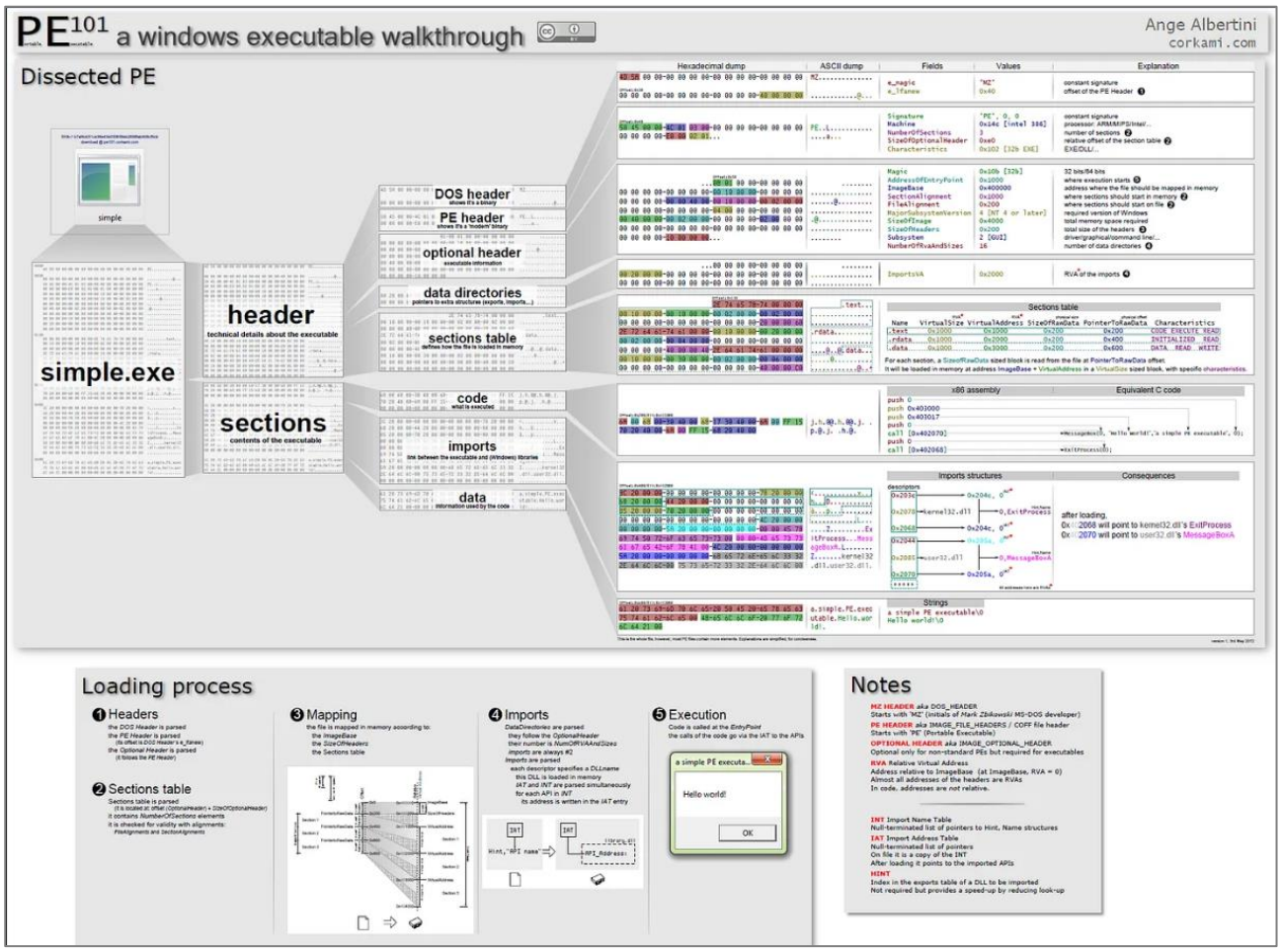

Fig. 2. PE file structure [9]

PE files are the primary format in which malware that targets Windows computers is distributed. Loaded libraries and imported functions are among the most crucial, if not the most crucial, pieces of data that we can statically extract from our malware. We can infer the features of the malware from these imported functions and libraries. For instance, the malware will employ some sort of network capabilities if it references the "ws2 32.dll" file. Another illustration is when malware imports the function "CreateProcessA" from kernel32.dll, indicating that the file will create a process

at some point while it is being executed. We'll be using a program called "Dependency Walker" to search our file and display any imported libraries and functions. Any libraries and imported functions that might be a sign of what the malware will do when run are what we need to be on the lookout for (Fig. 3) [9].

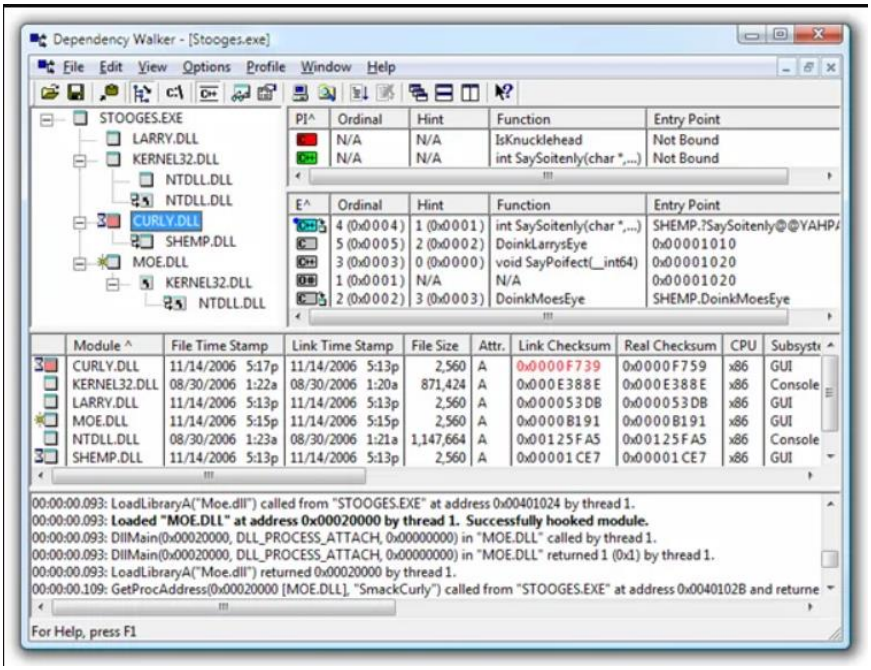

Fig. 3. Analysis of libraries [9]

The PE file format has sections and headers, as we've already explained. The resource part, also known as the.rsrc portion, is among the fascinating sections to examine. This area houses items like pictures, icons, and language strings. Malware writers occasionally use this area to conceal executables that will be used by the main program of the malware at some point during execution. Using Angus Johnson's "Resource Hacker," we may browse the resource area and begin looking for any "suspicious" or "malicious" indications (Fig. 4) [9].

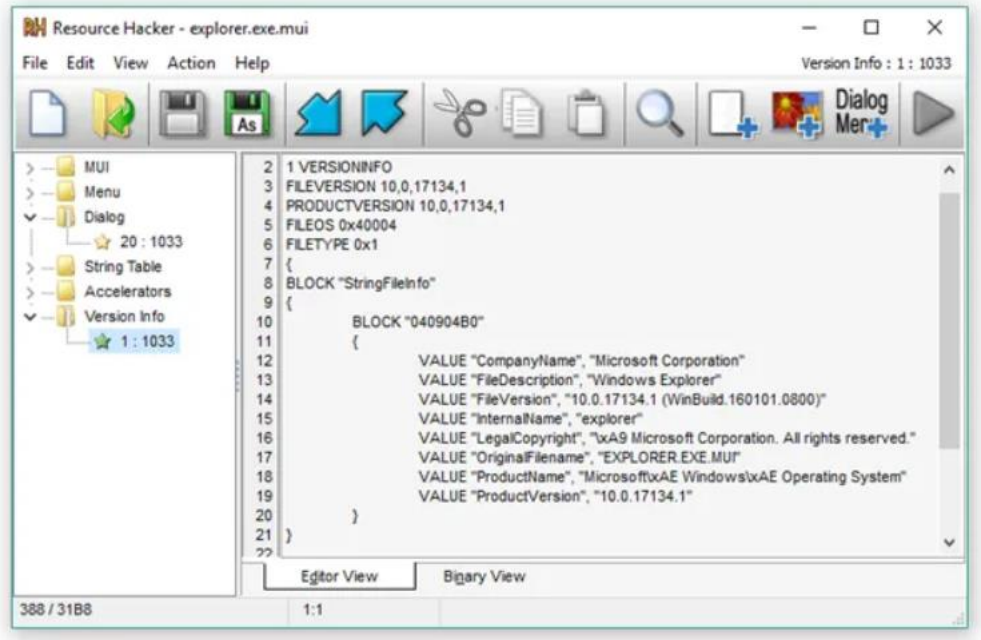

Fig. 4. Analysis of resources [9]

6. Conclusion

Following the static analysis we can see that it is not efficient compared to the dynamic or hybrid analysis. It is necessary to use different tools to analyze all the components of a PE file format, which implies a longer time to generate results.

In conclusion, static, dynamic, and hybrid malware analysis are important methods for detecting and analyzing malware. Static analysis examines code without actually executing it, while dynamic analysis observes the behavior of malware running in a controlled environment. Hybrid analysis combines both static and dynamic analysis techniques to provide a more comprehensive analysis of malware.

Each approach has strengths and weaknesses, and choosing the best analysis method depends on the specific context of the malware being analyzed. Static analysis helps identify known malware signatures and identify patterns that indicate malicious code. Dynamic analysis excels at detecting new, unknown malware that may evade detection by antivirus software.

Hybrid analysis offers the benefits of both static and dynamic analysis, allowing analysts to gain an in-depth understanding of malware behavior while identifying both known and unknown malware.

Overall, the choice of analytical method depends on the specific analytical goals and expertise of the analyst involved. Regardless of the method you choose, the ultimate goal is to detect and analyze malware and protect your system and users from its harmful effects.

References

[1]. Kurt Baker, “Malware Analysis,” 4 January 2022, Crowdstrike, https://www.crowdstrike.com/cybersecurity-101/malware/malware-analysis/.

[2]. Sihwail, Rami & Omar, Khairuddin & Zainol Ariffin, Khairul Akram. (2018). A Survey on Malware Analysis Techniques: Static, Dynamic, Hybrid and Memory Analysis. 8. 1662.10.18517/ijaseit.8.4-2.6827. https://www.researchgate.net/publication/328760930_A_Survey_on_Malware_Analysis_Techniques_Static_Dynamic_Hybrid_and_Memory_Analysis.

[3]. Damodaran, Anusha & Di Troia, Fabio & Visaggio, Corrado Aaron & Austin, Thomas & Stamp, Mark. (2017). A comparison of static, dynamic, and hybrid analysis for malware detection. Journal of Computer Virology and Hacking Techniques. 13. 10.1007/s11416-015-0261-z. https://www.researchgate.net/publication/288905288_A_comparison_of_static_dynamic_and_hybrid_analysis_for_malware_detection.

[4]. Chiradeep BasuMallick, “What Is Malware Analysis? Definition, Types, Stages, and Best Practices”, 19 August 2021, Spiceworks. https://www.spiceworks.com/it-security/data-security/articles/what-is-malware-analysis-definition-types-stages-best-practices/.

[5]. Tayyab, U.-e.-H.; Khan, F.B.; Durad, M.H.; Khan, A.; Lee, Y.S. A Survey of the Recent Trends in Deep Learning Based Malware Detection. J. Cybersecur. Priv. 2022, 2, 800-829. https://doi.org/10.3390/jcp2040041.

[6]. A M. Ijaz, M. H. Durad and M. Ismail, "Static and Dynamic Malware Analysis Using Machine Learning," 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, 2019, pp. 687-691, doi: 10.1109/IBCAST.2019.8667136., https://ieeexplore.ieee.org/document/8667136.

[7]. Rabia Tahir, "A Study on Malware and Malware Detection Techniques", Department of Computer Science, Virtual University of Pakistan, https://www.mecs-press.org/ijeme/ijeme-v8-n2/IJEME-V8-N2-3.pdf.

[8]. Shijo, P.V. & Salim, A. (2015). Integrated Static and Dynamic Analysis for Malware Detection. Procedia Computer Science. 46. 804-811. 10.1016/j.procs.2015.02.149. https://www.researchgate.net/publication/276109044_Integrated_Static_and_Dynamic_Analysis_for_Malware_Detection.

[9]. Malware Analysis Techniques - Basic Static Analysis, Nasreddine Bencherchali, https://nasbench.medium.com/malware-analysis-techniques-basic-static-analysis-335a7286a176.

[10]. PE Format, Microsoft Article, 03/06/2023. https://learn.microsoft.com/en-us/windows/win32/debug/pe-format.