Project Title:
SIEM-Enabled Honeypot Monitoring System

Objective

To design and implement a custom honeypot that simulates vulnerable network services, captures attacker activities, and integrates with a Security Information and Event Management (SIEM) platform for real-time monitoring, visualization, and analysis.

Problem Statement

Cyber attackers are continuously probing networks for vulnerabilities. Traditional defense systems (firewalls, IDS/IPS) may detect attacks but often fail to provide insights into attacker behavior before a full compromise occurs. Off-the-shelf honeypots are limited in flexibility, while SIEM deployments often lack attacker-contextual data.
A custom honeypot integrated with SIEM provides both deception and actionable intelligence, allowing organizations to study real-world attack methods and strengthen their security posture.

Proposed Approach

- Develop a **Python-based honeypot** simulating SSH and HTTP services.
- Capture:
  - Connection metadata (IP, port, timestamp)
  - Login attempts (username/password)
  - Commands or payloads entered by attacker
- Store activity in structured JSON logs.
- Use **Filebeat** to forward logs to **Elasticsearch**.
- Build **Kibana dashboards** for visualization (attacker IP trends, command frequency, geo-location).
- Ensure modular architecture so more services (FTP, SMB) can be added later.
- Deploy honeypot inside isolated VM/Docker for safety.

Technologies & Tools

- **Programming:** Python (Socket Programming, JSON logging)
- **Monitoring:** Elasticsearch, Logstash, Kibana (ELK Stack)
- **Log Forwarding:** Filebeat/Syslog
- **Platform:** Linux (Ubuntu VM)
- **Version Control:** GitHub

Expected Outcomes

- A functioning honeypot logging intrusion attempts.
- Real-time SIEM dashboards for monitoring attacks.
- Enhanced understanding of attacker behaviour, commands, and trends.
- Documentation and code base for future research or production deployment.