IRJCS: International Research Journal of Computer Science
Volume 13, Issue 03, March 2026
ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.17

Design and Implementation of a Honeypot Network for Malware Collection and Analysis

D.Amuthvalli
Assistant Professor, Department of CSE (Cyber security)
Sengunthar Engineering College (Autonomus), Tiruchengode, India
amuthavallicse@gmail.com
https://orcid.org/0009-0005-6256-8579
Devajanani V, Nikitha M, Nishandhi P, Oviya S
Department of Computer Science and Engineering (Cyber security)
Sengunthar Engineering College (Autonomus),Tiruchengode, India

Publication History
BibTeX Key Devajanani@2026Design
Orcid: https://orcid.org/0009-0004-9398-7488
Manuscript Reference: IRJCS/RS/Vol.13/Issue03/CSMR26.MRCS10096
Research Article | Open Access | Double-Blind Peer Reviewed Article ID: IRJCS/RS/Vol.13/Issue03/CSMR26.MRCS10096
Received: 30, January 2026, Revised: 13, February 2026, Accepted: 28 February 2026 Published Online: 25 March 2026
https://www.irjcs.com/volumes/Vol13/iss-03/17.CSMR26.MRCS10096.pdf
Article Citation: Devajanani,Nikitha,Nishandhi,Oviya(2026),Design and Implementation of a Honeypot Network for Malware Collection and Analysis, IRJCS: International Research Journal of Computer Science, Volume 13,Issue 03 of 2026 pages 196-199 **Doi:>** https://doi.org/10.26562/irjcs.2026.v1303.17
IRJCS papers should be cited as IRJCS (International Research Journal of Computer Science, AM Publications, India 2026, ISSN 2393-9842, https://doi.org/10.26562/ijrcs.2025.v1303.17 The journal's official abbreviation is IRJCS.

Abstract: Cyber threats are increasing rapidly due to the expansion of internet connectivity and digital services. Modern attackers use automated tools, malware frame works, and botnets to compromise vulnerable systems. Traditional security solutions such as firewalls and intrusion detection systems mainly focus on blocking attacks but often fail to analyze attacker behavior. This paper proposes a honeypot based monitoring system designed for malware collection and cyber attack analysis. The system simulates vulnerable services such as SSH, FTP, and HTTP to attract attackers. All interactions are logged and stored in a centralized database for further analysis. The proposed system provides a monitoring dashboard to visualize attack statistics and patterns. Experimental results show that the system successfully captures brute-force attacks, automated scanning, and malware payload downloads. The proposed system provides a valuable platform for cyber security research and threat intelligence.
Index Terms: Honeypot, Malware Analysis, Cybersecurity, Network Security, Threat Intelligence, Intrusion Monitoring

I. INTRODUCTION

The growth of digital technology has significantly increased the number of internet-connected devices. While this development has improved communication and accessibility, it has also created numerous cybersecurity challenges. Cyber attackers continuously scan networks for vulnerable systems to exploit. Traditional security mechanisms focus mainly on preventing attacks rather than studying attacker behavior. Understanding attacker strategies is crucial for improving cyber security defenses. Honey pots provide an effective solution for studying cyber threats by acting as decoy systems that attract attackers. A honeypot is a computer system designed to appear vulnerable in order to lure attackers. When attackers interact with the honeypot, their actions are recorded and analyzed. This allows researchers to study attack techniques and malware behavior. This project focuses on designing and implementing a honeypot network capable of collecting malware samples and analyzing attacker behavior in real time.

II. RELATED WORK

Several studies have explored honeypot technologies for cyber security research. Honeypots can be classified into two categories:

- Low Interaction Honeypots
- High Interaction Honeypots

Low interaction honeypots simulate limited services and capture basic attack data such as login attempts. High inter-action honeypots provide full operating system environments that allow attackers to interact with real systems. Research shows that honeypots are useful for studying botnet activity, malware propagation, and network scanning attacks.

III. SYSTEM ARCHITECTURE

The proposed honeypot monitoring system consists of several components that work together to collect and analyze cyber attack data.

- Honeypot Service Layer
- Logging and Monitoring Module
- Malware Collection Engine
- Data Analysis Engine

ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.17

- Admin Monitoring Dashboard

The honeypot services simulate vulnerable network services to attract attackers. When attackers interact with these services, their actions are recorded by the logging module. The collected data is then analyzed and visualized through the dashboard.

IV. SYSTEM SPECIFICATIONS

A. Hardware Requirements

- Processor: IntelCorei5 or higher
- RAM: 8GB minimum
- Storage: 256GB SSD
- Network Interface Card

B. Software Requirements

- Operating System: Ubuntu Linux
- Programming Language: Python
- Database: MySQL
- Backend Framework: Flask
- Frontend: HTML, CSS, JavaScript

V. MODULE DESCRIPTION

The system consists of several modules:

A. Honeypot Deployment Module

Deploys simulated services such as SSH, FTP, and HTTP to attract attackers.

B. Logging Module

Records attacker activities including login attempts, commands executed, and connection timestamps.

C. Malware Collection Module

Captures malicious files downloaded by attackers and stores them in the database.

D. Analysis Module

Processes collected logs and identifies attack patterns.

E. Dashboard Module

Displays attack statistics and visualizations.

METHODOLOGY

The honeypot system is deployed in a controlled network environment. When attackers attempt to connect to vulnerable services, their actions are recorded. The collected data is analyzed to identify patterns such as brute-force attacks and malware propagation.

SYSTEM IMPLEMENTATION

The system is implemented using Python for backend processing. Flask is used as the backend framework for managing server communication. MySQL is used to store captured logs and malware samples. The dashboard interface is developed using web technologies such as HTML, CSS, and JavaScript.

SYSTEMTESTING

Several testing methods were used to validate the system:

A. Unit Testing

Each module wastested independently.

B. Integration Testing

Ensured proper communication between system modules.

C. Security Testing

Simulated attacks such as brute-force login attempts and port scanning.

RESULTS AND DISCUSSION

The honeypot system successfully captured several types of cyber attacks including:

- SSH brute-force attacks
- Automated bots canning
- Malware downloads
- Command injection attempts

The results demonstrate the effectiveness of honeypots in studying cyber threats.

NOVELTY OF THE PROPOSED SYSTEM

The proposed system introduces several improvements over traditional honeypot implementations.

- Integration of multiple honeypot services
- Real-time monitoring dashboard
- Automated malware collection
- Centralized attack data analysis
- Scalable architecture

THREAT MODEL

In order to evaluate the effectiveness of the proposed honeypot network, a threat model was defined. The threat model identifies potential attackers, their capabilities, and the possible attack vectors that can be used to compromise network systems.

ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.17

The primary threat actors considered in this system include:

- Automated Bots
- Script Kiddies
- Malware Distribution Networks
- Advanced Persistent Threat (APT) attackers

Automated bots perform large-scale internet scanning to identify vulnerable systems. Script kiddies use publicly avail- able hacking tools to attempt unauthorized access to servers. Malware distribution networks aim to install malicious soft- ware on vulnerable systems to create bot nets or steal sensitive information. The honeypot system is designed to attract these attackers by exposing simulated services that appear vulnerable. When attackers interact with these services, the system captures valuable data including attacker IP addresses, attack methods, login attempts, and payload downloads. This information can be used for threat intelligence analysis and cyber security research.

ATTACK DETECTION ALGORITHM

The honeypot monitoring system uses a simple attack detection algorithm to analyze suspicious behavior.

The algorithm works as follows:

1) Monitor incoming network traffic.
2) Identify connection attempts to honeypot services.
3) Log all attacker interactions.
4) Detect abnormal behavior such as repeated log in failures.
5) Store attack information in the database.
6) Generate alerts for the administrator.

The algorithm continuously monitors network activity and automatically records malicious interactions. By analyzing patterns in the collected data, security analysts can identify common attack techniques.

MATHEMATICAL MODEL

The honeypot monitoring system can be mathematically represented using a set of parameters that describe network interactions. Let:

- A represent the set of attackers.
- H represent the honey pot system.
- L represent the log data collected.
- M represent the malware samples collected.

The interaction between attackers and the honeypot system can be represented as: $A+H \rightarrow L+M$ Where attacker interactions with the honeypot produce logs and malware samples.

The probability of attack detection can be expressed as: $P=\frac{N_a}{N_t}$ Where:

- N_a represents the number of attacks captured.
- N_t represents the total number of attack attempts.
- Higher detection probability indicates improved system effectiveness.

PERFORMANCE EVALUATION

The performance of the honeypot monitoring system was evaluated based on several metrics.

A. Attack Capture Rate

The attack capture rate measures how effectively the honeypot system records malicious activities. During the testing phase, the system captured a large number of unauthorized login attempts and automated scanning activities.

B. System Response Time

The system response time measures how quickly the hon-ypot responds to attacker interactions. The results show that the honeypot services respond quickly to connection attempts, ensuring that attackers continue interacting with the system.

C. Data Storage Efficiency

All captured logs and malware samples are stored in a structured database. Efficient data storage ensures that large volumes of attack data can be analyzed without affecting system performance.

ADVANTAGES OF THE PROPOSED SYSTEM

The proposed honeypot monitoring system offers several advantages:

- Provides real-time monitoring of cyber attacks.
- Collects malware samples for research purposes.
- Helps cybersecurity analysts understand attacker behavior.
- Improves threat intelligence capabilities.
- Supports scalable deployment in different network environments.

These advantages make the proposed system a valuable tool for cyber security research and education.

LIMITATIONS OF THE SYSTEM

Although the proposed system provides several benefits, there are some limitations:

- Skilled attackers may detect honey pot environments.
- High interaction honey pots require strong isolation mechanisms.
- Large amounts of log data require efficient analysis tools. Future research can address these limitations by integrating artificial intelligence techniques for automated threat detection.

ISSN: 2393-9842
https://www.irjcs.com/archives
https://doi.org/10.26562/irjcs.2025.v1303.17

FUTURE WORK

Future work will focus on improving the capabilities of the honeypot monitoring system. Several enhancements are planned for future development.

- Integration of machine learning for automated attack detection.
- Real-time threat intelligence sharing.
- Cloud-based honey pot deployment.
- Advanced malwares and box analysis.
- Visualization dashboards for attack analytics.

These improvements will make the honey pot system more effective in identifying and analyzing modern cyber threats.

ACKNOWLEDGMENT

The author would like to thank the faculty members and mentors who provided valuable guidance during the development of this project. Their support and encouragement contributed significantly to the successful completion of this research work.

CONCLUSION

This paper presented the design and implementation of a honey pot monitoring system for malware collection and analysis. The system effectively captures cyber attack activities and provides valuable insights into attacker behavior.
Future work will focus on integrating artificial intelligence techniques for automated threat detection.

REFERENCES

1. N.Provos and T.Holz,Virtual Honeypots: From Botnet Tracking to Intrusion Detection.
2. L.Spitzner, Honeypots: Tracking Hackers.
3. The Honeynet Project, Know Your Enemy: Learning About Security Threats.