1.Title of the Project

Threat Intelligence Platform

2. Abstract

The Threat Intelligence Platform is a security-based project designed to collect, analyze, and visualize cyber threat data to help organizations identify potential risks. The system aims to detect suspicious activities, such as phishing links, malicious IP addresses, and data breaches, by gathering information from multiple open-source feeds and APIs. Using Python, the platform will automate data collection and analysis, generate threat reports, and display insights through a simple dashboard. The objective is to support cybersecurity teams in making faster and more informed decisions to protect their systems and networks. The expected outcome is a user-friendly, Python-based tool that helps in real-time threat monitoring, incident analysis, and prevention of cyberattacks.

3. Introduction

➢ Background:

Cyber threats are increasing daily, and manual monitoring is time-consuming. Organizations need automated systems that can analyze and report security risks efficiently.

➢ Importance and Relevance:

A Threat Intelligence Platform helps detect, analyze, and respond to cyber threats quickly, reducing security risks and protecting data.

➢ Problem Statement:

Current monitoring systems are expensive and complex. Small businesses lack affordable tools for collecting and analyzing threat data.

➢ Objectives:

- Build an automated threat data collection tool using Python.
- Analyze and categorize potential cyber threats.
- Display results in an easy-to-understand dashboard.

4. Literature Review

➢ Existing Systems Summary:

Many commercial platforms like IBM X-Force and ThreatConnect provide threat intelligence, but they are paid and complex.

➢ Key Findings:

These systems use real-time data, APIs, and automation for detection but are often enterprise-level tools.

➢ Gaps Identified:

Lack of open-source, easy-to-use platforms for small-scale or educational use.

5. Proposed Solution / Methodology.

➢ Description:

The proposed solution is a Python-based Threat Intelligence Platform that collects data from open APIs like VirusTotal, AbuseIPDB, and AlienVault OTX. It will analyze and display results such as malicious domains or IPs.

➢ Tools, Technologies, Platforms:

Python

APIs (VirusTotal, AbuseIPDB, OTX)

Flask (for web interface)

SQLite (for local database)

Matplotlib / Plotly (for visualization)

➢ Project Phases:

1. Data Collection: Fetch threat data via APIs.

2. Design: Create system architecture and database.

3. Development: Implement backend scripts and web dashboard.

4. Testing: Validate API responses and data accuracy.

5. Evaluation: Check performance and usability.

➢ Flowchart (optional):

API Data → Data Analysis → Database → Dashboard Visualization

6. Implementation Plan

➢ Timeline (Week-wise):

Week 1: Research and API integration setup

Week 2: Database and data storage design

Week 3: Data analysis logic and report generation

Week 4: Web interface using Flask

Week 5: Testing and final evaluation

➢ Roles & Responsibilities (if team):

Developer: API integration and backend logic

Designer: Dashboard and UI

Tester: Bug testing and performance review

➢ Milestones & Deliverables:

Working API-based data collector

Threat analysis dashboard

Final documentation

7. Expected Outcomes

A working Python-based tool for real-time threat monitoring

Clear visualization of threat data

Affordable and easy-to-use security platform

Future Scope: Integration with machine learning for predictive threat detection

8. References

IBM X-Force Exchange

VirusTotal API Documentation

AlienVault OTX API Documentation

Research papers on Cyber Threat Intelligence (IEEE)