Cybersecurity Log Analysis Mini Data Warehouse

Project Overview

This project builds a mini data warehouse focused on analyzing authentication logs to detect security threats such as brute-force login attempts. The system ingests CSV log files, loads data into a PostgreSQL database, and provides analysis and visualization tools to identify suspicious IP addresses and visualize their geographic locations.

Objectives

- Design a relational schema to store authentication logs.
- Create sample log data mimicking real-world authentication attempts.
- Develop ETL (Extract, Transform, Load) scripts to process and load data into PostgreSQL.
- Write SQL queries to detect suspicious activity such as failed logins.
- Visualize top attacker IPs and their geographic locations on an interactive map.
- Enable daily automation and reporting for proactive security monitoring.

Technical Components

1. Data Schema Design
- Table: auth_logs
- Columns: timestamp, username, ip_address, user_agent, login_result, auth_method, location

2. Sample Data Generation
- Synthetic CSV files generated with public IP addresses for geolocation.
- Mixed successful and failed login attempts to simulate attack patterns.

3. ETL Script
- Python script using pandas and psycopg2 to load CSV, create tables, and insert data.

4. Data Analysis
- SQL queries identify failed logins and potential brute-force IPs.

5. Visualization

Cybersecurity Log Analysis Mini Data Warehouse

- Bar charts and geo-mapping of attacker IPs using matplotlib and folium.

6. Automation & Reporting (Future Scope)

- Scheduling, email reports, and alerting.

Challenges & Solutions

- Database permissions issues resolved by running ETL as superuser.
- Column naming mismatch fixed by aligning script and DB schema.
- Used public IPs to enable geo-location since private IPs cannot be geolocated.
- Installed necessary Python libraries for smooth script execution.

Outcome

- Working mini data warehouse for auth logs.
- Efficient data ingestion.
- Meaningful security queries.
- Interactive visualizations.
- Solid foundation for expansion.

Future Enhancements

- Support for other log types (firewall, network traffic).
- Anomaly detection using machine learning.
- Real-time alerting pipelines.
- Web dashboard for live monitoring.

Conclusion

This project demonstrates a practical approach to cybersecurity log analysis using open-source tools and APIs, providing actionable security insights.