Malware Analysis

Allison Felsheim

Technology Department, University of Southern Maine

ITT 460: Capstone

Professor Andrew Hughen

September 23, 2024

Abstract

As the increase in dependency on technology grows, the prevalence of malware also increases, which poses significant threats to individuals and organizations. Therefore, Malware analysis has become a vital field of study that helps in identifying and understanding different cyber-attacks, such as, backdoors, rootkits, trojans, and ransomware. Although, due to a plethora of available tools it is extremely intimidating to a malware analyst in determining the right tool for their success (Lebbie, et al. 2022). The researcher uses key methodologies to test specific tools in malware analysis, focusing on static, dynamic, and hybrid techniques. The setting consists of a lab using VirtualBox with a FlareVM and Remnux virtual machine (VM). The goal aims to evaluate the strengths and weaknesses of various malware analysis tools, providing insights into their capabilities in detecting file system changes, registry modifications, and import detection malware features. By employing a mixed-methods approach, the researcher compares the tools IDA Pro, Regshot, Ghidra, and ProcMon, contributing to a guide that enhances malware detection and defense strategies. Research and experimentation continued to suggest a need for a combination of tools and specific features analyzed to improve upon the knowledge gaps (Higuera et al., 2020; Damodaran et al., 2015). Although the researcher's results indicated the need for future research, an analysis of available results, which the researcher has done in a mixed methods approach, reaffirms the researchers' suspicions, with the integration of hybrid analysis, one can determine the most sufficient tools for every malware feature tested.

Malware Analysis

As society and businesses increasingly depend on technology, malware grows exponentially, posing a significant threat to organizations and individuals alike. Businesses of all sizes are increasingly integrating technology that handles sensitive information, including payment details, banking transactions, personal data, and more. This technological advancement has unfortunately been accompanied by a significant rise in cyberattacks, involving various types of malwares such as backdoors, rootkits, trojans, and ransomware. Consequently, malware analysis has emerged as a critical discipline for researchers and scientists to identify, understand, and mitigate these threats effectively. The following review of literature reinforces that exploring common themes found among malware analysis tools, assessing their strengths to identify the most effective strategies for analyzing and defending against malware attacks contributes to the enhancement of cybersecurity measures in an increasingly digital world.

Literature Review

Malware analysis involves the systematic dissection and examination of malicious software to understand its behavior and intent. This process is crucial for comprehending the threats posed by malware and for developing effective defenses against them. Although, with the abundance of available tools it is extremely intimidating to the malware analyst, making the selection of the right tool pivotal to their success (Lebbie, et al. 2022). Cybercriminals, particularly those operating in the dark web, continuously purchase, modify, and enhance malware, increasing its complexity to evade detection by antivirus solutions. To effectively combat this, it is vital to understand the methodologies employed by these malicious actors. Analyzing malware samples enables researchers to uncover the specific actions and tactics employed by different malware variants, shedding light on how they execute successful attacks

(Wong et al., 2021). Thus, effective malware analysis not only strengthens our ability to detect and respond to current threats but also plays a crucial role in anticipating and mitigating future cyber risks.

Two major themes throughout the project are identified as: static and dynamic analysis (Leon, et al. 2021). Static malware analysis, which involves examining malware without executing it in a runtime environment. While static analysis may not uncover every detail, it can provide critical insights that guide analysts in focusing their subsequent investigative efforts (Monnappa, 2018). Various tools facilitate this method, including IDA Pro, VirusTotal, Ghidra, and VxStream Sandbox, all of which are referenced throughout the project. Moreover, dynamic malware analysis, which entails executing a malware sample in an isolated environment to monitor its activities, interactions, and effects on the system (Monnappa, 2018). Tools such as Process Hacker, Second Look, and Belkasoft support this method. Some methodologies highlighted in the key references advocate for conducting dynamic analysis first, followed by static analysis, to achieve a more accurate understanding of how the code operates (Higuera et al., 2020). Additionally, hybrid analysis, which combines both static and dynamic approaches to create a more diverse framework for malware examination. In this methodology, static analysis is performed initially on the malicious code, followed by dynamic analysis to accurately detect malicious behaviors (Dutta et al., 2022). According to key references, hybrid analysis is notably more effective, as it leverages the strengths of both approaches, enhancing the overall understanding of the malware's impact and intent. In this research, our goal is to deepen our understanding of the relative advantages and disadvantages of static, dynamic, and hybrid approaches (Damodaran, et al. 2015).

A properly configured lab setup is crucial for malware analysis because it provides a safe, isolated environment to study malicious software without risking the analyst's primary system or network. Some of the articles such as *Learning Malware Analysis : Explore the Concepts, Tools, and Techniques to Analyze and Investigate Windows Malware* and *Malware Analysis and Detection Engineering* (Monnappa, 2018) and (Mohanta & Saldanha, 2020) reiterate this importance and provide a step-by-step guide on building and maintaining a safe and effective malware analysis lab, supported by extensive research on their architecture and the application of static and dynamic analysis. In their books, they discover how the internals of malware work and how to analyze and detect it. They also cover classifying and categorizing it, giving insight into the intent of the malware. Malware Analysis and Detection Engineering is a one-stop guide to malware analysis that simplifies the topic by teaching you undocumented tricks used by analysts in the industry. The book starts with an introduction to malware analysis and reverse engineering to provide insight on the different types of malwares along with the terminology used in the anti-malware industry. Following the introduction a guide is instructed on how to set up an isolated lab environment to safely execute and analyze malware, as well as learn about malware packing, code injection, and process hollowing. It is crucial this step of the process is not overlooked, as the lab setup determines the analyst's success in their endeavors.

The validity of this project is supported by a range of key references that provide essential insights into the landscape of malware and its analysis. *Cyber Security Issues and Current Trends* (Dutta et al., 2022) offers a comprehensive overview of malware, detailing its various types and existing detection and analysis methodologies. Additionally, several recommendations emerge from this body of work, highlighting diverse approaches that are pertinent to the project, such as the role of cloud computing in combating cyber malware

(Almomani et al., 2024) and the application of machine learning techniques in static malware analysis (Shalaginov et al., 2018). Further contributions to this field include research that presents unique analysis methods tailored to understanding malware components. For instance, *A Basic Malware Analysis Method* discusses not only the software and hardware systems utilized in studies but also emphasizes the significance of developing feasible and effective analysis strategies (Kara, 2019). These references collectively inform and enhance the framework of this project, providing a robust foundation for further exploration and development in malware detection and analysis.

In conclusion, the examination of malware through diverse methodologies is essential for understanding and mitigating the inherent risks associated with its proliferation. This literature review underscores the necessity of employing a range of analytical techniques to enhance our defenses against ever-evolving cyber threats. The contributions of the existing literature are significant, providing a comprehensive understanding of both static and dynamic analysis methodologies. This body of work lays the groundwork for identifying effective strategies to counteract malware threats. The strengths of these studies lie in their systematic exploration of various analysis techniques, their emphasis on lab setups, and the integration of machine learning and cloud computing in malware detection. Together, these elements enhance our ability to anticipate and respond to cyber threats more effectively.

Objective/Purpose

As malware threats become increasingly sophisticated, many cybersecurity professionals struggle to identify and utilize the most effective malware analysis tools and techniques. This lack of knowledge can lead to inadequate detection and mitigation, leaving organizations vulnerable to attacks. To address this critical issue, there is a pressing need for a detailed guide

that evaluates the strengths, weaknesses, and effectiveness of various malware analysis tools and techniques. Such a resource will empower professionals to make informed decisions and enhance their cybersecurity strategies. However, literature is not without its weaknesses. A notable gap is the insufficient emphasis on the psychological and behavioral aspects of cybercriminals, which could inform more robust defense strategies. Additionally, while tools and methodologies are extensively discussed, there is a lack of empirical evidence demonstrating the real-world efficacy of these approaches in varied operational contexts. This disconnect may hinder the application of findings in practical settings. By examining how these tools perform in various environments and against diverse malware types, researchers can refine their recommendations and enhance the practical applicability of their findings. Moreover, exploring the integration of behavioral analytics and user-centric approaches in malware analysis could yield valuable insights, providing a more comprehensive understanding of the threats posed by cybercriminals. In summary, while the existing literature on malware analysis has made substantial contributions to the field of cybersecurity, addressing its shortcomings and gaps is imperative for the continued advancement of effective defense mechanisms. As cyber threats grow increasingly sophisticated, a concerted effort to evolve research methodologies and frameworks will be crucial in safeguarding our digital landscapes.

The goal of this research paper is to assist cybersecurity professionals in selecting the most suitable tools for their specific tasks at hand. This is accomplished by comparing the detection capabilities of various malware analysis tools in relation to malware features that represent some of the most important qualities of malware. By integrating insights from the literature review with practical evaluations of the tools in action, this study aims to develop a comprehensive guide that reflects the author’s perspectives on the chosen tools and their

techniques. The quantitative component of this study will thoroughly select a diverse array of malware analysis tools based on their prevalence in the market. This selection process is then complemented by the thorough literature review that solidifies the research conducted on static, dynamic, and hybrid analysis methods.

To strengthen the purpose and overall framework of the study, the researcher formulated four key research questions aimed at evaluating the research process. These questions brought the study closer to one of its primary objectives: establishing that the malware feature under investigation aligns with the specific analysis tool being used. The research questions were as follows: (1) What are the strengths and weaknesses of static versus dynamic analysis in identifying malware threats? (2) How effective are various malware analysis tools in detecting specific types of malware? (3) What emerging trends in malware analysis tools and techniques are expected to shape the field in the next five years? (4) How do different types of malware impact the selection of analysis techniques? The insights gained from these questions provided critical information for the researcher, ultimately guiding their conclusions on the optimal tool selection for the study.

By equipping security professionals with both data and insights derived from previous research, this study aims to strengthen current and future analysis techniques. The creation of a comprehensive malware analysis guide is essential for anticipating emerging trends in the field. General inquiries will help clarify how the landscape of malware analysis is expected to evolve in the coming years. Specific questions regarding emerging trends in tools and techniques will offer insights into how new malware influences the selection of analysis methodologies. This approach will yield valuable data on the techniques employed in current research. Questions regarding emerging trends in tools and techniques will provide insights into how new malware

influences the selection of analysis methodologies. This approach will yield valuable data about specific techniques employed in current research.

While the methodology is designed to facilitate a thorough assessment, it is important to acknowledge potential limitations. Biases in tool selection and the inherent variability in malware behavior during testing could influence the results. Recognizing these limitations will help ensure the validity and reliability of the study's conclusions.

This research aims to create a detailed guide on malware analysis tools and techniques, empowering cybersecurity professionals to make informed decisions in their strategies. By addressing key research questions and employing a rigorous mixed-methods approach, this study seeks to provide valuable insights into the effectiveness of various tools, ultimately contributing to improved cybersecurity practices. The methods section will elaborate on how this study will address all the questions outlined in the objectives, utilizing data obtained from both research findings and a foundational literature review.

Methods and Techniques

Scope of Methods Analysis

This study utilizes VirtualBox to create a malware analysis lab. The lab consists of a FlareVm and Remnux VM. Remnux acts as a simulated internet environment by running InetSim while the research on the malware is conducted in FlareVm. The literature review prompts the organizational baseline for conducting the analysis. A mixed methods approach is deployed in this research, relying on expert insight to assess the most popular tools, techniques, and malware features. The author has carefully curated malware samples for tool performance analysis, drawing on insights from existing literature (Damodaran et al., 2015; Dutta et al., 2022). Three types of malwares were selected: WannaCry ransomware, Remote Access Trojan (RAT), and

BackDoor malware, sourced from the TCM Security repository. The malware samples were used to assess tool capabilities regarding file system changes, registry modifications, and import detections.

Method 1

The first method employed in this study is static analysis, which involves examining the malware without executing it. This technique allows for the identification of potential malicious behavior by analyzing file properties, structures, and signatures. Research indicates that static analysis is effective in detecting known malware through signature databases. Uncovering clues regarding the nature of the malware, such as filenames, hashes, IP addresses, domains, and file header data is crucial for malware analysis research.

For this analysis IDA Pro and Ghidra are utilized to monitor file system changes, registry modifications, and import detection. Additionally, entropy analysis will be performed to detect packed or encrypted malware segments. All analyzed data will be documented and summarized in a structured format for clarity. The results of this analysis will provide foundational insights into the malware's capabilities.

While static analysis is valuable for detecting known malware and identifying potential vulnerabilities, it has limitations. It may miss runtime behaviors or attacks that are triggered only during execution, such as file manipulation or network communications. Additionally, sophisticated malware may employ obfuscation techniques, which could hinder accurate signature detection. Static analysis also does not account for evasive malware behavior, such as malware that alters its behavior in response to the analysis environment.

Method 2

The second method will incorporate dynamic analysis, which entails executing the malware. This approach enables researchers to observe real-time behavior, including system interactions, file manipulations, and network communications. Studies have shown that dynamic analysis is crucial for understanding the full scope of malware behavior. This methodology reveals critics about its functionality, attack methods, and potential impacts that might not be apparent through static analysis alone.

For this analysis, ProcMon, and RegShot are utilized to monitor file system changes, registry modifications, and import detection. The dynamic analysis result complements the static findings, revealing runtime behaviors that static methods may overlook. All analyzed data will then be documented and summarized, as well as compared with method one's results.

Dynamic analysis provides valuable insights into malware behavior, but its subject to limitations as well. Malware can potentially detect the analysis environment and alter its behavior to avoid detection. Additionally, some malware employs anti-debugging techniques that can prevent accurate observation during analysis. Moreover, dynamic analysis may not fully capture all potential attack vectors if the malware is highly evasive or has delayed execution mechanisms.

Method 3

The final method is hybrid analysis, which integrated both static and dynamic approaches for a comprehensive examination of the malware. By combining these techniques, researchers can validate static findings with dynamic behaviors, leading to a deeper understanding of the malware's functions. This hybrid approach will begin with static examination, followed by the execution of the malware. Findings for both analyses will be cross-referenced to identify correlations and discrepancies in behavior. This method provides the needed holistic view of

malware, assisting in the identification of sophisticated evasion techniques and advanced functionalists.

Hybrid analysis and cuckoo sandbox are utilized for method three. Hybrid analysis is a free service owned by CrowdStrike. Hybrid analysis uses Falcon Sandbox, an automated malware analysis solution. Additionally, Cuckoo sandbox is an automated malware analysis tool that provides knowledge on how the malware operates in order to understand the context, motivations, and goals of the breach. Both tools utilized during this method provide their own detailed report outlining the behavior of the file when executed. As a result, all documents provided by each methodology will be added to a final product that compares the strengths and weaknesses of the tools and their techniques.

Hybrid analysis combines the benefits of both static and dynamic analysis; however, it also has its limitations. Automated analysis tools like Falcon Sandbox and Cuckoo Sandbox may not always detect novel or sophisticated malware, particularly if it employs advanced evasion techniques. Moreover, hybrid analysis might not fully replicate a real-world environment where the malware could behave differently due to system configurations, user actions, or external network interactions. Furthermore, relying on automated tools may result in an incomplete understanding of malware behavior if the tools fail to capture nuanced or context-dependent actions.

The research design aligns well with the purpose of the study: analyzing malware behavior and evaluating the effectiveness of analysis tools. The use of a mixed-methods approach, incorporating static, dynamic, and hybrid analyses, allows for a comprehensive evaluation of the malware's characteristics. Static analysis uncovers fundamental properties and signatures, dynamic analysis captures real-time behaviors, and hybrid analysis provides a

cross-validation of both approaches, ensuring robust findings. Each method complements the others, providing a deeper understanding of malware behavior and tool performance. Quality assurance in this study is maintained through the careful selection of reliable, widely used tools and a curated set of malware samples that ensure validity and reliability of the results. To further ensure accuracy, the results from each method will be cross-validated and compared against one another.

Data Collection and Analysis

Data collection templates were utilized to log results at each phase of the analysis. Malware features examined in this study include file system changes, registry modifications, and import detections. According to previous studies, these are crucial indicators of malware behavior (Kara, 2019; Mohanta & Saldanha, 2020). Despite time constraints limiting the analysis to three malware samples, the collected data helped assess tool effectiveness across key features.

Samples	System File Changes	Registry Changes	Import Detection
WannaCry	✔	✔	✔
Remote Access Trojan	✔	✔	✔
BackDoor	✔	✔	✔

Tools	System File Changes	Registry Changes	Import Detection
Procmon	✔	✔	✔
IDA Pro	✔	□	✔
Ghidra	✔	□	✔
Regshot	□	✔	□

Data Analysis

A mixed-methods approach to data analysis includes thematic analysis to identify the most widely recognized tools based on professional insights (Leon et al., 2021). Following this, the author's research evaluates tool effectiveness in relation to specific malware features. Comparative analysis is conducted to determine which tool best detects file system changes, registry interactions, and dynamic imports during malware execution. As suggested in previous works, such as those by Higuera et al. (2020), effective malware analysis often requires a combination of static and dynamic analysis methods.

Preliminary Findings

The analysis of the selected malware samples yielded several key insights into the effectiveness of various malware analysis tools. These tools were evaluated based on their ability to detect system file changes, registry modifications, and import detections. The findings indicate the following:

File System Changes

Procmon: Procmon was highly effective at detecting file system changes across all three malware samples. It captured file creation, deletion, and modification events in real-time, providing crucial insights into how the malware interacts with the file system. This is consistent with previous findings by Leon et al. (2021) that emphasize Procmon's strengths in dynamic analysis.

IDA Pro: While IDA Pro detected system file changes in WannaCry by identifying Windows API functions such as CreateFileA and MoveFileExA, the obfuscation in WannaCry made detection more challenging. This is in line with Damodaran et al. (2015), who noted the

limitations of static analysis when dealing with obfuscation techniques.

Ghidra: Like IDA Pro, Ghidra could detect system file changes in WannaCry, but its ability to track real-time changes was limited. This finding supports Kara (2019), who highlighted the gap between static and dynamic analysis tools.

Registry Changes

Procmon: Procmon was effective in detecting registry changes across all samples, capturing modifications related to persistence mechanisms, configuration changes, and system settings. This aligns with findings in the literature (Monnappa, 2018) emphasizing Procmon's strengths in live monitoring.

IDA Pro: For the BackDoor sample, IDA Pro identified registry-related functions like internall_syscall_windows_registry_Key_getvalue, but static analysis tools struggle with real-time monitoring. This confirms conclusions from previous studies, such as those by Shalaginov et al. (2018), which noted the limitations of static analysis for dynamic behaviors like registry interactions.

RegShot: RegShot excelled in tracking registry changes, providing a before-and-after comparison of the registry state. However, as noted by Dutta et al. (2022), its functionality is limited to registry analysis, making it less useful for detecting other features such as file system changes or dynamic imports.

Import Detection

Procmon: Procmon was highly effective in capturing dynamic imports, particularly in WannaCry, which used GetProcAddress to dynamically resolve function addresses. This supports findings by Leon et al. (2021), who highlighted Procmon's capabilities in tracking system calls and dynamic library interactions.

IDA Pro: IDA Pro could identify imports like GetProcAddress in WannaCry but struggled to track dynamic imports during execution. As highlighted by Shalaginov et al. (2018), static analysis tools face challenges in capturing runtime resolution of imports.

Ghidra: Like IDA Pro, Ghidra could identify static imports but could not effectively capture dynamic imports in real-time, particularly in RAT and BackDoor samples. This highlights the limitations of static analysis tools for detecting dynamically resolved imports (Kara, 2019).

RegShot: RegShot did not capture import activities, as it is limited to registry analysis (Mohanta & Saldanha, 2020).

While the tools used in this study were effective for detecting certain malware behaviors, there are several limitations. The study only used three malware samples, which limits the generalizability of the findings. A larger sample size would provide stronger conclusions and allow for a more comprehensive tool comparison. Additionally, due to time constraints, only a small subset of features (file system changes, registry modifications, and import detections) were analyzed, leaving out other potentially significant aspects of malware behavior, such as network activity or memory analysis. Moreover, each tool has its own limitations: ProcMon excels in dynamic analysis but is unable to detect certain static behaviors, while static tools like IDA Pro and Ghidra struggle with dynamic or obfuscated malware. These limitations suggest that a combination of tools is necessary for comprehensive malware analysis, as supported by Higuera et al. (2020). Further testing with a larger sample set will help refine the tool selection process and confirm these preliminary results.

Results and Conclusion

Results

Malware analysts have a hard time deciphering between the many tools available based on the specific malware features at hand. With this wide range of tools, clearly determining which tools to use determines their success. The purpose of this project is to explore some of the most popular tools available and determine their success at detecting important malware features.

Based on the findings from this research and the data collected, figure 1 shows a breakdown that summarizes the effectiveness of the various malware analysis tools for detecting file system changes, registry changes, and import activities.

Feature	Procmon	IDA Pro	Ghidra	Regshot
System File Changes	Highly Effective	Effective (with some limitations in obfuscation)	Effective	Not Supported
Registry Changes	Highly Effective	Effective	Effective	Highly Effective (focused on registry)
Import Detection	Effective	Highly Effective	Highly Effective	Not Supported

(Figure 1: effectiveness of the various malware analysis tools for detecting file system changes, registry changes, and import activities).

The first feature tested to determine the tools success was system file changes. Procmon was the most effective tool, capturing file creation, modification, and deletion. This performance was consistent across all malware samples tested. While IDA Pro and Ghidra proved useful for

static analysis, they were less capable in tracking file system changes, especially when dealing with obfuscation techniques, such as those used by malware like WannaCry. Regshot, on the other hand, was ineffective for file system analysis, as its focus is solely on registry changes.

The next feature tested was registry changes. Procmon proved to be the most effective tool, offering robust detection of registry modifications, especially those associated with malware persistence and configuration. Both IDA Pro and Ghidra also demonstrated strong capabilities in identifying static registry modifications. Regshot, which specializes in detecting registry system changes, excelled at monitoring registry modifications and provided clear before and after snapshots of registry states. Although, its narrowed focus made it unsuitable for broader malware analysis, as it only tracked registry activity.

The last feature tested was import detection. IDA Pro and Ghidra excelled at detecting imports and were more user friendly for this feature. For the dynamic tools, Procmon was also effective at detecting imports, although less user friendly. Regshot was not capable of detecting import activities, as it focused exclusively on the registry.

Procmon stood out as the most versatile tool, demonstrating high effectiveness across all three malware features, including obfuscation techniques. Regshot, while highly effective for registry analysis, was too narrow in its focus to be a comprehensive tool for malware analysis. Both IDA Pro and Ghidra performed well for static analysis but were less effective when it involved obfuscation techniques from malware like WannaCry.

Data Analysis

The findings align with the literacy review, which emphasizes the need for both static and dynamic analysis techniques to fully understand malware behavior (Damodaran et al., 2015; Leon et al., 2021). These results reinforce the idea that a hybrid approach is the most effective

way to analyze malware. The data suggests that while individual tools may excel in certain areas, no single tool can provide a comprehensive analysis of all malware behaviors. A hybrid strategy, leveraging the strengths of multiple tools, offers the best chance of success in malware detection.

Conclusion

This study highlights the strengths and limitations of several prominent malware analysis tools. Procmon proved to be the most effective overall, excelling at detecting file system changes, registry modifications, and dynamic imports. IDA Pro and Ghidra were also powerful but struggled when highly persistent obfuscation techniques were used in the malware tested. Regshot, though highly effective for registry monitoring, was too limited in scope to offer a complete view of malware activity. Moreover, these findings support the growing body of research advocating for a hybrid approach to malware analysis, combining both static and dynamic techniques. While Procmon emerged as the standout tool, the study underscores the importance of selecting the appropriate tool based on the specific features being analyzed. Future research should involve testing a broader range of malware samples to validate these findings and refine the tool selection process. Future research should also explore the integration of hybrid analysis techniques in order to determine the most sufficient tools for every malware feature tested.

References

Almomani, I., Maglaras, L., Ferrag, M., Ayres, N. (2024). *Cyber Malware*. Security Informatics and Law Enforcement. https://doi.org/10.1007/978-3-031-34969-0

Damodaran, A., Troia, F. D., Visaggio, C. A., Austin, T. H., & Stamp, M. (2015). A comparison of static, dynamic, and hybrid analysis for malware detection. *Journal of Computer Virology and Hacking Techniques*, *13*(1), 1–12. https://doi.org/10.1007/s11416-015-0261-z

Dutta, N., Jadav, N., Tanwar, S., Sarma, H. K. D., & Pricop, E. (2022). Cyber Security: Issues and Current Trends. In *Studies in Computational Intelligence*. Springer Singapore. https://doi.org/10.1007/978-981-16-6597-4

Higuera, J., Aramburu, C., Higuera, J.-R., Urban, M. A., & Montalvo, J. A. (2020*). Systematic Approach to Malware Analysis (SAMA)*. Applied Sciences, 10(4), 1360. https://doi.org/10.3390/app10041360

Kara, I. (2019). *A basic malware analysis method.* Computer Fraud & Security, 2019(6), 11–19. https://doi.org/10.1016/s1361-3723(19)30064-8

Lebbie, M., Prabhu, S. R., & Agrawal, A. K. (2022). Comparative Analysis of Dynamic Malware Analysis Tools. *Algorithms for Intelligent Systems*, 359–368. https://doi.org/10.1007/978-981-16-5747-4_31

Leon, R. S., Kiperberg, M., Leon Zabag, A. A., & Zaidenberg, N. J. (2021). *Hypervisor-assisted dynamic malware analysis*. Cybersecurity, 4(1). https://doi.org/10.1186/s42400-021-00083-9

Monnappa, A. (2018). *Learning Malware Analysis : Explore the Concepts, Tools, and Techniques to Analyze and Investigate Windows Malware*. Packet Publishing Ltd. https://libribook.com/ebook/12754/learning-malware-analysis-pdf/?bookid=45368

Mohanta, A., & Saldanha, A. (2020). *Malware Analysis and Detection Engineering*. Apress. https://doi.org/10.1007/978-1-4842-6193-4

Shalaginov A., Banin, S., Dehghantanha, A., Franke, K. (2018). *Cyber Threat Intelligence*. "Machine Learning Aided Static Malware Analysis: A Survey and Tutorial." vol. 70. Switzerland: Springer International Publishing AG, 7–45. Web.

Wong, M., Landen, M., Antonakakis, M., Blough, D., Redmiles, E., & Ahamad, M. (2021). *An Inside Look into the Practice of Malware Analysis.* Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. https://doi.org/10.1145/3460120.3484759