IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 2, February 2022

DOI: 10.17148/IJARCCE.2022.11240

A Thin Hypervisor Assisting Threat Hunt Based on Behavioral Observation

Geetha G[1], Shanthi Bala P[2]

[1] M. Tech– Network and Information Security, Pondicherry University, Puducherry

[2] Assistant Professor, Department of Computer Science, Pondicherry University, Puducherry

Abstract: Reverse-engineering malware is an important task in cyber security. The thesis presents a method for malware analysis which helps to detect the possible threats and by-pass vulnerabilities using hypervisor-based method. The purpose of this study is to develop a thin hypervisor with a monitoring component to provide a data security and protection to the host system. Also, the thesis aims to provide a behavioral based malware analysis on a malware lab to hunt for various malware with evasion resistance. The thin protective hypervisor aims to analyze the threat behavior and mitigate those using innovative monitoring component with better performance, transparency, kernel integrity and scalability.

Keywords: Hypervisor, Virtual Machine, Monitoring Component, Malware, Virtualization, Cyber Security

1. INTRODUCTION

Malware is a piece of code, to carry out cyberattacks on an internet either for political or financial interest. Cyber-attacks have been analysed using chain-kill model as shown in Figure 1, to target the victim's machine to compromise on confidentiality, integrity, availability of data and resources. Initial malware activities include defacing graffiti on an organization website, file infectors spread via floppy disk, brain (stealth virus), Jerusalem (DOS virus) and so on. The threat landscape has changed dramatically and new threats have risen drastically. Malwares are fully grown into a giant cyber-crime like DDOS attack infecting prominent websites and IOT, ransomware attacks (WannaCry and Thanatos).
A sandbox environment is an isolated virtual machine in which potentially unsafe software code can execute without affecting network resources or local applications.

Cybersecurity researchers use sandboxes to run suspicious code from unknown attachments and URLs and observe its behaviour. The attack on organization's infrastructure emerged with ransomware, fake firewalls and antivirus software tools are more vulnerable and sensitive. There arises a need for malware analysis for identifying the attacks in a sandbox environment.

The malware can be analysed by two basic approaches: static and dynamic malware analysis. Static malware analysis involves examining any given malware sample without actually running or executing the code whereas dynamic malware analysis involves analysis while running the code in a controlled closed, isolated virtual environment and then its behavior studied. The evasion made by intelligent malwares to conceal their identity irrespective of sophisticated static malware analysis tool.

The researchers have developed various malware analysis techniques to detect and bypass anti-analysis using behavioral based analysis or dynamic analysis monitors. There are other malware analysis methods like fuzzing, symbolic execution and concolic execution. Malware analyst recommends dedicated environment to process the behavioral based analysis tools. Safely designed environment is important to carry out the execution of malware in an analysis system, to avoid the damage. Malware analysis environment is tabulated in Table 1.

Hypervisor based approaches are efficient because it uses introspection tools with greater privilege than malware as it can be hidden from malware. Data security and protection under non-trusted data in VMs can be easily exploited by attackers to initiate various advanced attacks such as a stealthy rootkit, trojan, and regular DoS and DDoS against those VMs. These attacks take place under hypervisor level leads to malfunctioning of host OS and even in cloud platform. Most of the cloud level intrusions happens through hypervisor. There are two hypervisors: "Type 1" (or "bare metal") shown in Figure 2.1 and "Type 2" (or "hosted") shown in Figure 2.2. It is found that every day, the AV-TEST Institute registers over 450,000 new malicious programs (malware) and potentially unwanted applications (PUA). These are examined and classified according to their characteristics and saved. Statistical report chart shown in Figure 3 [21].

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 2, February 2022

DOI: 10.17148/IJARCCE.2022.11240

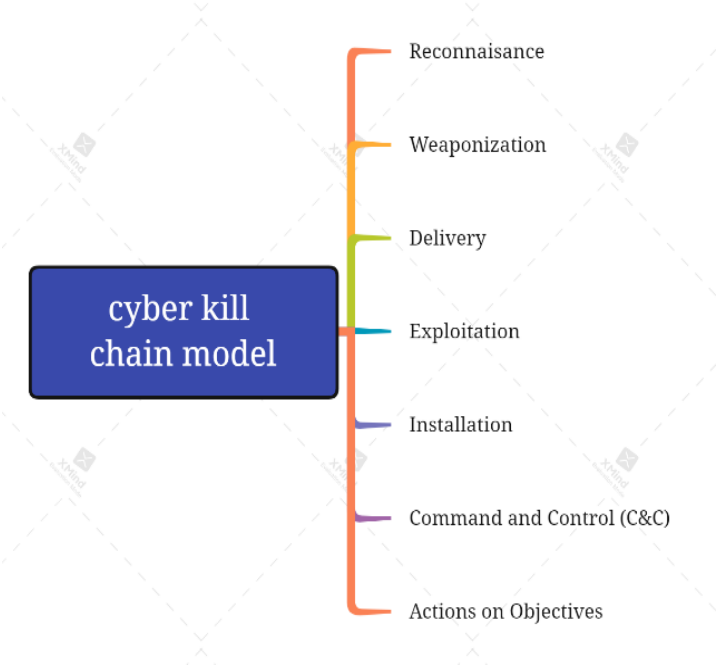

Fig 1. Cyber Kill Chain Model

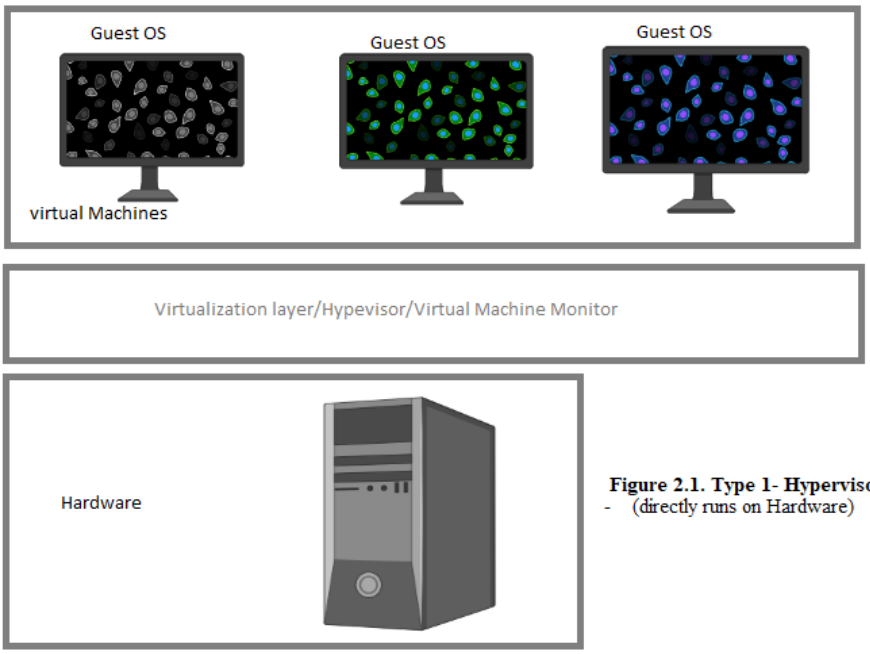

Figure 2.1. Type 1- Hypervisor
- (directly runs on Hardware)

Fig 2.1. Type 1 – Hypervisor

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 2, February 2022

DOI: 10.17148/IJARCCE.2022.11240

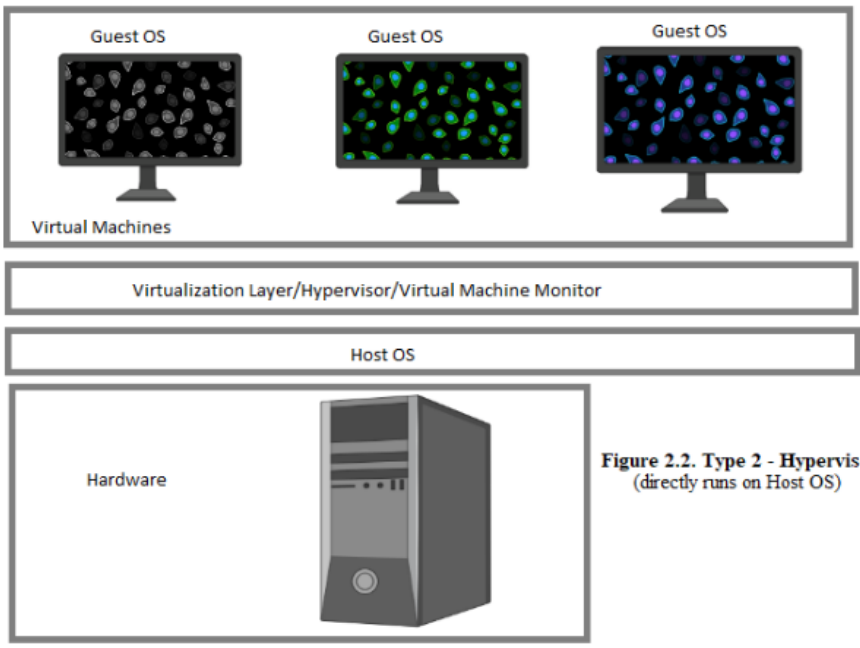

Figure 2.2. Type 2 - Hypervisor
(directly runs on Host OS)

Fig 2.2. Type 2 – Hypervisor

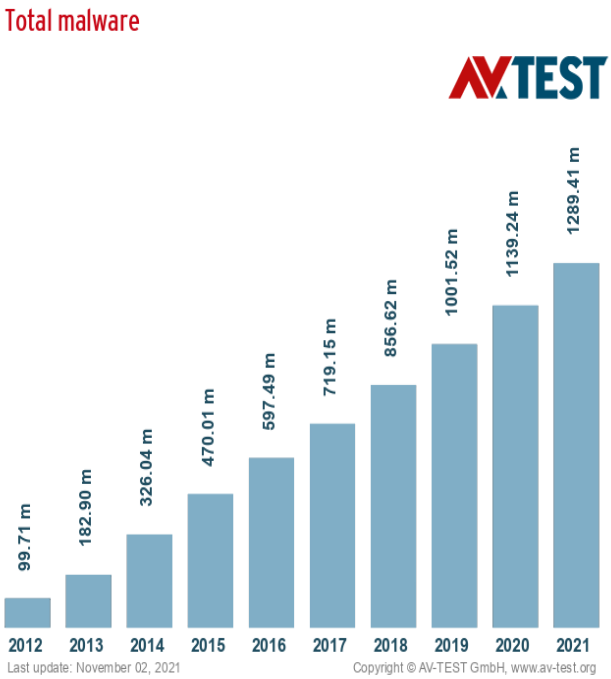

Figure 3. Statistical Report of Total Malware – Year-Wise by AV-TEST

	Machine Emulator	Type 2 - Hypervisor	Type -1 Hypervisor	Bare-Metal
Architecture	Code Based	Runs In Host OS	Runs On System Directly	No Virtualization
Pros	Easy To Use	Good Control	Medium Transparency	High Transparency
Cons	Low Transparency	Low Transparency	Less Control	High Transparency

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 2, February 2022

DOI: 10.17148/IJARCCE.2022.11240

Example	QEMU	Virtual Box	Xen	Barecloud

Table 1. Comparison of Malware Analysis Environment

2. RELATED WORK

Montasari et al (2021) explained hypervisor-based attacks in cloud computing environment comprised the components security in the cloud virtual infrastructure negatively affected the security of cloud elements and system security [1]. They provided an analysis of common and underexplored network and hypervisor-based attacks against CC systems from a technical viewpoint.

George et al (2021) explained about the analysis and classification of mitigation tools against cyber-attacks that prevented the vulnerability assessment in order to protect the organization's computer networks from black hat hackers [2].

Zaidenberg et al (2021) tested a proprietary thin hypervisor for dynamic malware analysis, not detected by malware and provided a complete transparent to the running OS and its applications [3]. A sophisticated malware under malware lab setup described inside a simulated environment. They provided the protection over detection and modification of monitoring component with proprietary hypervisor.

Amit et al (2021) analyzed data leakage prevention systems which attempted to prevent intentional or accidental disclosure of sensitive information by monitoring the content or the context in which the information is transferred under thin hypervisor named DLP-Visor [4]. They examined the data loss prevention for real-world applications deployment.

Mooney (2021) presented an approach to detect and stop ransomware as a service threat actor [5]. He identified the RAAS threat actors are perpetrated attacks against prominent organizations. He described basic security guidelines to protect against ransomware intrusion methods.

Alfred et al (2021) conducted a study on dynamic malware attack dataset leveraging virtual machine monitor [6]. They audited their described dataset versus prevailed dataset for IDS beneath Machine Learning (ML) algorithms with 10-fold cross-validation in cloud. They found that the intrusion detection accuracy over the dataset best suited for implementing IDS for cloud. The experiment results had significant deficiencies against ADFA and LID dataset.

Betsy et al (2021) tested dynamic malware analysis with the behavior of nine different malware samples based on the network traffic generated [7]. The experiment results are compared with a physical versus virtual environment. They presented how the malware behavior differs significantly in these two environments.

Hardy (2020) presented the RADICL lab to support cybersecurity research and training for students and the community [8]. Cyber situational awareness tools are implemented to monitor hardware, software and network packets. When suspicious activity or malware is detected, RADICL administrators gets alerted. The RADICL lab provided a solution for small to large businesses and research labs

Michael et al (2020) introduced a proprietary hypervisor 'hyperwall' for detection and prevention of malicious communication as external security tools (e.g., Firewalls) lack important information about the origin of the intercepted packets, thus filtering policy is usually insufficient to prevent communication between the malicious program and its operator [9]. They prevented malicious communication with a proprietary thin hypervisor which helped in deployment of real-world applications with less performance degradation.

Emmanuel et al (2020) presented a taxonomy of hypervisor forensics tool which provided a searchable catalog for forensic practitioners to identify specific tools that fulfill technical requirements for researchers [10]. They dealt with hypervisor operations with copious amounts of data that are of value in forensic investigations of virtualized cloud environments.

Muzahid et al (2020) presented a dynamic malware analysis under two scenarios' i.e.) Agent-based and agent less sandbox environment [11]. The dynamic analysis performed agent-based using Cuckoo open-source sandbox and agent-less using DRAKVUF by hypervisor and virtualization extension. They examined dynamic malware analysis over few pre-defined criteria including network requests, system injections and modifications, security measures and kernel alteration.

Jaspher et al (2020) presented a comparison between Drakvuf -VMI-based and Cuckoo Sandbox-based Technique for dynamic malware analysis with large number of malware samples [12]. They discussed two major categories to carry out dynamic malware analysis include out-of-box (Virtual Machine Introspection-based) and inside-the-box (Sandbox) techniques.

Igor et al (2020) provided how rootkits are injected into the target, as they can either resided in user or kernel space [13]. Kernel space rootkits are the hardest to detect and prevent because they usually take control of the kernel once compromised. They evaluated Write-protection Enforcement: a hypervisor-backed kernel hardening system to preserve the kernel self-protection active in presence of attacks. They experimented the integration of WpE with KVM and Linux kernel.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 2, February 2022

DOI: 10.17148/IJARCCE.2022.11240

Javier et al (2020) presented a method for malware analysis [14]. They examined both systematic and methodological process to analyze other malware threats like 'Stuxnet', 'Dark Comet', 'Poison Ivy', 'Locky', 'Careto', and 'Sofacy Carberp'. They investigated two well-known malwares as 'Flame' and 'Red October' to understand the malware intention entirely.

Geus et al (2019) introduced 'Lokke', a security hypervisor to protect the OS kernel against a class of attacks, not rely upon any specific vector [15]. They defined hybrid virtualization that combined the advantages of Type 1 and 2 hypervisors, where the hypervisor runs at the same level as the OS kernel does, but within a privileged execution framework. They designed a security framework to integrate with other security subsystems with relevant security policies enforced by the hypervisor and independent of the kernel.

Maliha et al (2019) used Cuckoo sandbox for dynamic malware analysis [16]. They tested the accuracy of dynamic malware analysis versus static malware analysis in a controlled network.

David et al (2019) conducted a comprehensive survey on malware dynamic analysis evasion techniques with a detailed classification [17]. They demonstrated how their efficacy holds against different types of detection and analysis approaches.

Vaidehi et al (2019) described an advanced persistent threat (APTs) that paved the way for most of the cyber espionages and sabotages [18]. The advanced evasive techniques (AET) used in APTs bypassed the stateful firewalls housed in the enterprise choke points at ease. They conducted a survey for the sophisticated attack and evasion techniques used by the contemporary malwares.

Harsha et al (2019) focused the vulnerability assessment and patching in VM-assisted agent-based malware detection (AMD) framework for high-risk virtual machines (VMs) security in cloud [19]. They described an agent at VM and anomaly detection at hypervisor. They experimented an anomaly detection using random forest classifier and classified the executable to either normal or malware and generated an alert to VM user. The examined and validated AMD framework over cloud testbed at NIT Goa with latest malware datasets.

Alexei et al (2017) presented an automated dynamic malware analysis systems include i) fingerprint-based evasion techniques for PC, mobile, and web, ii) evasion detection, iii) evasion mitigation, and iv) offensive and defensive evasion [20]. They designed the system for offensive and defensive research in anti-analysis.

3. PROPOSED WORK

Behavioral-based malware analysis is a well-known approach for analyzing and detecting the malicious content for a decade. Dynamic malware analysis can be done through a) emulation b) hooking c) hypervisor d) bare-metal. Various malware analyzing techniques include static, dynamic, hybrid, symbolic execution, fuzzing and concolic execution. A method for analyzing the malware under a proprietary hypervisor because commercial hypervisors are easily detected by day-to-day malwares easily.

Today, CPU's support MMU virtualization in hardware and provide SLAT to speed up the memory access in the guest mode. The research aim is to address effective dynamic analysis and threat detection with scalability, evasion resistance, kernel integrity with a proprietary thin hypervisor. As a thin hypervisor is designed to safeguard from hypervisor attack which in turn protects virtual machines (OS and application programs). The malware analysis components are concealed by the hypervisor. Threat hunting made using different techniques under undetectable monitoring component to have better performance, Security and transparency.

The thin VMM (thin hypervisor) having the virtual machines (OS and application programs) bounds the scope of attack and makes it much more difficult for the attacker to access unauthorized data and resources on the physical machine.VM state restore allows users to return to a state prior to attack or data loss, providing an easy method of malware removal and data preservation. By allowing users to start and stop VMs remotely, attackers have small time windows in which their preparation and attack must take place. This security aspect is important to safeguard the physical machine (Hosts). Since hypervisors run outside the VM, they have the potential to monitor for various malware intrusions. It is necessary that VM infrastructures to be safer and more secure than physical server infrastructures. The primary steps in modelling the thin hypervisor are shown in Figure 4.

The main objective of the thin hypervisor includes

- Monitoring component
- Isolation
- Threat hunting (static, dynamic)
- Report generation

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 2, February 2022

DOI: 10.17148/IJARCCE.2022.11240

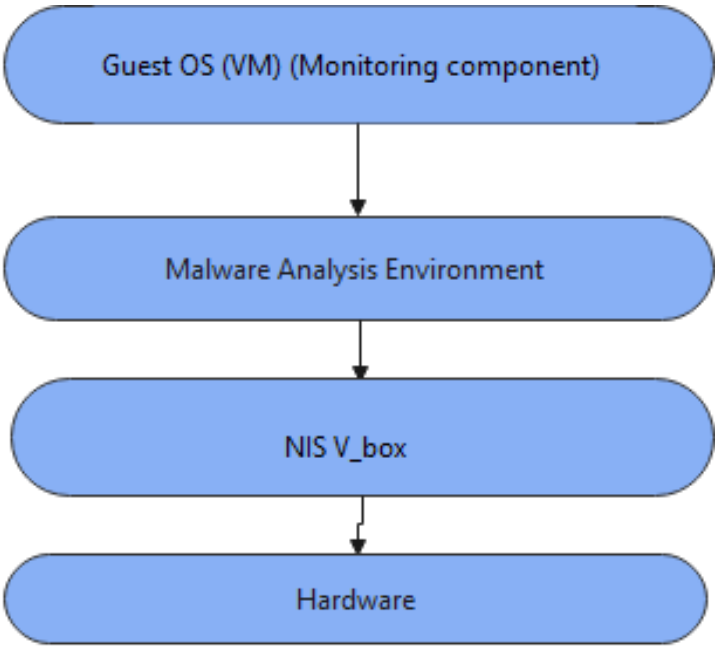

Fig 4. Primary Steps in Modelling the Thin Hypervisor (NIS V-Box)

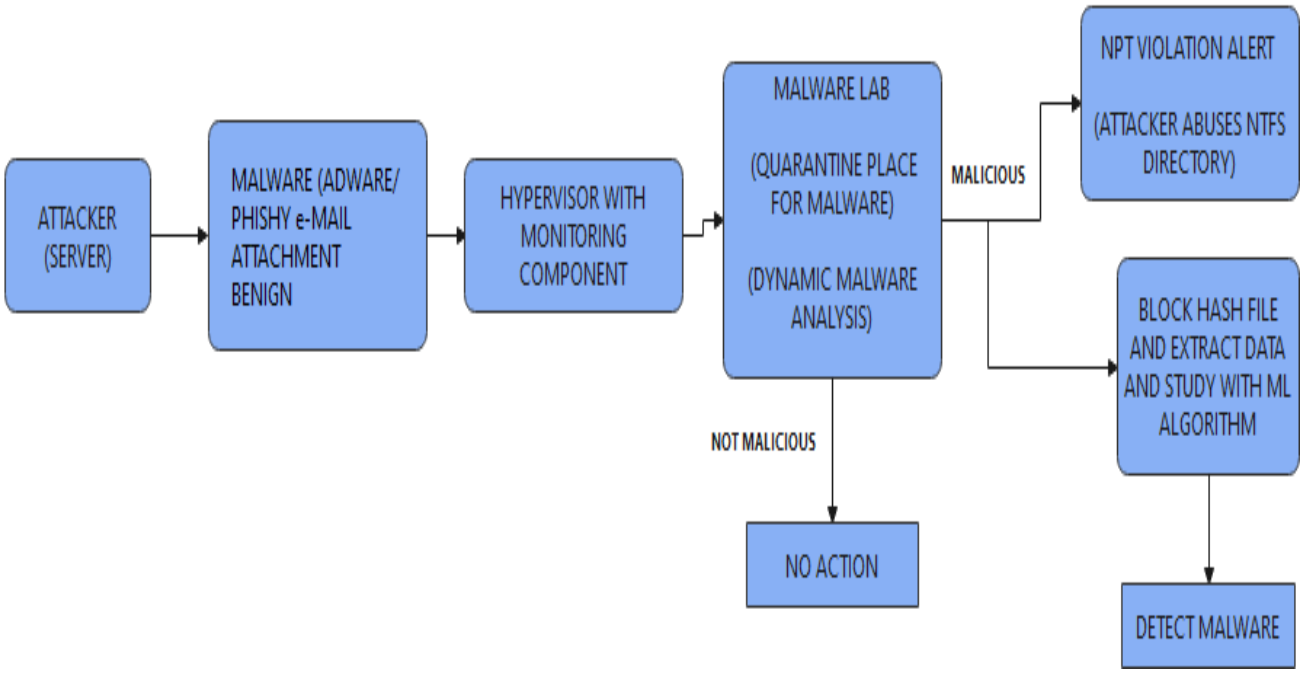

Fig 5. Work Flow of Proposed System

As security maintenance is important, hypervisor intrusion and modification have to be monitored and mitigated from any kind of vulnerabilities. It is necessary to be safeguard hypervisor from malware authors as they can run arbitrary code directly to host OS results in DOS attacks when security is breached through malicious content. The work flow of proposed system is shown in Figure 5.

The aim of the paper is to detect the intrusions made by malware-authors at hypervisor level and generate a report using SIEM tool (Splunk) and with dataset under IBM SPSS using ML algorithm. Architecture diagram of proposed work is shown in Figure 6.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 2, February 2022

DOI: 10.17148/IJARCCE.2022.11240

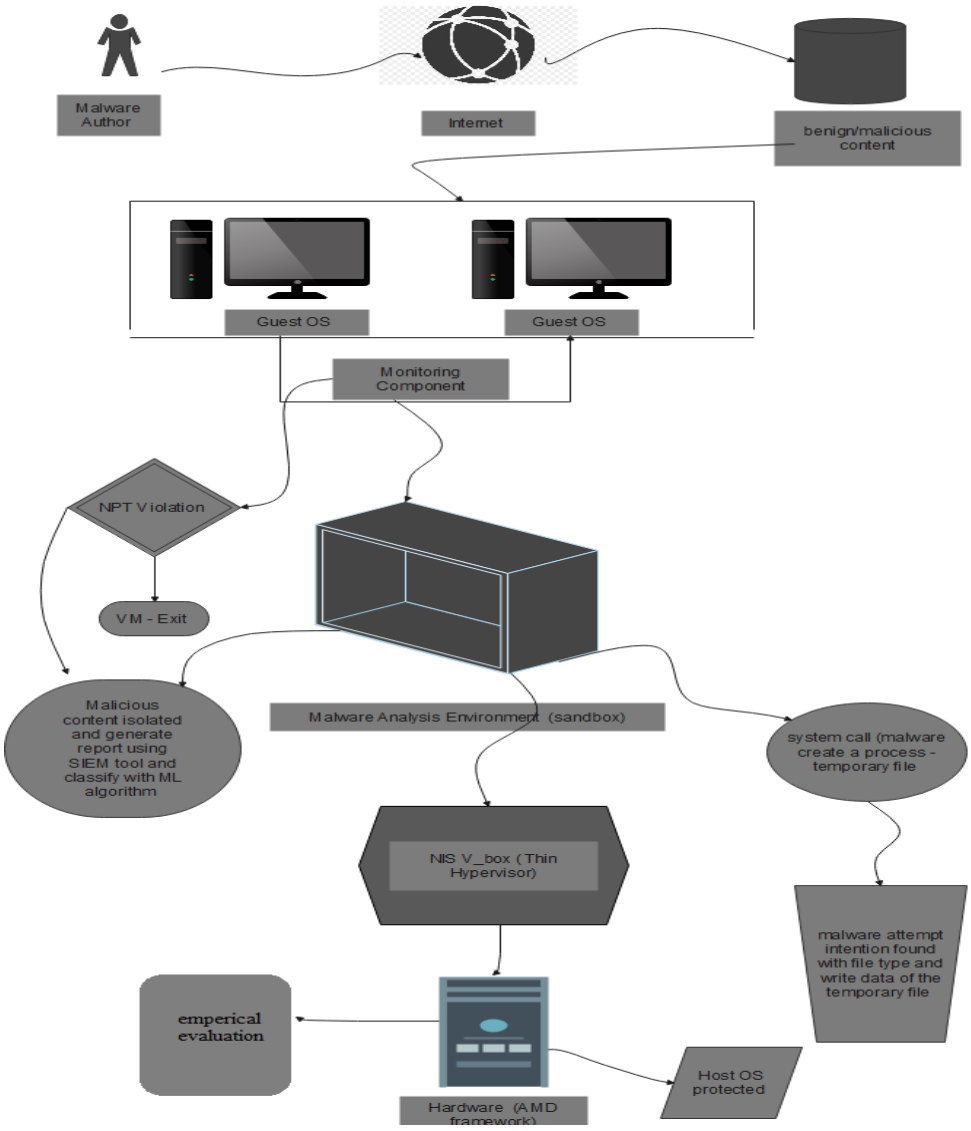

Fig 6. Architecture Diagram of Proposed Work

CONCLUSION

In cloud computing system, hypervisor (software component) serves as a main pillar of virtualization. This paper offers a discussion of implementing a thin hypervisor with a monitoring component to have data security and protection under non-trusted data in VMs. By developing a thin hypervisor (minimal code) with monitoring component, looks for various malicious contents which tries to attack and defend against those malicious attacks at the hypervisor level to protect the virtualized resources of the guest OS and in turn protect the host OS and its resources. In addition, setup malware lab for hunting the threats beneath behavioral based analysis and test the efficiency of the commercial hypervisors with the proprietary hypervisor. The empirical evaluation is presented using benchmarking tools.

REFERENCES

1. Montasari, R., Macdonald, S., Hosseinian-Far, A., Carroll, F., & Daneshkhah, A. (2021). Network and hypervisor-based attacks in cloud computing environments. International Journal of Electronic Security and Digital Forensics, 13(6), 630-651.
2. Iakovakis, G., Xarhoulacos, C. G., Giovas, K., & Gritzalis, D. (2021). Analysis and Classification of Mitigation Tools against Cyberattacks in COVID-19 Era. Security and Communication Networks, 2021.
3. Nezer Jacob Zaidenberg et al. (2021) Hypervisor-assisted dynamic malware analysis. leon2021hypervisor, 4,1,1--14, journal: Cybersecurity, Publisher SpringerOpen

DOI: 10.17148/IJARCCE.2022.11240

4. Amit, G., Yeshooroon, A., Kiperberg, M., & Zaidenberg, N. J. (2021). DLP-Visor: A Hypervisor-based Data Leakage Prevention System. In ICISSP (pp. 416-423).
5. Mooney, C. P. (2021). Detecting and Stopping Ransomware as a Service Threat Actors (Doctoral dissertation, Utica College)
6. Melvin AAR, Kathrine GJW, Ilango SS, Vimal S, Rho S, Xiong NN, Nam Y (2021) Dynamic malware attack dataset leveraging virtual machine monitor audit data for the detection of intrusions in cloud. Transactions on Emerging Telecommunications Technologies
7. Nazareno, B., Torres, I., & Jaramillo, J. (2021, March). Dynamic Malware Analysis: Contrast between Physical and Virtual Environment. In Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (pp. 1385-1385).
8. Hardy, R. N. (2020). Network Security Monitoring for Cyber Situational Awareness (Doctoral dissertation, University of Idaho).
9. Kiperberg, M., Yehuda, R. B., & Zaidenberg, N. J. (2020, November). HyperWall: A Hypervisor for Detection and Prevention of Malicious Communication. In International Conference on Network and System Security (pp. 79-93). Springer, Cham.
10. Mishra, A. K., Govil, M., & Pilli, E. (2020, January). A Taxonomy of Hypervisor Forensic Tools. In IFIP International Conference on Digital Forensics (pp. 181-199). Springer, Cham.
11. Muzahid, M. Z., Akram, M. B., & Alamgir, A. K. M. (2020, February). Analysis of Agent-Based and Agent-Less Sandboxing for Dynamic Malware Analysis. In International Conference on Cyber Security and Computer Science (pp. 72-84). Springer, Cham.
12. Melvin, A. A. R., & Kathrine, G. J. W. (2021). A Quest for Best: A Detailed Comparison Between Drakvuf-VMI-Based and Cuckoo Sandbox-Based Technique for Dynamic Malware Analysis. In Intelligence in Big Data Technologies—Beyond the Hype (pp. 275-290). Springer, Singapore.
13. Abdelraoof, A., Azab, M., & Stoppa, I. (2020, March). Write-protection enforcement: hypervisor-backed kernel hardening. In Proceedings of the 35th Annual ACM Symposium on Applied Computing (pp. 1736-1744).
14. Bermejo Higuera, J., Abad Aramburu, C., Bermejo Higuera, J. R., Sicilia Urban, M. A., & Sicilia Montalvo, J. A. (2020). Systematic approach to malware analysis (SAMA). Applied Sciences, 10(4), 1360.
15. Silva, O. A., & Paulo L´ıcio de Geus. Lokke, a hybrid security hypervisor.
16. M. Ijaz, M. H. Durad and Maliha Ismail, "Static and Dynamic Malware Analysis Using Machine Learning," 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), 2019, pp. 687-691, doi: 10.1109/IBCAST.2019.8667136.
17. Afianian, A., Niksefat, S., Sadeghiyan, B., & David Baptiste (2019). Malware dynamic analysis evasion techniques: A survey. ACM Computing Surveys (CSUR), 52(6), 1-28.
18. Chakkaravarthy, S. S., Sangeetha, D., & Vaidehi, V. (2019). A Survey on malware analysis and mitigation techniques. Computer Science Review, 32, 1-23.
19. Patil, R., Dudeja, H., & Modi, C. (2020). Designing in-VM-assisted lightweight agent-based malware detection framework for securing virtual machines in cloud computing. International Journal of Information Security, 19(2), 147-162.
20. Bulazel, A., & Yener, B. (2017, November). A survey on automated dynamic malware analysis evasion and counter-evasion: Pc, mobile, and web. In Proceedings of the 1st Reversing and Offensive-oriented Trends Symposium (pp. 1-21).
21. AV-TEST Institute malware test report chart.

