**Cyber Threat Intelligence Dashboard Report**

**Introduction:**

* Cyber threats are becoming increasingly sophisticated, demanding robust monitoring and swift response mechanisms.
* The Cyber Threat Intelligence Dashboard (CTI Dashboard) addresses this need by offering real-time visibility into security threats.
* The dashboard focuses on analyzing Indicators of Compromise (IOCs) such as malicious IP addresses.
* It leverages open-source intelligence tools and public APIs to enrich and verify threat data.
* The primary objective is to enhance threat detection, automate IOC analysis, and accelerate incident response for security analysts.

**Abstract:**

* The CTI Dashboard is a lightweight and efficient system focused on cyber threat data analysis.
* It retrieves and visualizes threat intelligence specifically related to IP addresses.
* Built using Flask for the backend and MongoDB for data storage, ensuring fast and scalable performance.
* Operates within Docker containers to maintain isolated and consistent environments.
* Integrates with VirusTotal and AbuseIPDB APIs to fetch threat scores and abuse history.
* Offers a user-friendly interface for submitting IPs and receiving instant analysis.
* Streamlines cybersecurity operations by automating IOC validation and report generation.

**Tools Used:**

* Python & Flask – Backend framework for the web app
* MongoDB – NoSQL database for storing IOC records
* Docker & Docker Compose – Containerization and orchestration
* VirusTotal API – For scanning and threat reputation
* AbuseIPDB API – For checking IP abuse history
* Jinja2 & HTML – Templating and UI presentation
* .env File – For securely managing API keys and secrets

**Steps Involved in Building the Project:**

1. **Project Initialization:**
   * Directory structure created using Flask factory pattern.
   * Created required modules and folders (app/, templates/, apis/, etc.).
2. **Environment Setup:**
   * Installed dependencies via requirements.txt.
   * Configured Flask app and MongoDB integration.
3. **Dockerization:**
   * Defined services in docker-compose.yml.
   * Created a custom Dockerfile for Flask app deployment.
4. **API Integration:**
   * Integrated VirusTotal and AbuseIPDB for IOC lookups.
   * Handled response parsing and JSON data binding.
5. **Dashboard UI:**
   * Developed dashboard.html for displaying IOC results.
   * Displayed metadata like timestamp, threat scores, and reputation.
6. **Testing and Debugging:**
   * Container logs analyzed via Docker output.
   * Verified end-to-end workflow with real IOC lookups (e.g., 8.8.8.8, 45.83.64.1).

**Conclusion:**

This project successfully delivers a foundational cyber threat intelligence dashboard that combines real-time threat feeds, efficient data storage, and a clean user interface. Its modular and containerized design enables easy expansion, such as integrating more APIs, exporting reports, or adding authentication. The CTI Dashboard stands as a powerful starting point for security automation and threat analysis.