Honeypot + Threat Intelligence Dashboard

❖ **Project Summary:**

I created and deployed a Python-based honeypot and threat intelligence dashboard using SQLite and Flask to identify, monitor, and display unauthorised access attempts and malicious activity in real-time.

This project simulates a vulnerable setup that attracts attackers, records their activities, and enriches the collected data with threat intelligence, providing cybersecurity experts with actionable insights. information to improve defences.

❖ Key Features:

✅ Honeypot Simulation

- An imitation of a service environment used to lure unauthorized access attempts.
- Records attacker IP addresses, geolocation, timestamps, and user agents.

✅ Threat Intelligence Lookup

- Automated lookup of IP reputation.
- Enhances captured IPs with third-party threat data (AbuseIPDB, Virus Total-ready format).

✅ Severity Scoring System

- Assigns each identified threat a severity score based on:
- Reputation.
- Count of attack attempts.
- Country of origin.
- Known malicious behaviour.
- Data is color-coded for simple human understanding.

✅ Real-Time Dashboard

- Built on Flask and Bootstrap for a clean visualization.

Displays:

- Identified threats.
- Severity scores.
- IP and geo-location information.

Harshini Surabhi
https://www.linkedin.com/in/sriharshini-surabhi-054566215/

- Timestamped logs.I'm thrilled to share my latest cybersecurity project —
  🛡️ **Honeypot + Threat Intelligence Dashboard**
- This project was designed to simulate a deceptive environment that **traps unauthorized access attempts**, captures attacker data, and visualizes threat patterns in real-time.
-

✅ **Map Visualization**

- Uses Python's folium library to plot attack origins on a world map.
- Gives an intuitive geographical understanding of threat sources.

✅ **SQLite Database Integration**

Saves all threat information permanently for long-term analysis.

💡 **Tools & Technologies Used:**

- 🐍 Python
- 💾 SQLite
- 🌐 Flask
- 📊 Folium / PyDeck (Map Visualization)
- 💻 HTML, CSS, Bootstrap
- 🔍 Threat Intelligence APIs (Pluggable)
- 💡 Scapy (optional for packet inspection)

🎯 **Learning Outcome:**

This project taught me how to:

- ➢ Simulate deceptive environments using honeypots.
- ➢ Enrich raw data with third-party threat intelligence.
- ➢ Design dashboards for cybersecurity visibility.
- ➢ Apply severity-based threat ranking.
- ➢ Develop backend-to-frontend pipelines for real-time monitoring.

💡 **GitHub Link:**

👉 https://github.com/Harshinisurabh/honeypot-threat-dashboard

Harshini Surabhi
https://www.linkedin.com/in/sriharshini-surabhi-054566215/