B.M.S. COLLEGE OF ENGINEERING
Bengaluru-560019.

Autonomous College, affiliated to
Visvesvaraya Technological University, Belgaum

Project Work - II Report on

"NETWORK HONEYPOT USING RASPBERRY PI TO MONITOR MALICIOUS ACTIVITY"

Submitted in partial fulfillment of the requirement for completion of
PROJECT WORK - II [22EC6PWPJ2]

Submitted by

Ananya S Bhat	**1BM21EC014**
Tejaswini J	**1BM21EC186**
Hamsa V N	**1BM22EC407**
Tanusha Prashanth	**1BM21EC182**

Under the guidance of

Dr. K. P. Lakshmi
Professor
BMS College of Engineering
Bengaluru

Academic Year

2023 – 2024

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BMS College of Engineering
Bull Temple Road, Basavanagudi, Bengaluru-560019

BMS COLLEGE OF ENGINEERING

Autonomous college, affiliated to VTU

Bull Temple Road, Bengaluru – 560 019

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the Project work - II entitled **“NETWORK HONEYPOT USING RASPBERRY PI TO MONITOR MALICIOUS ACTIVITY”** is a bonafide work carried out by **Ananya S Bhat (1BM21EC014), Tejaswini J (1BM21EC186), Hamsa V N (1BM22EC407) and Tanusha Prashanth (1BM21EC182)** submitted in partial fulfillment of the requirement for completion of PROJECT WORK - II [22EC6PWPJ2] of Bachelor of Engineering in Electronics and Communication during the academic year 2023-24. The Project Work - II report has been approved as it satisfies the academic requirements.

Guide

Dr. K.P.Lakshmi
Professor
Department of ECE
BMS College of Engineering

Head of Department

Dr. Siddappaji
Professor and Head
Department of ECE
BMS College of Engineering

Principal
BMS College of Engineering

External Viva

Name of the Examiners — Signature with Date

1.

2

DECLARATION

We, **Ananya S Bhat (1BM21EC014), Tejaswini J (1BM21EC186), Hamsa V N (1BM22EC407), and Tanusha Prashanth (1BM21EC182),** hereby declare that the Project Work -II entitled **NETWORK HONEYPOT USING RASPBERRY PI TO MONITOR MALICIOUS ACTIVITY** is a bonafide work and has been carried out by us under the guidance of **Dr. K.P. Lakshmi,** Professor, Department of Electronics and Communication Engineering, BMS College of Engineering, Bengaluru submitted in partial fulfilment of the requirement for completion of PROJECT WORK - II [22EC6PWPJ2] of Bachelor of Engineering in Electronics and Communication during the academic year 2023-24. The Project Work - II report has been approved as it satisfies the academic requirements in Electronics and Communication engineering, Visvesvaraya Technological University, Belagavi, during the academic year 2023-24.
We further declare that, to the best of our knowledge and belief, this Project Work - II has not been submitted either in part or in full to any other university.

Place: Bengaluru
Date:

Ananya S Bhat: 1BM21EC014
Tejaswini J: 1BM21EC186
Hamsa V N: 1BM22EC407
Tanusha Prashanth: 1BM21EC182

ACKNOWLEDGEMENTS

We take this opportunity to express our profound gratitude to the respected principal Dr. S. Muralidhara, BMS College of Engineering for providing a congenial environment to work in. Our sincere gratitude to Dr. Siddappaji, Head of the Department, Electronics and Communication Engineering for encouraging and providing an opportunity to carry Project Work - II in the department.

We heartily thank our guide Dr. K. P. Lakshmi for the guidance and constant encouragement throughout the course of Project Work - II without which this project would not be successful.

A number of personalities, in their own capacities have helped us in carrying out this Project Work - II. We would like to take this opportunity to thank them all.

Last but not the least, we thank our friends and family for their encouragement and help in accomplishing the objective of the Project Work - II work.

1) Ananya S Bhat : 1BM21EC014
2) Tejaswini J : 1BM21EC186
3) Hamsa V N : 1BM22EC407
4) Tanusha Prashanth : 1BM21EC182

ABSTRACT

In the contemporary digital era, the reliance on computer memory and cloud infrastructures for storing critical data has superseded traditional physical records. This transition has precipitated a marked increase in cyber thefts and cyber crimes, as malicious actors relentlessly seek to compromise legitimate servers to access valuable information. To address these burgeoning threats, the cybersecurity domain has developed various advanced defensive mechanisms. Among these, the honeypot stands out as a crucial tool.

A honeypot is a sophisticated security system that simulates vulnerable systems to attract and monitor attackers, thereby capturing their activities for in-depth analysis. This project focuses on the deployment of a honeypot using DShield software on a Raspberry Pi, a cost-effective and versatile platform, to study and analyze malicious cyber activities.

The honeypot successfully recorded a multitude of intrusion attempts, including various scanning techniques and exploit attempts, highlighting the persistent and evolving nature of cyber threats. The captured data were meticulously analyzed to decode the methodologies employed by attackers, providing valuable insights that can enhance network security protocols.

This implementation demonstrates the practicality and efficacy of leveraging Raspberry Pi for honeypot deployment in both educational and small-scale operational environments. The findings underscore the importance of proactive security measures and offer deep insights into the attack vectors and strategies utilized by cyber adversaries. This project not only contributes empirical data to the field of cybersecurity but also underscores the potential of low-cost solutions in strengthening digital security infrastructures.

TABLE OF CONTENTS

List of figures

List of abbreviations

Abbreviations	Full Form
H.T.T.P.	Hypertext Transfer Protocol
L.A.N.	Local Area Network
V.L.A.N.	Virtual Local Area Network
A.I.	Artificial Intelligence
A.P.I.	Application Programming Interface
S.S.H.	Secure Shell
U.R.L.	Uniform Resource Locator

Chapter 1:

Introduction

The exponential increase in global connectivity has led to an increase in internet usage and has transformed how individuals and organizations operate. This increased connectivity has also increased the vulnerability of data, making it accessible to unintentional users and susceptible to breaches. Recent statistics highlight a 20% rise in data breaches during 2022 - 2023, as reported by the Harvard Business Review [1]. Similarly, a C.N.B.C. report underlines the escalating threat posed by cyberattacks, particularly those driven by advancements in artificial intelligence (A.I.) [2]. These trends urge the need to enhance network security to mitigate the growing risk of cyber threats. One strategic measure that has gained prominence in cybersecurity is the deployment of honeypots. A honeypot is a deliberate vulnerable environment designed to attract potential attackers. This serves a dual purpose: it deceives adversaries and allows for an analysis on the unintended access requests. [3]. By systematically observing, analyzing, and gathering intelligence on cyber threats, particularly those involving automated bot activity, honeypots provide invaluable insights into the behavior and tactics of malicious access.

The intelligence gathered from honeypots aids in identifying new vulnerabilities and attack vectors, allowing organizations to strengthen their defenses proactively. This information is crucial for developing effective cybersecurity strategies and improving incident response protocols. Additionally, the data collected from honeypots can be shared with the broader cybersecurity community, contributing to collective efforts in combating cyber threats.

In this project we have deployed a network honeypot on a Raspberry Pi using Dshield, an open source intrusion detection software.

Chapter 2:

Literature survey

Honeypots, which originated from Clifford Stoll's experiment in the late 1980s, have evolved from basic traps into sophisticated tools capable of simulating entire networks or individual services, enabling detailed observation of attacker behavior [4]. They are divided into low-interaction honeypots, which emulate specific services to capture basic attack data, and high-interaction honeypots, which replicate complete systems for deeper insights into complex attack strategies [5]. Established in 1999, the Honeynet Project advanced the field by creating networks of honeypots that collect and share information on global cyber threats [6][7]. Modern honeypots now incorporate AI and machine learning, allowing them to dynamically respond to evolving threats and automate attack pattern analysis [8]. Advanced honeypots, like Cowrie, can log detailed interactions and adapt to sophisticated attacks [9].

The primary advantage of honeypots is their ability to gather actionable intelligence on attackers, aiding in the development of stronger defense mechanisms and improving incident response [10]. However, deploying honeypots, especially high-interaction ones, requires careful management to prevent them from becoming tools for exploitation and raises ethical and legal concerns about monitoring attacker activities [11]. The integration of AI with honeypot technology and the shift towards cloud-based honeypots promise to enhance their effectiveness and scalability in combating cyber threats [12]. Bots, sophisticated automated software applications, have become increasingly advanced with the integration of AI technologies. Their capabilities now encompass various malicious activities, from orchestrating cyberattacks to scanning for system vulnerabilities and spreading malware [13]. Within this context, honeypots play a critical role in analyzing bots' adaptive behaviors and responses to cybersecurity defenses. This analysis is crucial for developing more robust defensive strategies, ensuring proactive resilience against the ever-evolving threat landscape of the digital realm [14].

In addition to traditional honeypots, this work explores the implementation of a network honeypot on Raspberry Pi using DShield, a distributed intrusion detection system (IDS) that aggregates data from multiple sources to monitor and analyze global attack patterns. DShield is a community-based collaborative firewall log correlation system [15]. The DShield honeypot collects logs from participating devices, including those deployed on low-cost platforms like Raspberry Pi, and contributes to a broader understanding of internet-wide security threats. This approach combines the lightweight and affordable Raspberry Pi hardware with the capabilities of DShield, creating a scalable and effective solution for tracking and analyzing cyber threats.

The setup involves configuring the Raspberry Pi to maintain open ports for monitoring inbound traffic logs within a simulated vulnerable network. Insights derived from this setup can inform decisions regarding the blacklisting or blocking of identified threat actors, thereby enhancing the overall security posture of the network. This method leverages the cost-effectiveness and versatility of Raspberry Pi devices, making advanced cybersecurity measures more accessible and feasible for a wider range of organizations. The data gathered from these network honeypots can be invaluable in identifying trends, understanding attacker methodologies, and developing stronger defensive strategies.

Chapter 3:

Problem Analysis & Solution

3.1 Problem Definition

Existing cybersecurity measures face challenges in effectively detecting and mitigating evolving cyber threats, leading to vulnerabilities in network security. Traditional approaches often struggle to keep pace with sophisticated attack techniques, resulting in gaps in threat detection and response capabilities. There is a critical need for innovative solutions that can proactively monitor network activity, analyze threat behaviors, and provide actionable insights to enhance overall cybersecurity.

1. Explore the concept of Honeypots, their role in network security.
2. Explore the deployment of Honeypots on stand-alone systems such as Raspberry Pi, and other development boards.
3. Explore how IP addresses can be mirrored, pseudo IP address assignment.
4. Explore how the Honeypot can be made to seem vulnerable, i.e., how and which ports can be opened.
5. Explore the concept of in-system Honeypots and how they can be incorporated into existing computer systems.
6. Explore how a port can be configured to only allow incoming interactions, but not allow outgoing interactions.
7. Explore how ports can be configured using Kali Linux.
8. Explore how ports can be scanned using Kali Linux.
9. Observe how attackers interact with the Honeypot to prepare a list of U.R.L.s.
10. Explore Honeypots for S.S.H., Telnet and Firewall attackers.
11. Explore how a Honeypot could be virtually set up on a computer without the use of any Hardware.
12. Observe and continuously monitor the logs coming from the Honeypot and analyze the data using algorithms to prepare a model for blacklisting U.R.L.s.
13. Observe how many AI bots regularly try to interact with the network and how they can affect the network traffic.

14. Learn how our internet activity is continuously monitored i.e. Threat Intelligence exploration.
15. Explore the application of this technology to home networks and large networks such as Company networks.

3.2 Proposed Solution

The proposed solution involves deploying a Raspberry Pi-based honeypot within the network infrastructure, simulating vulnerable services and systems to attract and gather data on malicious activities. This honeypot will be integrated with DShield, a collaborative network security platform, to aggregate and analyze real-time threat intelligence from a global network of sources. Advanced algorithms and machine learning techniques will be utilized to detect patterns indicative of cyber threats, allowing for proactive threat mitigation measures. We would need to further look into using other softwares such as Open Canary and our own configurations to customize the ports of the Honeypot according to our needs.

Block Diagram:

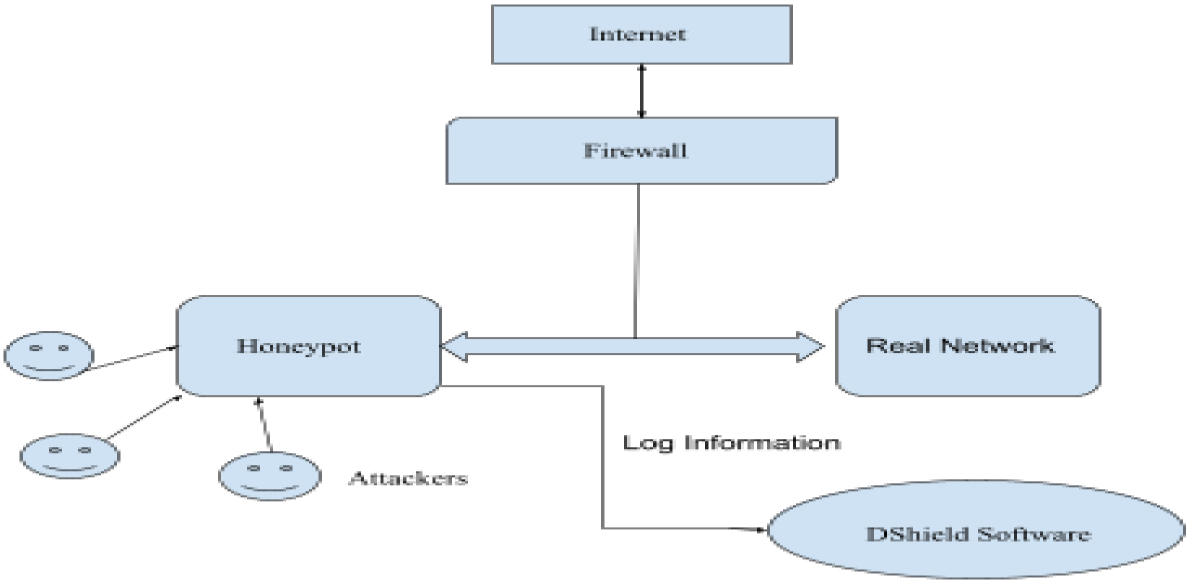


Fig1. Block diagram of Proposed Solution

Chapter 4:

Methodology & Implementation

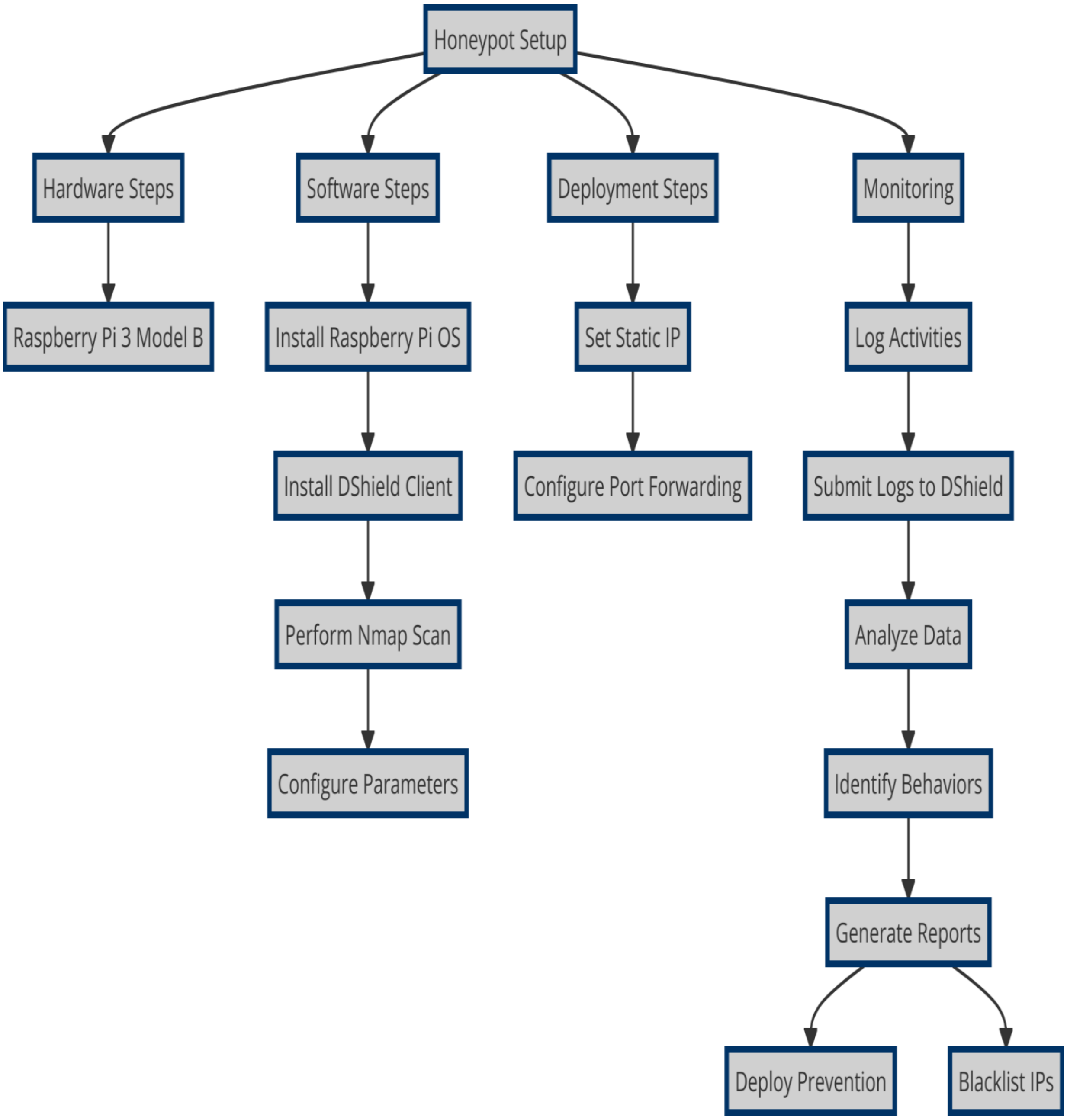


Fig 2. Flow Chart of Methodology of Implementation

The honeypot was deployed on a Raspberry Pi 3 Model B with 1 GB of R.A.M. This model was chosen for its processing power, built-in Wi-Fi, ethernet connectivity, and cost-effectiveness. The Raspberry Pi was equipped with a 64 GB microSD card for storage.

The software setup began with installing a suitable Raspberry Pi Operating System, followed by the installation of the DShield client honeypot software. The DShield software repository was cloned onto the Raspberry Pi to ensure access to the latest version of the DShield software and its associated resources. An installation script provided in the repository by the SANS Internet Storm Center was executed, and the necessary settings for the proper functioning of the honeypot, including the preferred method for obtaining logs (automatic or manual), were configured.

To complete the setup, registration on the DShield website was required, after which a unique A.P.I. key was assigned to the Raspberry Pi honeypot. This A.P.I. key served as a critical authentication token, enabling secure communication between the Raspberry Pi honeypot and the DShield servers. The A.P.I. key was integrated into the honeypot during the software configuration, establishing a secure connection for reporting security events and suspicious activities.

As part of the configuration process, static IP addresses were assigned to the Raspberry Pi, making the device exclusively a honeypot dedicated to monitoring network activity and logging it. Data collection involved capturing and logging network activity and system events generated by the honeypot. The DShield client software collected and aggregated firewall and IDS logs from the Raspberry Pi, which were then automatically submitted to the DShield database for analysis as the Raspberry Pi received the logs.

Chapter 5:

Results & Discussion

In this work, a honeypot was deployed using a Raspberry Pi 3 Model B with DShield client software to simulate a vulnerable web server within a Local Area Network (LAN). The objective was to mimic common web-based attacks and analyze the honeypot's responses using controlled simulations from internal devices.

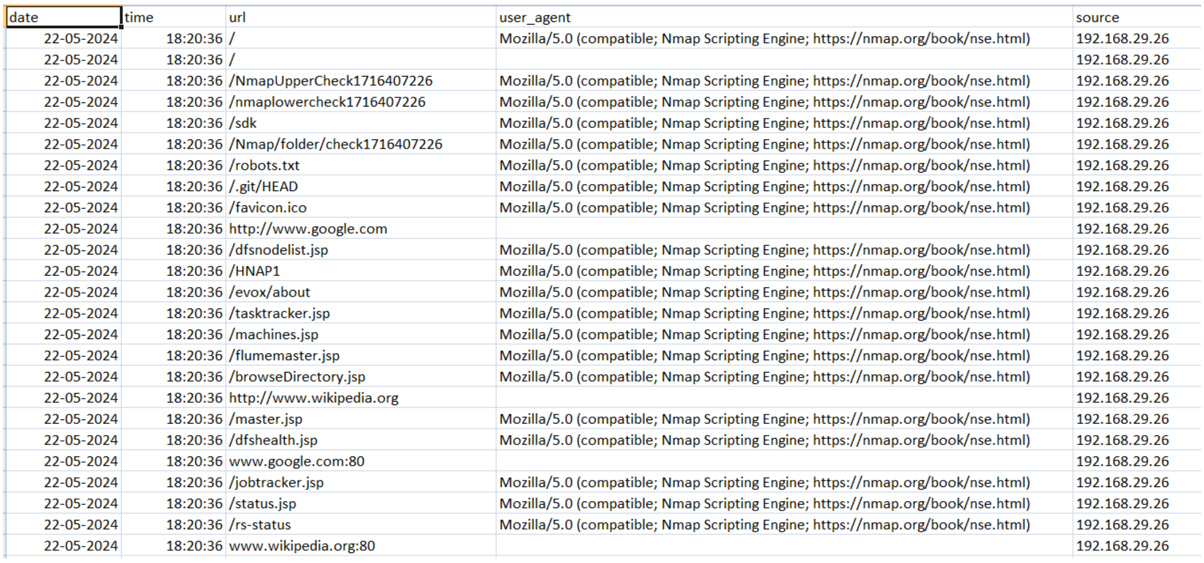
The honeypot was configured to maintain open H.T.T.P. ports, presenting itself as an exposed web server. Multiple devices within the L.A.N. were utilized to execute simulated attacks, including U.R.L. probing and vulnerability scans. Nmap, a well-known network scanning tool, was used to carry out these simulated attacks. Automated Nmap scripts mimicked typical attack behaviors by scanning for open ports, identifying services, and probing for potential vulnerabilities on the web server.

Figure 3 displays sample logs captured by the honeypot, showcasing detailed interactions resulting from the simulated attacks. The logs included H.T.T.P. requests detailing the U.R.L.s accessed, the nature of the H.T.T.P. requests, and response patterns. U.R.L. probing was simulated by accessing various endpoints to emulate attempts to discover and exploit web resources. Service detection scans simulated efforts to identify the types of services running on the honeypot by analyzing H.T.T.P. headers and response codes.

Analysis of the logs revealed consistent patterns of interaction with the simulated attacks. The honeypot effectively recorded the date and time of the requests and the types of H.T.T.P. requests, demonstrating its capability to log detailed data from the simulated attack vectors, as illustrated in Figure 3.

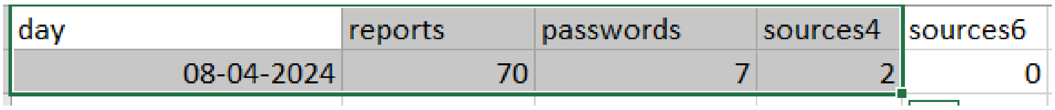
The data was displayed in a tabulated form on the DShield website, showing accessed U.R.L.s, timestamps, and types of H.T.T.P. requests. The DShield database also contains historical information from past users of the software regarding logs received from the same U.R.L.s and IP addresses as those captured by our honeypot. This enables users to understand the frequency of attacks from certain U.R.L.s and addresses on other honeypots deployed worldwide. All log data could be downloaded as C.S.V. files, allowing for further analysis and visualization using tools like Microsoft Excel to gain a better understanding of network activity on the honeypot. The DShield website also provided options for visual representation of the statistics related to the U.R.L.s and IP addresses associated with the logs.

Additionally, some S.S.H. logs were recorded by the software, capturing usernames and passwords used during the simulated Nmap scans executed on the device. The analysis confirmed the honeypot's effectiveness in detecting and recording web-based attacks within a controlled setting. The honeypot successfully provided a realistic platform for observing common attack techniques, capturing interaction data crucial for understanding how typical vulnerabilities could be probed and exploited.



date	time	url	user_agent	source
22-05-2024	18:20:36	/	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/		192.168.29.26
22-05-2024	18:20:36	/NmapUpperCheck1716407226	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/nmaplowercheck1716407226	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/sdk	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/Nmap/folder/check1716407226	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/robots.txt	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/.git/HEAD	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/favicon.ico	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	http://www.google.com		192.168.29.26
22-05-2024	18:20:36	/dfsnodelist.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/HNAP1	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/evox/about	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/tasktracker.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/machines.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/flumemaster.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/browseDirectory.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	http://www.wikipedia.org		192.168.29.26
22-05-2024	18:20:36	/master.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/dfshealth.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	www.google.com:80		192.168.29.26
22-05-2024	18:20:36	/jobtracker.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/status.jsp	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	/rs-status	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)	192.168.29.26
22-05-2024	18:20:36	www.wikipedia.org:80		192.168.29.26

Fig 3. Logs recorded on the Dshield Website



day	reports	passwords	sources4	sources6
08-04-2024	70	7	2	0

Fig 4. Total logs captured on a particular day

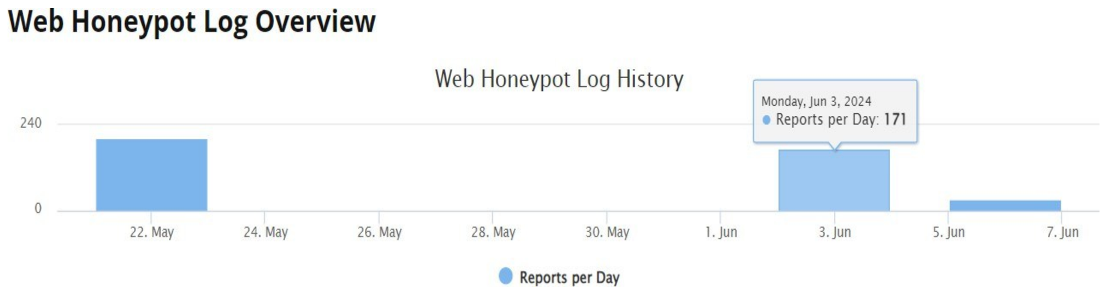


Fig 5. Log Reports

Chapter 6:

Future Trends and Conclusion

6.1 Conclusion

Implementing a honeypot using software like DShield on a device such as a Raspberry Pi is a practical and effective way to enhance network security. Though practical implementation, such as creating a virtual network for a honeypot, can be quite heavy work and risky, this serves as a first step towards learning. This setup allows for the capture and analysis of malicious activity, providing valuable insights into attack methods and behaviors. By monitoring and analyzing this data, organizations can strengthen their security posture, improve incident response, and proactively defend against potential threats. Regular maintenance and updates ensure the honeypot remains a valuable tool in the dynamic field of cybersecurity. Moreover, it is very successful in capturing all logs, making it an invaluable resource for understanding and mitigating cyber threats.

6.2 Future Trends

For securing the honeypot on your network, you can set up a Virtual Local Area Network (V.L.A.N.) or a guest network. A V.L.A.N., if supported by your router, can isolate the honeypot from your main network, while a guest network can similarly keep the honeypot separate. It is crucial to configure the V.L.A.N. or guest network to ensure that there is no communication between the honeypot and your primary devices, thus enhancing security. Features like AP Isolation can be employed to restrict devices on the honeypot network from accessing your main network. Assign specific ports for the honeypot to listen to, such as port 22 for S.S.H. and port 80 for H.T.T.P., and configure your router to allow traffic on these ports to the Raspberry Pi within the isolated network. This setup helps contain any potential threats within the honeypot environment without impacting the main network.

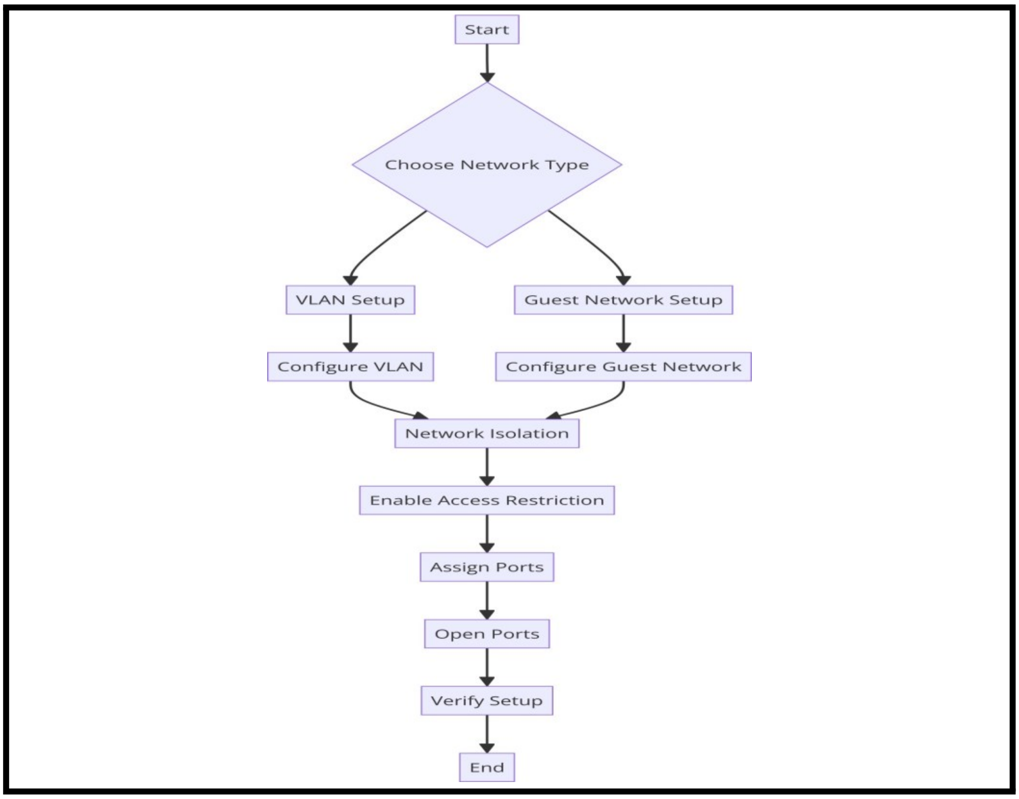


Fig 6. Future Scope for Work through Virtual L.A.N.

REFERENCES

[1] Stuart Madnick, (February 19, 2024), “Why Data Breaches Spiked in 2023”, Harvard Business Review, https://hbr.org/2024/02/why-data-breaches-spiked-in-2023.

[2] Kevin Williams,(March 3, 2024), "Cyber-physical attacks’ fueled by A.I. are a growing threat”, CNBC https://www.cnbc.com/2024/03/03/cyber-physical-attacks-fueled-by-ai-are-a-growing-threat-experts-say.html.

[3] Spitzner, L. "Honeypots: Tracking Hackers." Addison-Wesley Longman Publishing Co.,2002.

[4] Caleb Townsend,. "What is a Honeypot", United States Cybersecurity Magazine. https://www.uscybersecurity.net/honeypot/

[5] M.Shukla,P.Verma,“Honeypot: Concepts, Types and Working”, IJEDR, Volume 3, Issue 4, ISSN: 2321-9939, 2015.

[6] The Honeynet Project. (2024). "About Us”.

[7] Spitzner, L. "The Honeynet Project: Trapping the Hackers.", IEEE Security & Privacy, Volume: 1, Issue: 2 IEEE Xplore, 2003.

[8] Sanders, C. "Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems." No Starch Press,2010.

[9] Cowrie. (2024). "Cowrie: Medium Interaction S.S.H. and Telnet Honeypot." , GitHub. https://github.com/cowrie/cowrie.

[10] V. A. Memos and K. E. Psannis, "A.I.-Powered Honeypots for Enhanced IoT Botnet Detection,", 3rd World Symposium on Communication Engineering (WSCE), Thessaloniki, Greece, 2020, pp. 64-68, doi: 10.1109/WSCE51339.2020.9275581, 2020

[11] Yang, X.; Yuan, J.; Yang, H.; Kong, Y.; Zhang, H.; Zhao, J. A Highly Interactive Honeypot-Based Approach to Network Threat Management. Future Internet,2023.

[12] Alyas, Tahir & Alissa, Khalid & Alqahtani, Mohammed & Faiz, Tauqeer & Alsaif, Suleiman & Tabassum, Nadia & Naqvi, Hafiz. , “Multi-Cloud Integration Security Framework Using Honeypots”, Mobile Information Systems, 2022.

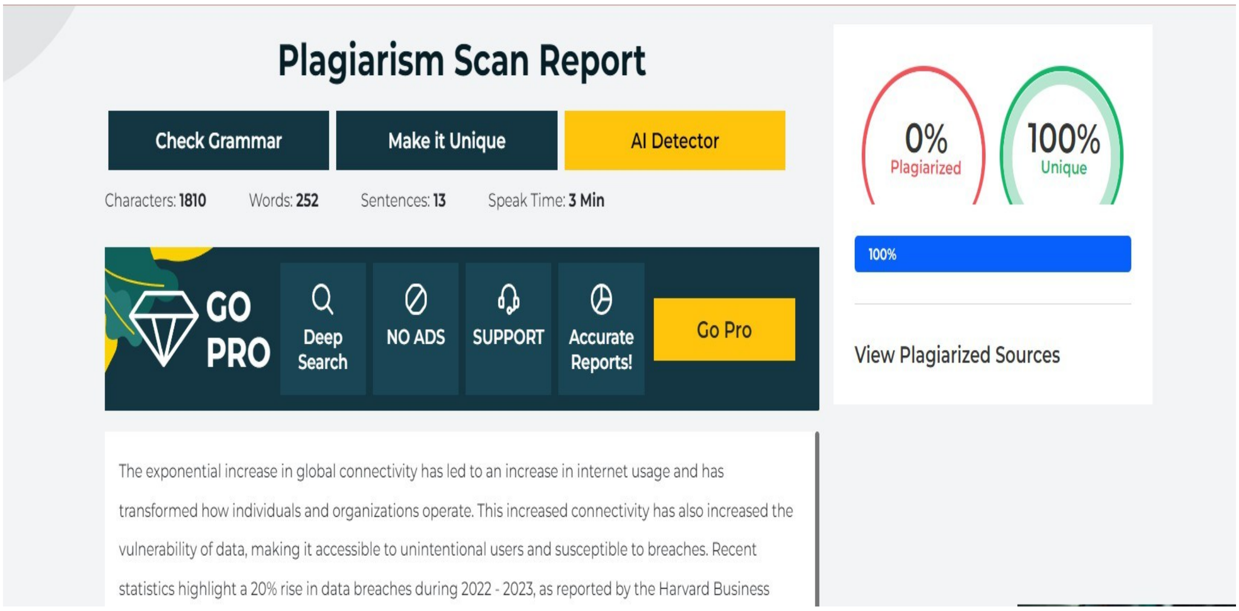
[13] Provos, N., & Holz, T. "Virtual Honeypots: From Botnet Tracking to Intrusion Detection." Addison-Wesley, 2008.

[14] Dr. V S Narayana Tinnaluri “A Comprehensive Approach: Developing A Honeypot System To Thwart Cyber Attackers”, Educational Administration: Theory and Practice,2024.

[15] SANS Internet Storm Center. (2024). "DShield Overview”.

APPENDIX A:

Plagiarism Report



Characters: **1810** Words: **252** Sentences: **13** Speak Time: **3 Min**

The exponential increase in global connectivity has led to an increase in internet usage and has transformed how individuals and organizations operate. This increased connectivity has also increased the vulnerability of data, making it accessible to unintentional users and susceptible to breaches. Recent statistics highlight a 20% rise in data breaches during 2022 - 2023, as reported by the Harvard Business

APPENDIX B:

Research Publications:

Paper Submitted to IEEE ICEC: International Conference on Intelligent Computing and Emerging Communication.