International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 05 | May - 2023 SJIF 2023: 8.176 ISSN: 2582-3930

NETWORK SECURITY AND ISOLATION USING HONEYPOT

Prem Waghulde[1], Vedant Patil[2], Shivam Patil[3] , Saurabh Wavre[4], Prof. Dr.Prakash Patil[5]

[1] Student, Department of Information Technology, D Y Patil College of engineering, Maharashtra, India
[2] Student, Department of Information Technology, D Y Patil College of engineering, Maharashtra, India
[3] Student, Department of Information Technology, D Y Patil College of engineering, Maharashtra, India
[4] Student, Department of Information Technology, D Y Patil College of engineering, Maharashtra, India
[5] Professor, Department of Electronics & Telecommunications, D Y Patil College of engineering, Maharashtra, India

---

Abstract -

. This abstract explores the use of honeypots for network security and isolation. Honeypots act as decoy systems to deceive attackers and gather valuable intelligence. By strategically deploying honeypots, organizations can identify vulnerabilities, mitigate risks, and enhance overall network defense. Honeypots enable the study of attacker behavior, collection of forensic data, and proactive defense measures. Continuous monitoring and timely response are crucial for effective implementation. Network security and isolation using honeypots offer enhanced visibility, proactive defense, and valuable insights in cybersecurity.

***Key Words*:** kippo, honeypot, network security, cyber threats, intrusion detection

1.INTRODUCTION

.. This project implements network security and isolation using honeypots. By strategically placing honeypots within the network, alongside standard security measures, the project aims to enhance network security. Honeypots lure attackers into simulated environments, recording their actions for analysis. VMware with clustering is used to create a virtualized network environment, ensuring scalability and flexibility. Failover clustering provides high availability and optimized resource utilization. The project combines honeypots, virtualization, and failover clustering to strengthen network security and isolation.

2. PROJECT SCOPE

This project aims to enhance network security using honeypots, which are decoy systems that trick attackers and alert administrators of unauthorized access attempts. The honeypot system diverts malicious traffic, provides real-time notifications to administrators, and isolates attackers within a controlled environment. Additional security measures, such as IP address authentication and Two-Factor Authentication, are implemented for enhanced protection. The project's goal is to improve network security by deploying a honeypot system that safeguards critical systems and enables efficient response to potential threats.

3. DESIGN SPECIFICATION

This project creates a virtual honeypot using a hypervisor to protect the network. Mimicking a real computer, it opens the SSH port to attract attackers who can gain access with incorrect credentials. Once inside, the honeypot captures keystrokes, logs activities, and records the attacker's IP address.

The honeypot system analyzes attacker behavior, such as popular passwords, common commands, and top IP addresses by country. Administrators can review these actions through a browser interface, gaining valuable insights.

Through these specifications, the project aims to build an effective honeypot that enhances network security, alerts administrators to threats, and enables detailed analysis of attacker behavior.

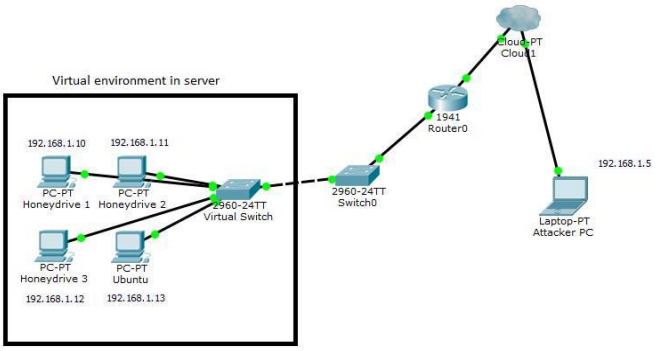

Figure: Design Specification

The virtual environment in the server is clustered with 3 servers which consists of 1 iSCSI storage server and 2 server nodes. The benefits of clustering are increase resource availability, effective resource usage, enhance performance, provide scalability and more simplified management. This system enables more resilient to server failures and provides high availability of the processing power. It has the flexibility for scaling the resources provisioning according to network demand.

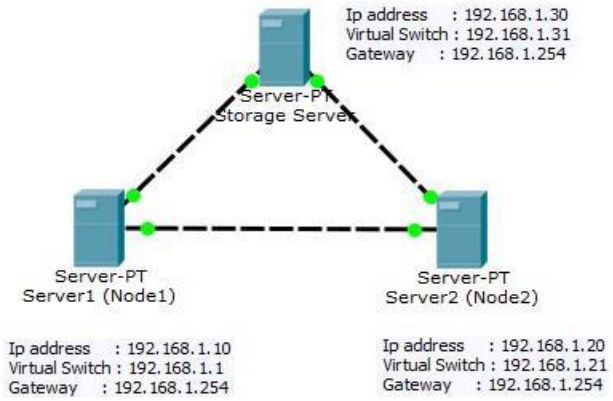

Figure : Clustering Design for Virtual Environment

4. SOFTWARE REQUIREMENTS

• ***VMware***
VMware can host multiple operating system as a virtual machine on the server. It is used to create a virtual and isolated environment of the Honeypot system.
• ***Nmap***
Nmap is used as a port scanner in the network. This is to determine which port is open for any kind of service in the network.
• ***Putty*** inside the virtual environment.
Putty is an SSH client which allows the user to SSH
• ***Kippo Honeypot***
Kippo is a medium-interaction SSH honeypot written in Python. It can log the brute force attack and every command typed in the system.
• ***Google Chrome***
Access the Internet or the localhost of the server webpage
• ***Google Authenticator***
Mobile apps to retrieve the one-time password (OTP) for the validation to login to the honeypot SSH

5. METHODOLOGY

The proposed methodology for developing the Honeypot system follows a waterfall approach. This approach ensures that requirements are clearly defined and progress flows steadily in a downward direction. Each phase, including requirement analysis, system design, implementation, testing, deployment, and maintenance, is completed before moving on to the next phase. This sequential process avoids overlapping and ensures a structured and organized development process.

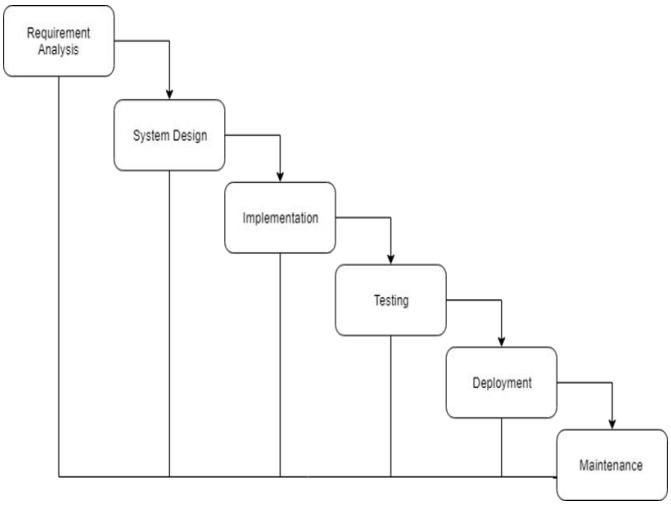

Figure :Waterfall Approach

During this phase, all the requirements for the Honeypot system are identified and documented for analysis. This includes determining the desired functionalities of the security systems, selecting the appropriate operating system, and ensuring the reliability of the Honeypot system.

System Design:

After studying the requirement specifications, the system design is created to define the necessary hardware and system requirements. This includes designing the implementation of virtual machines within a host and outlining how the system will operate.

Implementation:

Based on the system design, the Honeypot system is developed using a lab computer and the VirtualBox environment for testing. The virtual operating system with the Honeypot is set up, and the functionalities of the system are tested.

Integration and Testing:

The Honeypot system is integrated into the real server, which is hosted by Windows Server 2016. VMware is used to enable multiple virtual operating systems simultaneously. The entire system is tested for any failures, ensuring the proper functioning of the Honeypot.

Deployment of System:

Once functional and non-functional testing is complete, the system is deployed. Users in the network can perform port scanning to determine which port to access. Once they enter the Honeypot, the system records their keystrokes, IP address, and analyzes their behavior.

Maintenance:

To ensure a smooth working environment for users, the system is regularly maintained. Updates are applied to keep the system protected against the latest attacks by maintaining up-to-date virus signatures.

6. ISSUES AND CHALLENGES

Implementing the Honeypot system comes with its own set of challenges. One challenge is the compatibility of the downloaded HoneyDrive system, which is in .ova and .vmdk format, with the Windows Hypervisor Virtual machine. This requires converting the system files using specific commands to ensure compatibility. Additionally, configuring and installing the system involves various steps, from setting up the hard disk on the server machine to coding configurations.

During the system implementation, unexpected server breakdowns may occur, requiring troubleshooting to identify the root cause. This involves checking components such as the battery, RAM, and hard disk to determine the issue. In one instance, the power supply was found to be the problem, necessitating the replacement of the server machine. This experience provided valuable exposure to hardware repairs, benefiting future troubleshooting scenarios.

Another challenge involves establishing reliable communication between the three servers in the project. Initially, when attempting to cluster two servers and set up virtual storage, the clustering process failed. After investigating and resolving the issue, it was discovered that implementing Active Directory Domain (AD) was necessary for successful clustering. Furthermore, adding another server as a virtual storage provider was required to ensure proper functionality.

Despite these challenges, overcoming them through troubleshooting and problem-solving enhances the learning experience and prepares for future scenarios. The implementation of the Honeypot system allows for practical knowledge acquisition in hardware repairs, networking, and system configurations.

7. RESOLUTION OF THE LIMITATIONS

The implementation of Honeypot in the network system addresses the limitations identified in the literature review, offering unique advantages over traditional security systems.

• *Reaction to Network Attacks:*

While Firewall and Intrusion Prevention System (IPS) struggle to effectively react to network attacks, Honeypot captures the behavior of attackers within the network. Network administrators can observe and analyze malicious activities, gaining insights into the attacker's methods. This knowledge enables administrators to better protect the system by proactively countering new attack methods.

• *False Positives or False Negative Alarm Rates:*

Traditional security systems like Firewall and Intrusion Detection System (IDS) often generate false positives or false negatives, leading to incorrect decision-making and blocking legitimate connections. In contrast, Honeypot alerts the network administrator when an attack or suspicious behavior is detected. By allowing attackers to interact with the emulated system, the administrator can easily identify and assess their actions, significantly reducing false alarms.

• *Capturing of Data Packets:*

Unlike Firewalls that capture the entire packet traffic, Honeypot focuses on capturing specific traffic related to attacker attempts. This targeted approach eliminates the need for extensive storage for log files. Additionally, Honeypot captures keystrokes and utilizes tools like Kippo-Graph to provide administrators with a summarized overview of attacker actions, simplifying the analysis process and reducing storage requirements.

By leveraging the strengths of Honeypot, network administrators can enhance their ability to detect and respond to network attacks, minimize false alarms, and efficiently capture and analyze relevant data without overwhelming storage resources.

8. HONEYPOT DEPLOYMENT & CONFIGURATION

Honeypot deployment and configuration are essential for establishing a robust network security and isolation strategy. It requires careful planning and consideration of various factors to ensure honeypots are strategically positioned and designed to lure potential attackers. During deployment, it is important to determine the optimal placement of honeypots within the network, taking into account factors like network segmentation and critical assets. Configuring honeypots to closely resemble real systems and services is crucial to attract attackers. This involves setting up simulated services with open ports or vulnerable software versions and configuring logging and monitoring features to capture attacker activities. Regular monitoring and updating of honeypots are necessary to ensure their effectiveness against evolving attack techniques. By deploying and configuring honeypots effectively, organizations can bolster their network security, gain valuable insights into attacker behavior, and improve their incident response capabilities.

9.USE CASE DAIGRAM

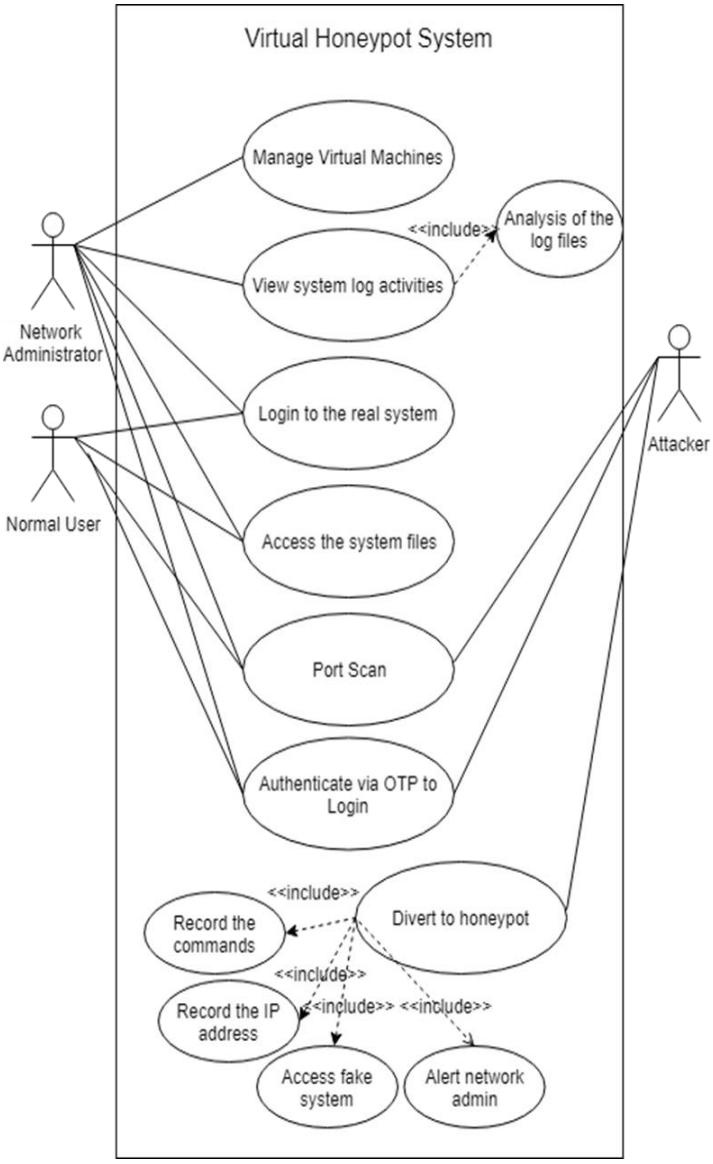

Figure :Use Case Daigram Of Virtual Honeypot System

10. FUTURE DIRECTIONS AND RESEARCH OPPORTUNITIES

Future directions and research opportunities in the field of honeypots offer exciting avenues for advancing network security and isolation. One potential area of exploration is the development of intelligent honeypots that utilize machine learning and artificial intelligence techniques to adapt and respond to emerging threats. These honeypots can continuously

learn from attacker behaviors, analyze patterns, and dynamically modify their responses to stay one step ahead. Additionally, there is a need for research on honeypot deception techniques, focusing on creating more realistic and convincing honeypot environments that can deceive even the most sophisticated attackers. Another promising direction is the integration of honeypots with threat intelligence platforms, enabling the sharing of real-time threat data and enhancing the detection and mitigation of cyber threats. Furthermore, research on the integration of honeypots with other security technologies, such as Intrusion Detection Systems (IDS) or Security Information and Event Management (SIEM) systems, can provide comprehensive threat visibility and improve incident response capabilities. Exploring the ethical implications of honeypot usage, such as legal considerations and privacy concerns, also presents an important avenue for future research. Overall, future research in honeypots holds significant potential in advancing network security, deception techniques, threat intelligence, integration with existing security technologies, and addressing ethical considerations for their effective and responsible use.

11. CONCLUSION

In summary, the virtual Honeypot system is set up in a server machine, consisting of 3 Honeypot and 1 Ubuntu virtual machines running simultaneously in VMware. The Honeypot software, Kippo, provides SSH service to the virtual machine, recording all keystrokes, IP addresses, and commands in a log file. The Kippo Graph enables network administrators to analyze attacker behavior using these log files, including charting and sorting services to identify the highest number of password attempts. Setting up the hardware involved configuring 3 server machines, including an iSCSI virtual storage server and clustering servers. I encountered a server breakdown and had to troubleshoot, repair, and replace the server. Implementing Failover Clustering required fulfilling prerequisite conditions like connecting servers to the same Active Directory and Domain. The virtual honeypot system offers a realistic environment that diverts attackers while integrating with the real network system to determine legitimate users. Unauthorized logins trigger notifications to the network administrator through a mobile application like Telegram, facilitating prompt countermeasures. Two-Factor Authentication adds an extra layer of security for user login. The frustration caused by the Honeypot system can discourage attackers, providing cost savings by deploying virtual honeypots and enhancing overall network security when combined with other security tools.

12. REFERENCES

[1]Akshay, H., Sanket, T., Ganesh K.,, n.d. Detection and Analysis of Network & Application Layer Attacks.

[2]Anon., 2016. Introduction to Honeypots. [Online]
Available at: https://blog.rapid7.com/2016/12/06/introduction-to-honeypots/

[3]Anon., n.d. [Online]
Available:
https://www.paloaltonetworks.com/cyberpedia/what-is-an-intrusion-prevention- system-ips

[4]Anon., n.d. CONPOT ICS/SCADA Honeypot. [Online]
Available at: http://conpot.org/

[5]Anon., n.d. HoneyDrive. [Online]
Available at: https://bruteforcelab.com/honeydrive

[6]Anon., n.d. Intrusion Detection System (IDS). [Online]
Available:
https://www.techopedia.com/definition/3988/intrusion-detection-system-ids

[7]Anon., n.d. What Is a Firewall?. [Online]
Available:
https://www.cisco.com/c/en/us/products/security/firewalls/what-is-a- firewall.html

[8]Bose, M., 2018. VMware or VirtualBox – Which One to Choose for Your Infrastructure?.
[Online]
Available https://www.nakivo.com/blog/hyper-v-virtualbox-one-choose-infrastructure/

[9]Brough, D., 2003. Second Generation Honeynet Honeywall.

[10]Christian, S., Ian, W., Peter, K., 2006. Taxonomy of honeypots. s.l.:s.n.

[11]Hsamanoudy, 2018. Advantages vs Disadvantages of Honeypots. [Online]
Available at: https://es.infosecaddicts.com/advantages-vs-disadvantages-of-honeypots/

[12]Nick, I., Cesar, U., Richard, B., 2005. Intrusion prevention systems.

[13]provos, N., Holz, T., 2007. Virtual Honeypots: From Botnet Tracking to Intrusion.

[14]Provos, N., 2004. A Virtual Honeypot Framework. s.l.:s.n.

[15]Provos, N., n.d. Honeyd: A Virtual Honeypot Daemon.

[16]Walsh, A., 2017. Alphabay and Hansa darknet markets shut down after international police operation. [Online]
Available at: https://www.dw.com/en/alphabay-and-hansa-darknet-markets-shut-down-after-international-police-operation/a-39776885

[17]X. Jhang, C. Li, W. Zheng,, 2004. Intrusion Prevention System Design. s.l.:s.n.

[18]Xinwen, F. Wei, Y., n.d. On Recognizing Virtual Honeypots and Countermeasures.