OPEN ACCESS

Saudi Journal of Engineering and Technology
Abbreviated Key Title: Saudi J Eng Technol
ISSN 2415-6272 (Print) |ISSN 2415-6264 (Online)
Scholars Middle East Publishers, Dubai, United Arab Emirates
Journal homepage: https://saudijournals.com

Original Research Article

Virtual Machines vs. Containerized Environments: A Comparative Study for Malware Analysis

Gideon Emmanuel Oki[1*], Emmanuel Wemogene Sadiq[2]

[1]Independent Researcher, Jos, Nigeria
[2]Independent Researcher, Lagos, Nigeria

DOI: https://doi.org/10.36348/sjet.2025.v10i04.011 | **Received:** 03.03.2025 | **Accepted:** 09.04.2025 | **Published:** 26.04.2025

***Corresponding author:** Gideon Emmanuel Oki
Independent Researcher, Jos, Nigeria

Abstract

Malware analysis is critical to cybersecurity because it enables researchers and security professionals to better understand threats, their indicators of compromise, and then provide mitigating measures. This paper presents the comparative analysis of two popular malware analysis environments, i.e, Virtual Machines (VMs) and Containerized Environments (CEs). The team evaluated these platforms based on key factors such as isolation, resource usage, startup time, scalability, operating system support and emulation capabilities. Our findings reveal that Virtual Machines offer stronger isolation and better operating system emulation, while Containerized Environments provide faster startup time, better scalability and a lower resource overhead. This study provides a valuable insight for cybersecurity professionals seeking to choose the most suitable environment for malware analysis.

Keywords: Malware Analysis, Virtual Machines, Containerized Environments, Cybersecurity.

Copyright © 2025 The Author(s): This is an open-access article distributed under the terms of the Creative Commons Attribution **4.0 International License (CC BY-NC 4.0)** which permits unrestricted use, distribution, and reproduction in any medium for non-commercial use provided the original author and source are credited.

I. INTRODUCTION

Malware which is short for malicious software, is any program or code intentionally designed to cause harm to computers, networks, or digital devices. Malware comes in various forms, including viruses, worms, Trojans, ransomware, spyware, and adware, each with different methods of attack. Some malware is designed to steal sensitive data, such as login credentials, financial details, or personal information. Others aim to corrupt files, disrupt systems, or gain unauthorized control over devices, often to be used in larger cyberattacks like botnets. Malware spreads through a variety of means, including deceptive email attachments, compromised websites, malicious downloads, and even infected external devices. To protect against malware, users should practice cyber-hygiene, safe browsing, keep software updated, and use reliable cybersecurity tools to detect and prevent threats before they cause damage.

Malware analysis is the process of examining and dissecting malware with the aim of understanding its behavior, purpose, and potential impact on infected systems. This analysis helps researchers and cybersecurity professionals identify how malware operates, uncover its origin, and determine the best ways to defend against it. The primary objective of malware analysis is to develop effective countermeasures, such as antivirus signatures, indicators of compromise (IOCs), and mitigation strategies to reduce the risks posed by malicious software.

Malware analysis plays a crucial role in strengthening cybersecurity by enabling researchers and security experts detect, prevent, and neutralize threats before they cause significant harm.

There are different approaches to malware analysis, these include static and dynamic analysis. Static analysis involves examining the malware's code without executing it, while dynamic analysis observes the malware's behavior in a controlled environment. By understanding malware, security professionals and researchers can enhance defense mechanisms, improve threat intelligence, and protect networks and systems from evolving cyber threats.

To perform malware analysis safely, researchers require isolated environments that prevent

malware from affecting the host system [1]. Two common approaches used for creating such environments are: Virtual Machines (VMs) and Containerized Environments (CEs).

A. Why is Malware Analysis Important

Malware analysis is important for several reasons, as it plays a crucial role in understanding and effectively mitigating cyber threats. Malware analysis is a critical aspect of cybersecurity that enables security professionals and researchers to identify and understand threats, develop effective countermeasures, improve their security posture, and maintain regulatory compliance. Some key reasons include:

- **Threat Identification and Understanding:** Malware analysis is crucial for identifying and understanding emerging or new threats. It allows security professionals and researchers to detect new variants of malware, analyze their behavior, objectives, and attack methods. Gaining this insight helps assess the potential risks and impact malware can have on infected systems and networks. It also helps in enabling effective countermeasures.
- **Development of Countermeasures:** Malware analysis plays a vital role in developing effective countermeasures to combat cyber threats. Security professionals and researchers use the insights gained from analyzing the malware to create antivirus signatures, behavioral detection rules, and indicators of compromise (IOCs). These countermeasures help in identifying, preventing, and eliminating malware infections, ultimately strengthening system security and reducing vulnerabilities.
- **Incident Response and Remediation:** Malware analysis is essential for effective incident response and remediation. When a security breach or malware infection occurs, analyzing the malware helps incident response teams understand its behavior, purpose, and impact. This knowledge helps the team to implement targeted remediation strategies, contain the threat, and restore affected systems more efficiently, thereby minimizing the potential damage and downtime.
- **Enhancing Cybersecurity Defenses:** Malware analysis plays a crucial role in strengthening cyber defenses by helping organizations identify vulnerabilities in their security infrastructure. By studying malware techniques and attack methods, organizations can enhance their security measures, implement stronger protective strategies, and deploy advanced security controls to reduce the risk of future cyber threats and attacks and improve their posture.
- **Cyber Threat Intelligence:** Malware analysis plays a vital role in cyber threat intelligence by expanding the security community's knowledge of emerging threats. It provides professionals and researchers insight into the tactics, techniques, and procedures (TTPs) used by cybercriminals. By sharing this intelligence, organizations can stay updated on evolving threats and adjust their security strategies to enhance protection against attacks.
- **Legal and Regulatory Compliance:** Malware analysis is essential for organizations to comply with legal and regulatory cybersecurity requirements. By continuously monitoring, analyzing, and mitigating threats, organizations can demonstrate their commitment to data protection and security. This proactive approach helps ensure compliance with regulations such as GDPR, HIPAA, and PCI DSS, reducing legal risks and enhancing overall cybersecurity resilience.

Malware analysis tools are specialized software and utilities designed to aid researchers and security professionals in dissecting and understanding malicious software (malware). These tools can help streamline the analysis process and provide valuable insights into the malware's behavior, functionality, and potential impact. Some common malware analysis tools include: Disassemblers and Debuggers, Sand-boxing and Automated Analysis, Network Traffic Analysis, Analysis (Static and Dynamic), Behavioral Analysis, De-obfuscation and Unpacking amongst others.

B. Problem Statement

Virtual Machines (VMs) and Containers are widely used for malware analysis, but differ significantly in terms of isolation, resource efficiency, and operational flexibility. Understanding how they differ is crucial for cybersecurity professionals to select the most appropriate environment for their needs. This work seeks to present the two platforms and examine them, giving recommendations to help researchers make a more informed decision when choosing a platform for malware analysis.

C. What is the Aim of This Work?

This project aims to compare and contrast two malware analysis platforms. Specifically, the team would be looking into Virtual Machine options, their benefits and pitfalls while comparing them to containerized options with their associated benefits and failures as well. In the end, we will present our findings based on the tests we performed.

II. BACKGROUND AND RELATED WORK

This research relies on previous work on similar issues especially works that touch on virtual machines and containers [2]. Previous studies carried out numerous experimental assessments which this research looked into and also attempted to replicate. It should be noted however, that we will do our best not to emulate what they did but take a unique approach to ours while still using theirs as a sort of guide. Some papers explore the option of using container-based environments posing as a good alternative to the regular hypervisor-based platforms [3]. While some other studies helped us clearly outline the differences between both approaches weighing their pros and cons [4]. These past works

informed some of the metrics we picked [5], while some although they don't have a direct link to our interest area, they still came in handy [6].

A. Gaps this Work Addresses

Having gone through previous literature on this subject, this study has not come across any work in the community which offers researchers insight into the strengths and weaknesses of both platforms used for malware analysis. This study seeks to bridge this gap and help researchers make a more informed decision based on the current standards available.

III. METHODS

For the methodology, the team will be creating two virtual environments. On the first, we will have FlareVM installed and on the second we will have an Ubuntu Machine with Docker Installed and then we will have our containerized solution running there.

A. Virtual Machine (VM) Setup

The team used FlareVM which is a Windows-based malware analysis environment, as our virtual machine platform. The set-up involved follows these steps:

- Provisioning a Windows OS on a virtual machine.
- Disabling Windows updates and defender to prevent interference.
- Installing FlareVM using PowerShell scripts and following the official documentation.

B. Containerized Environment Setup

For the containerized environment, the team used REMnux which is a Linux-based malware analysis toolkit, deployed via Docker and visualized using Kasm Workspaces. The set-up followed the following steps: • Installing Docker on the Ubuntu Machine.

- Deploying REMnux as a container using Kasm

Workspaces

- Configuring the environment for malware analysis.

C. Malware Analysis Tools

The team then analyzed the same malware obtained from a publicly accessible repository. Tools such as PEStudio, FileInsight, Ghidra and DetectItEasy were used.

IV. COMPARATIVE ANALYSIS

The team settled for a few metrics which include; isolation, resource usage, startup time, scalability, operating system support and emulation capabilities.

A. Isolation

Virtual Machines:

VMs provide us with a stronger isolation since they run their own separate OS with a hypervisor which reduces the risk of malware compromising the host system. Containerized Environments: CEs have less isolation as the containers share the host OS and its kernel which makes them more vulnerable.

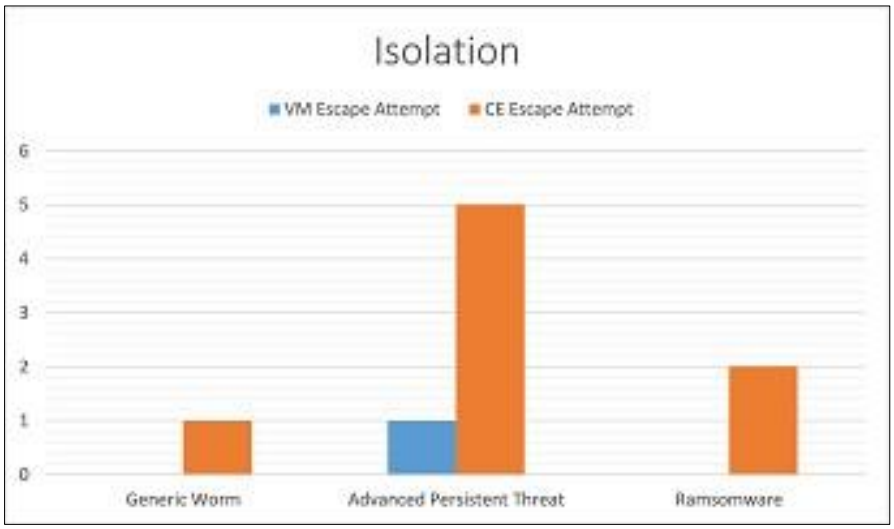

Fig. 1: Isolation Comparison - malware escape attempt

B. Resource Usage

Virtual Machines:

VMs are resource-intensive since they run an OS which results in higher resource consumption including memory, storage and CPU.

Containerized Environments: CEs are more resource efficient as they share the host OS which leads to lower overhead compared with VMs.

C. Startup Time

Virtual Machines:

VMs have a longer startup time since they need to load a full OS which could slow down the analysis. Containerized Environments: CEs are startup faster due to them being lightweight in nature and sharing the host OS allowing for a rapid deployment and analysis.

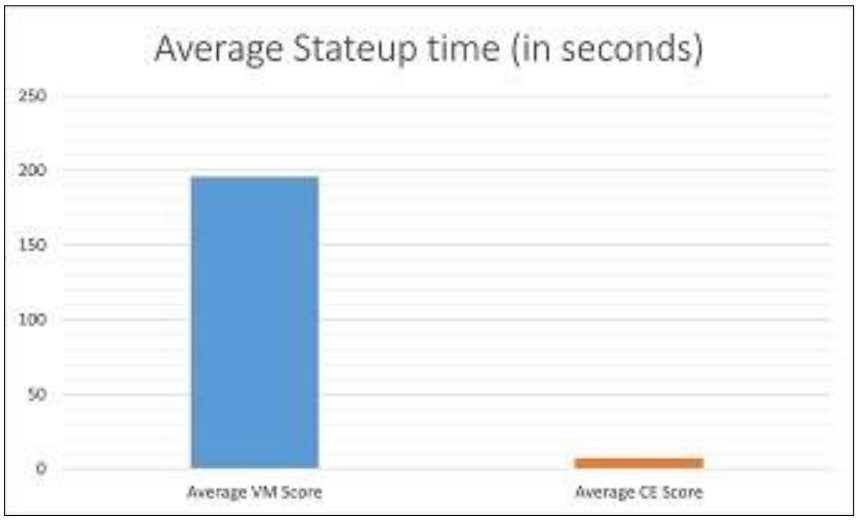

Fig. 2: Comparison of the average startup time in seconds

D. Scalability

Virtual Machines: VMs may not scale efficiently due to the resource intensive nature and longer startup time.

Containerized Environments: CEs offer better scalability as they can easily be replicated and distributed across different nodes.

E. Operating System Support

Virtual Machines: VMs can emulate different OS making them suitable for analyzing a wider range of malware.

Containerized Environments: CEs typically only support Linux-based systems which limits their applicability for analyzing malware targeting other OS.

Table I: Os Support Comparison

Operating System	VM Support	CE Support
Windows 10	Yes	Limited
Windows 7	Yes	No
Ubuntu 22.04	Yes	Yes
Alpine Linux 3.21.3	Yes	Yes
MacOS Big Sur 11.7	Yes	No

F. Emulation

Virtual Machines: VMs can emulate the targeted operating system a lot better compared to containers and as such provide us better results of analysis.

Containerized Environments: Containers may not fully emulate the targeted OS or environment of a specific malware which could potentially cause the malware to act differently than it would on the targeted system.

V. RESULTS AND DISCUSSION

For this part, the team will be looking at the comparisons between the two approaches weighing the pros and cons.

A. Virtual Machines (VMs)

Pros

- Virtual machines provide a better isolation given they run on a separate OS, reducing the risk of malware compromising the host system.
- Virtual machines can emulate various operating systems, making them ideal for analyzing a broader range of malware threats.
- Virtual machines support snapshot functionality, allowing analysts to save and quickly revert to a previous state during analysis.
- Virtual machines can fully emulate the targeted OS and environment of specific malware thereby enabling better analysis.

Cons

- Virtual machines consume more system resources since they run an entire operating system, making them significantly slower than containers.
- Virtual machines have a longer startup and configuration times, resulting in slower performance compared to containerized environments.
- Virtual machines have some sophisticated malware samples that can detect if they are being executed in a virtual environment and will in-turn alter their behavior to avoid, alter, hinder the analysis process.
- Managing multiple VMs for malware analysis can be complex and time-consuming compared to containerized environments.

B. Containerized Environments

Pros

- Containerized environments utilize the host operating system, making them more resource-efficient compared to virtual machines.

- Containerized environments initiate quickly enabling rapid deployment and quicker malware analysis.
- Containerized environments offer better scalability than VMs as they can be easily replicated and distributed across different node points.
- Containerized environments offer streamlined management and version control of the analysis environment, making it easier to maintain and update the analysis tools and libraries.

Cons

- Containerized environments share the same host OS making them more vulnerable to malware that can exploit the host.
- Containerized environments typically only have support for Linux-based systems with limited applicability for other operating systems.
- Containerized environments may not accurately replicate the intended OS or system configuration, potentially causing malware to behave differently than it would in its actual target environment.
- Containerized environments offer simplicity and flexibility, which could lead to misconfigurations that can increase security risks and potential system compromises.

VI. RECOMMENDATIONS

Here are some recommendations by the team;

- Use Virtual Machines where strong isolation is required and when analyzing malware that targets multiple operating systems.
- Use Containerized Environments when rapid deployment is needed or when scalable analysis is required.
- For complex tasks, consider a hybrid approach.

VII. CONCLUSION

This research aimed to compare Virtual Machines and Containerized Environments for malware analysis, highlighting the advantages and limitations of each. Virtual Machines provide better OS emulation and isolation, while Containerized Environments offer faster startup times and improved scalability. The choice depends on specific malware analysis needs. Future studies could explore hybrid approaches and the impact of emerging technologies on malware analysis techniques.

REFERENCES

1. R. Khalimov, M. Benahmed, R. Hussain, S. M. A. Kazmi, A. Oracevic, F. Hussain, and C. A. Kerrache, “Container-based sandboxes for malware analysis: a compromise worth considering,” Proc. 12th IEEE/ACM Int. Conf. Utility Cloud Comput., pp. 219-227, 2019. [Online]. Available: https://dl.acm.org/doi/10.1145/3344341.3368810.
2. H. Aqasizade, E. Ataie, and M. Bastam, “Experimental assessment of containers running on top of virtual machines,” arXiv preprint arXiv:2401.07539, 2024. [Online]. Available: https://arxiv.org/abs/2401.07539.
3. S. Alam, R. N. Horspool, I. Traore, and I. Sogukpinar, “A framework for metamorphic malware analysis and real-time detection,” Comput. Secur., vol. 48, no. C, pp. 212–233, Feb. 2015, doi: 10.1016/j.cose.2014.10.011.
4. Aqua Security, ”Vm vs. container: what’s the difference?,” Aqua Cloud Native Academy. [Online]. Available: https://www.aquasec.com/cloudnative-academy/docker-container/vm-vs-container/. [Accessed: 20-102024].
5. L. Chaufournier, P. Sharma, P. Shenoy, and Y. C. Tay, ”Containers and virtual machines at scale: a comparative study,” University of Massachusetts Amherst and National University of Singapore, 2018.
6. S. Deochake, S. Maheshwari, R. De, and A. Grover, “Comparative study of virtual machines and containers for devops developers,” arXiv preprint arXiv:1808.08192, 2018. [Online]. Available: https://arxiv.org/pdf/1808.08192.
7. Docker, ”What is a container?,” 2021. [Online]. Available: https://www.docker.com/resources/what-container. [Accessed: 2010-2024]
8. C. Feng et al., ”Secbox: a lightweight data mining platform for dynamic and reproducible malware analysis,” 2024 11th IEEE Swiss Conference on Data Science (SDS), Zurich, Switzerland, 2024, pp. 62-67, doi: 10.1109/SDS60720.2024.00017.
9. E. Debas, “Unveiling the dynamic landscape of malware sandboxing: a comprehensive review,” International Journal of Advanced Computer Science and Applications, vol. 15, no. 3, 2024.