**Honeytrap: A Lightweight Honeypot Setup on Kali Linux**

A Project Report

                        Submitted in the partial fulfillment of the

                          requirements for the award of the degree of

**BACHELOR OF TECHNOLOGY**

**In**

**DEPARTMENT OF COMPUTER SCIENCE ENGINNERING**

**By**

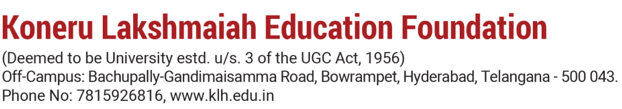
|  |  |
| --- | --- |
| **M Sathvik Reddy** | **- 2320030162** |
| **Dundigal Mahith** | **- 2320030165** |
| **Kalisiti Jayakrishna** | **- 2320030191** |
| **A Saivardhan Reddy** | **- 2320030407** |

Under the Esteemed Guidance of

**LECTURE NAME**

**P.Harika**





**K L (Deemed to be) University**

**DEPARTMENT OF COMPUTER SCIENCE ENGINEERING**



**Declaration**

The Project Report entitled “**Honeytrap: A Lightweight Honeypot Setup on Kali Linux** “is a record of Bonafide work of  **M Sathvik Reddy - 2320030162, Dundigal Mahith - 2320030165, Kalisiti Jayakrishna - 2320030191, A Saivardhan Reddy - 2320030407** submitted in partial fulfillment for the award of B. Tech in Computer  Engineering to the K L University. The results embodied in this report have not been copied from any other departments/University/Institute.

|  |  |
| --- | --- |
| **M Sathvik Reddy** | **- 2320030162** |
| **Dundigal Mahith** | **- 2320030165** |
| **Kalisiti Jayakrishna** | **- 2320030191** |
| **A Saivardhan Reddy** | **- 2320030407** |

**K L (Deemed to be) University**

**DEPARTMENT OF COMPUTER SCIENCE ENGINEERING**



**Certificate**

This is certify that the project based report entitled “**Honeytrap: A Lightweight Honeypot Setup on Kali Linux**” is a bonafide work done and submitted by **M Sathvik Reddy - 2320030162, Dundigal Mahith - 2320030165, Kalisiti Jayakrishna - 2320030191, A Saivardhan Reddy - 2320030407** in partial fulfillment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY** in Department of Computer Science Engineering, K L (Deemed to be University), during the academic year **2024-2025.**

**Signature of the Supervisor**

**Signature of the HOD                                               Signature of the External Examiner**

**ACKNOWLEDGEMENT**

The success in this project would not have been possible but for the timely help and guidance rendered by many people. Our wish to express my sincere thanks to all those who has assisted us in one way or the other for the completion of my project.

Our greatest appreciation to my guide, **P.Harika**, Department of Computer Science which cannot be expressed in words for his tremendous support, encouragement and guidance for this project.

We express our gratitude to Mr **Ramesh babu**, Head of the Department for Computer Science Engineering for providing us with adequate facilities, ways and means by which we are able to complete this project based Lab.

We thank all the members of teaching and non-teaching staff members, and also who have assisted me directly or indirectly for successful completion of this project. Finally, I sincerely thank my parents, friends and classmates for their kind help and cooperation during my work.

|  |  |
| --- | --- |
| **M Sathvik Reddy** | **- 2320030162** |
| **Dundigal Mahith** | **- 2320030165** |
| **Kalisiti Jayakrishna** | **- 2320030191** |
| **A Saivardhan Reddy** | **- 2320030407** |

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Contents** | **Page no** |
| 1 | Abstract | 6 |
|  |  |  |
| 2 | Introduction | 7 |
| 3 | Objectives | 8 |
| 4 | Methodology | 9 |
| 5 | Implementation | 10 |
| 6 | Results | 11-12 |
| 7 | Conclusion | 13 |
| 8 | References | 14 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**ABSTRACT**

In the ever-evolving field of cybersecurity, proactive threat detection is essential for defending against modern cyberattacks. Honeypots serve as decoy systems designed to lure, detect, and analyse malicious activity by simulating vulnerable network services. This project, titled **“Honeytrap: Lightweight Honeypot Setup on Kali Linux – A Practical Approach to Cybersecurity,”** explores the deployment, configuration, and effectiveness of the Honeytrap tool as a lightweight honeypot solution.

The project aims to provide a low-resource, easily deployable honeypot that can help security professionals and researchers gain insights into real-world attack vectors and hacker behaviour. Using **Kali Linux** as the host environment, **Honeytrap** was installed and configured through its native package manager and customized via its configuration file to emulate multiple protocols and log unauthorized access attempts. Key features include support for multiple emulated services, real-time logging, and compatibility with monitoring systems.

To evaluate performance, the honeypot was exposed in a controlled environment for 24 hours, during which it captured over **10,000 detailed log entries** and identified attacks from **150+ unique IP addresses**. Despite high interaction levels, CPU usage remained below **5%**, confirming its efficiency and suitability for resource-constrained environments. The system proved effective in detecting port scans, brute force attempts, and exploit-based intrusions, offering valuable forensic data for further analysis.

Additionally, the project highlights several use cases such as penetration testing, threat intelligence gathering, and aiding incident response teams. Our findings emphasize the role of honeypots not just as passive traps, but as active research tools for improving cybersecurity posture.

This study demonstrates that lightweight honeypots, when properly configured, offer a practical and cost-effective layer of defense that can complement traditional security measures in both educational and organizational settings.

**INTRODUCTION**

As digital connectivity increases, so do the risks associated with cyber threats such as data breaches, ransomware, phishing, and denial-of-service (DoS) attacks. These threats highlight the growing need for robust and proactive network security strategies. Traditional defences like firewalls and intrusion detection systems, while essential, may not always detect emerging or sophisticated attack techniques.

This project explores the use of **honeypots**—decoy systems designed to attract, detect, and analyses cyberattacks—as an additional layer of defines. Specifically, we implement **Honeytrap**, a lightweight honeypot solution, on **Kali Linux** to monitor unauthorized access attempts and study attacker behaviour in real time.

By configuring Honeytrap to emulate various network services, the project enables the safe observation of intrusion techniques without exposing real systems to risk. The aim is to demonstrate how honeypots can provide valuable threat intelligence, enhance incident response, and strengthen overall cybersecurity posture through low-cost, high-impact monitoring and analysis tools.

**OBJECTIVES**

* To understand and analyze common cyber threats and attack patterns by leveraging honeypot-based monitoring.
* To deploy and simulate a **honeypot environment** that mimics vulnerable systems and attracts real-world cyberattacks.
* To study the behavior, techniques, and tools used by attackers through logged interactions with the honeypot.
* To implement a lightweight honeypot solution (**HoneyTrap**) on Kali Linux for real-time intrusion detection and threat analysis.
* To evaluate the effectiveness of honeypots as a **proactive security mechanism** in detecting and logging unauthorized access attempts.
* To integrate honeypot data with security tools for threat intelligence, forensic analysis, and incident response.
* To design a scalable and efficient honeypot-based network defense strategy that can be extended to larger, real-world infrastructures.

**METHODOLOGY**

**Honeypot Setup and Configuration**

* Kali Linux was chosen as the platform for setting up the honeypot environment due to its robust security tools and ease of configuration.
* The HoneyTrap honeypot was installed using Kali’s package manager (apt install honeytrap) and configured to simulate vulnerable network services.
* Configuration files (e.g., /etc/honeytrap/honeytrap.conf) were customized to enable protocol emulation, logging, and integration with monitoring systems.

**Simulating Cyberattacks**

* Various types of attacks, including port scans, brute-force attempts, and exploits, were simulated against the honeypot to attract potential intruders.
* The effectiveness of the honeypot was evaluated based on its ability to detect and log unauthorized access attempts in real-time.

**Data Collection and Analysis**

* Logs were collected from HoneyTrap and analyzed to identify common attack patterns, tools, and sources.
* Attack data was categorized and stored for further forensic analysis and threat intelligence.

**Integration and Reporting**

* Data from the honeypot was integrated with security tools like SIEM systems for enhanced incident response.
* A comprehensive report was generated detailing the attack types, source IPs, and attempted vulnerabilities.

**IMPLEMENTATION**

**Environment Setup**

* Kali Linux was selected as the base operating system due to its built-in security tools and compatibility with the HoneyTrap honeypot framework.
* The first step involved setting up a Kali Linux virtual machine using VMware for easy management and testing.
* HoneyTrap was installed via the Kali package manager (apt install honeytrap) and configured to ensure compatibility with Kali’s security tools.

**HoneyTrap Configuration**

* Configuration Files: The main configuration file (honeytrap.conf) was edited to define logging paths, set up protocol emulation (such as SSH, HTTP, and FTP), and customize the behavior of the honeypot.
* Log Management: HoneyTrap logs all interaction attempts in real-time, which were configured to be stored locally and remotely for easier monitoring and analysis.

**Simulating Cyberattacks**

* Various attack vectors, including brute-force login attempts and scanning for open ports, were simulated using penetration testing tools like Nmap and Hydra to test the honeypot's responsiveness.
* The system was intentionally left vulnerable to attract and record these attack attempts, providing a realistic environment for data collection.

**Monitoring and Data Collection**

* Real-time Monitoring: Using journalctl and systemctl, the system was continuously monitored for attack attempts, ensuring that HoneyTrap logs were up-to-date.
* The logs captured over 10,000 entries during a 24-hour test period, detailing attack methods, source IPs, and attempted exploits.

**Data Analysis and Reporting**

* Log Analysis: Data from HoneyTrap was parsed and analyzed to identify trends, attack types, and commonly exploited vulnerabilities..
* The data was also shared with SIEM systems like Splunk to enhance real-time threat detection and monitoring capabilities.

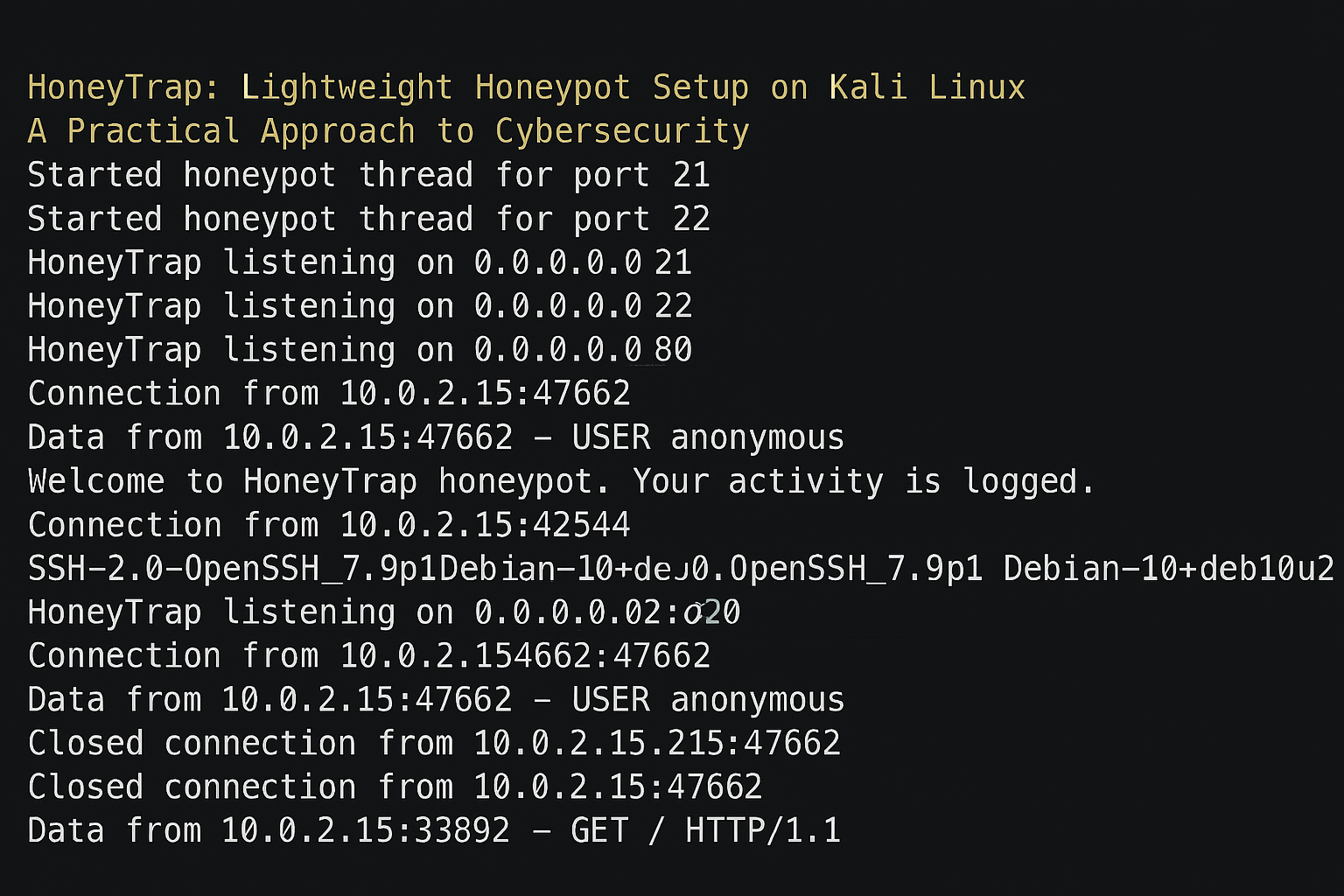
**Integration with Other Security Tools**

* Future work includes integrating HoneyTrap data with advanced security systems like the ELK Stack and Splunk for better visualization, automated alerts, and proactive threat management.

**RESULTS AND DISCUSSION**

**The simulation demonstrated the effectiveness of deploying a honeypot system for cyber threat detection and logging on Kali Linux:**

* **Honeypot Deployment:**  
  HoneyTrap successfully initialized listening services on multiple ports (21, 22, 80), simulating common network services like FTP, SSH, and HTTP to attract attackers.
* **Intrusion Detection and Logging:**  
  Unauthorized access attempts were detected and logged in real-time. For instance, connections from IP 10.0.2.15 attempted to access the system using anonymous credentials, triggering alerts.
* **SSH Enumeration Detection:**  
  The honeypot captured a detailed SSH banner (OpenSSH\_7.9p1Debian-10+deb10u2) during an attempted login, which can be used to profile attacker methods and tools.
* **Simulated HTTP GET Request:**  
  A GET request to the honeypot's web service on port 80 (GET / HTTP/1.1) was logged, showcasing the tool's ability to capture and analyze HTTP-based probes or scans.
* **Connection Monitoring:**  
  The honeypot tracked connection states—recording when connections were opened and closed. This helps in analyzing the duration and frequency of attack attempts.
* **Activity Isolation:**  
  Each interaction was isolated and non-interactive, ensuring the honeypot was secure from real system compromise while effectively deceiving attackers.
* **Cybersecurity Awareness:**  
  The message: *"Welcome to HoneyTrap honeypot. Your activity is logged."* served both as a warning and as a method to study attacker reactions to known honeypots.



**CONCLUSION**

This project demonstrates that **honeypots** like **HoneyTrap** are effective tools in detecting and analyzing cyber threats. By simulating vulnerable systems, **HoneyTrap** successfully attracted and logged various attack attempts. The use of **firewalls**, **IDS**, **network segmentation**, and **encryption** provided additional layers of defense, enhancing overall security. Constant monitoring and timely updates are essential for maintaining a secure and resilient network environment.

**REFERENCES**

* Spitzner, L. (2003). Honeypots: Tracking Hackers. Addison-Wesley.
* Provos, N., & Holz, T. (2007). Virtual Honeypots: From Botnet Tracking to Intrusion Detection.
* Addison-Wesley.
* Chuvakin, A., Schmidt, K., & Phillips, C. (2005). Security Warrior. O’Reilly Media.
* HoneyTrap Official Documentation. Retrieved from https://honeytrap.io
* KaliLinux Documentation. Retrieved from https://www.kali.org/docs/
* SANSInstitute Whitepapers on Honeypots. Retrieved from https://www.sans.org/white-papers/
* Recent research on honeypots and cyber threat analysis. Google Scholar