Threat Intelligence Aggregator (Non-AI)

is a project submitted in partial fulfillment of the requirements for the

Internship Role: Cyber Security Intern

Organization: Unified Mentor

Prepared By: Tanisha Sagar

Project Type: Defensive Cyber Security / Blue Team Project

Duration of Internship: 3 months

INDEX

INTRODUCTION

Threat intelligence plays a crucial role in modern cybersecurity defense. Organizations receive Indicators of Compromise (IOCs) such as malicious IP addresses, domains, URLs, and file hashes from multiple threat intelligence sources. However, these feeds are often unstructured, inconsistent, and difficult to correlate manually.

This project focuses on developing a **Threat Intelligence Aggregator** that collects threat data from multiple sources, normalizes the indicators, correlates them across feeds, and prioritizes high-risk indicators. The project is implemented without using artificial intelligence, relying instead on logical correlation techniques commonly used in Security Operations Centers (SOC).

PROJECT OVERVIEW

The Threat Intelligence Aggregator is a Python-based tool designed to process multiple threat intelligence feeds provided in different formats such as TXT, CSV, and JSON. The system extracts IOCs from these feeds, classifies them into categories, and identifies indicators that appear in more than one source.

Indicators detected across multiple independent sources are treated as higher risk and are exported separately to assist security teams in faster decision-making.

PROBLEM STATEMENT

Threat intelligence data is often scattered across multiple sources with different formats and structures. Manually analyzing these feeds is time-consuming and error-prone. Without proper correlation, security teams may fail to identify truly high-risk indicators or may respond to false positives.

The problem addressed by this project is the lack of a simple, automated method to aggregate, correlate, and prioritize threat intelligence indicators from heterogeneous sources.

PROJECT OBJECTIVES

The main objectives of this project are:

- To collect threat intelligence indicators from multiple sources
- To normalize different feed formats into a unified structure
- To classify indicators into IPs, domains, URLs, and hashes
- To correlate indicators appearing in multiple feeds
- To assign severity levels based on cross-feed occurrence
- To generate actionable reports and blocklists

SCOPE OF THE PROJECT

The scope of this project includes:

- Processing offline threat intelligence feeds
- Supporting TXT, CSV, and JSON formats
- Performing correlation based on source repetition
- Generating multiple output reports for analysis

The project does not include:

- Live threat feed ingestion
- AI or machine learning techniques
- Real-time SOC integration

This ensures the system remains lightweight, transparent, and easy to audit.

TOOLS AND TECHNOLOGIES USED

Programming Language:

- Python 3

Libraries Used:

- `re` for pattern matching
- `csv` and `json` for feed parsing
- `argparse` for command-line argument handling

- `collections.defaultdict` for efficient data storage

Development Environment:

- Visual Studio Code
- Windows Command Prompt

SYSTEM ARCHITECTURE AND WORKFLOW

Workflow Steps:

1. Load threat intelligence feeds from the specified directory
2. Parse IOCs from each feed based on file format
3. Classify indicators into appropriate categories
4. Track the source feeds for each IOC
5. Correlate indicators appearing across multiple sources
6. Assign severity based on occurrence count
7. Generate output files and reports

This modular workflow reflects practical SOC threat intelligence processing.

IMPLEMENTATION DETAILS

The system uses Python functions to handle different feed formats. Each indicator is stored along with the list of sources from which it was extracted. Severity is calculated based on the number of distinct sources reporting the same indicator.

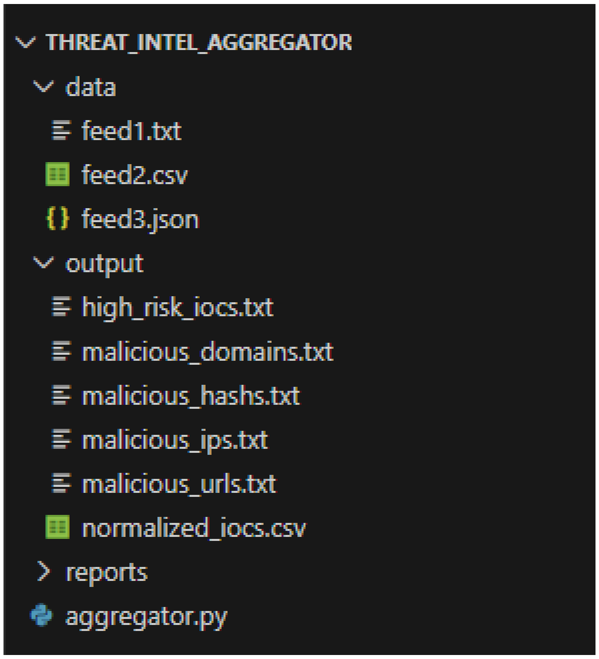
The tool supports command-line execution with an optional data directory argument, making it flexible for different environments.

OUTPUT AND RESULTS

The project generates the following outputs:

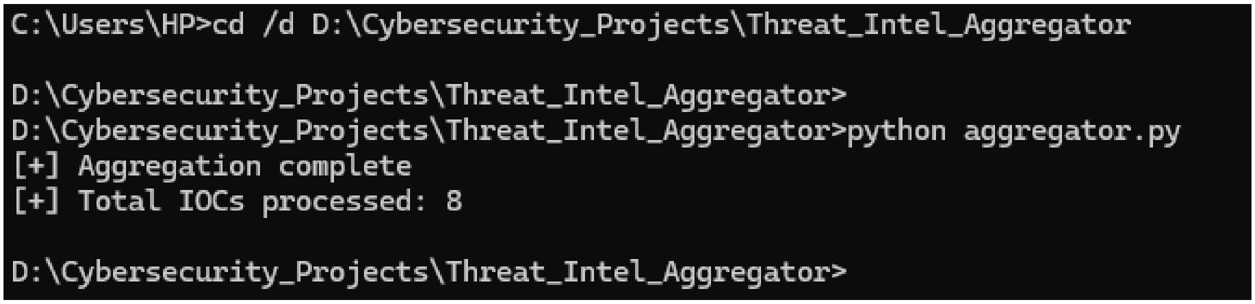
- Category-wise blocklists:
    - Malicious IPs
    - Malicious Domains
    - Malicious URLs
    - Malicious Hashes
- A normalized CSV file containing:
    - IOC
    - Type
    - Occurrence count
    - Severity
    - Source feeds
- A high-risk IOC report identifying indicators present in multiple feeds

These outputs can be directly used for defensive actions such as firewall blocking or SOC investigation.



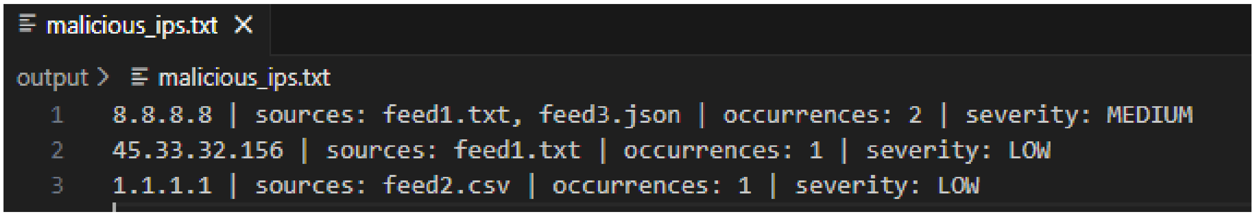
```
∨ THREAT_INTEL_AGGREGATOR
  ∨ data
      feed1.txt
      feed2.csv
      feed3.json
  ∨ output
      high_risk_iocs.txt
      malicious_domains.txt
      malicious_hashs.txt
      malicious_ips.txt
      malicious_urls.txt
      normalized_iocs.csv
  > reports
  aggregator.py
```

Figure 1: Project directory structure showing source code, input data, and generated outputs.



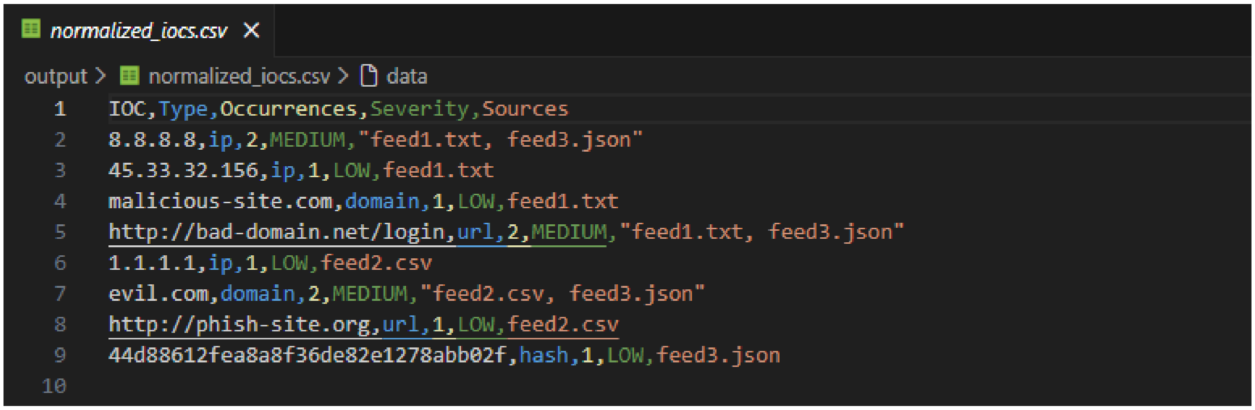
```
C:\Users\HP>cd /d D:\Cybersecurity_Projects\Threat_Intel_Aggregator

D:\Cybersecurity_Projects\Threat_Intel_Aggregator>
D:\Cybersecurity_Projects\Threat_Intel_Aggregator>python aggregator.py
[+] Aggregation complete
[+] Total IOCs processed: 8

D:\Cybersecurity_Projects\Threat_Intel_Aggregator>
```

Figure 2: Execution of the Threat Intelligence Aggregator using Command Prompt.



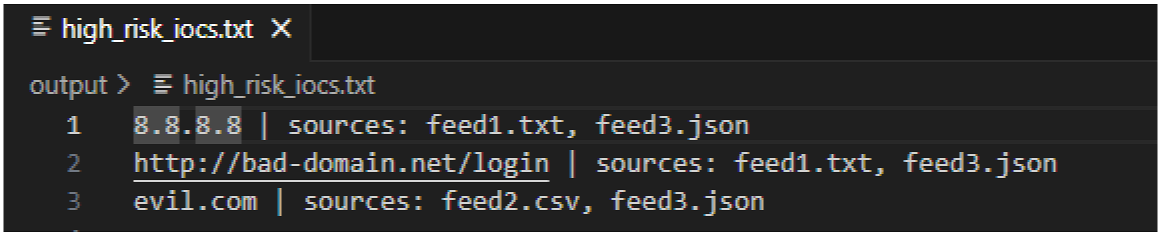
```
8.8.8.8 | sources: feed1.txt, feed3.json | occurrences: 2 | severity: MEDIUM
45.33.32.156 | sources: feed1.txt | occurrences: 1 | severity: LOW
1.1.1.1 | sources: feed2.csv | occurrences: 1 | severity: LOW
```

Figure 3: Sample category-wise blocklist generated by the system.



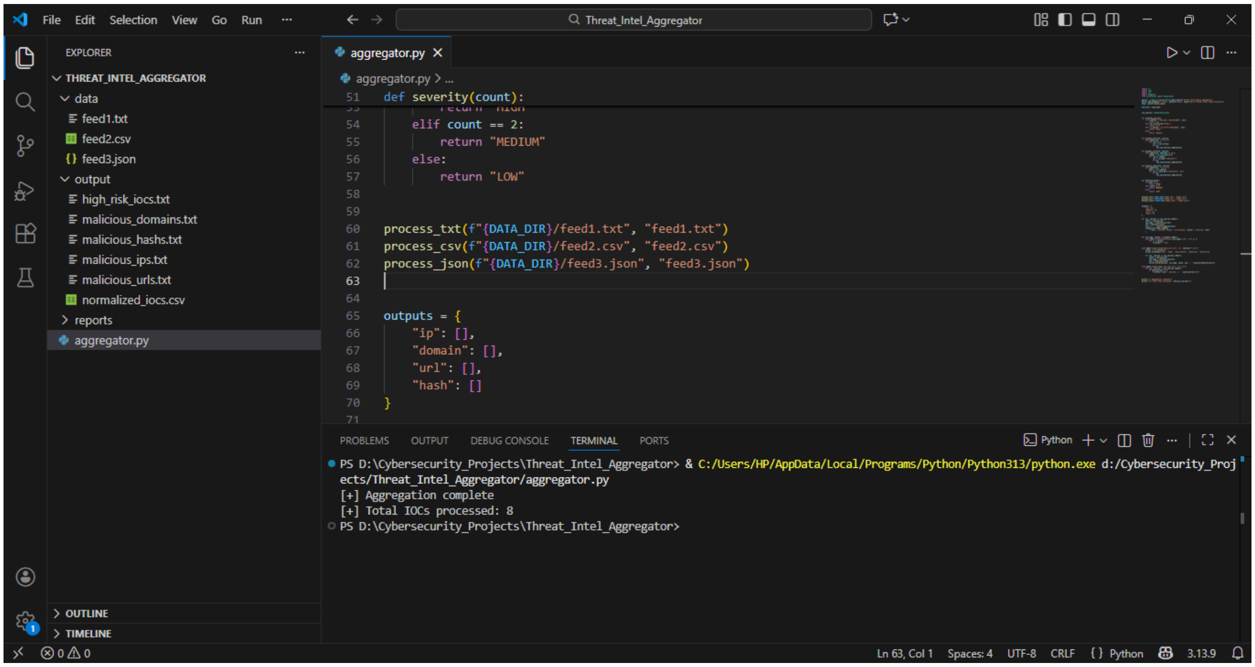
```
IOC,Type,Occurrences,Severity,Sources
8.8.8.8,ip,2,MEDIUM,"feed1.txt, feed3.json"
45.33.32.156,ip,1,LOW,feed1.txt
malicious-site.com,domain,1,LOW,feed1.txt
http://bad-domain.net/login,url,2,MEDIUM,"feed1.txt, feed3.json"
1.1.1.1,ip,1,LOW,feed2.csv
evil.com,domain,2,MEDIUM,"feed2.csv, feed3.json"
http://phish-site.org,url,1,LOW,feed2.csv
44d88612fea8a8f36de82e1278abb02f,hash,1,LOW,feed3.json

```

Figure 4: Normalized CSV output containing correlated IOCs with severity levels.



```
8.8.8.8 | sources: feed1.txt, feed3.json
http://bad-domain.net/login | sources: feed1.txt, feed3.json
evil.com | sources: feed2.csv, feed3.json
```

Figure 5: High-risk IOC report identifying indicators present across multiple feeds.



```python
def severity(count):
    elif count == 2:
        return "MEDIUM"
    else:
        return "LOW"


process_txt(f"{DATA_DIR}/feed1.txt", "feed1.txt")
process_csv(f"{DATA_DIR}/feed2.csv", "feed2.csv")
process_json(f"{DATA_DIR}/feed3.json", "feed3.json")


outputs = {
    "ip": [],
    "domain": [],
    "url": [],
    "hash": []
}
```

```
PS D:\Cybersecurity_Projects\Threat_Intel_Aggregator> & C:/Users/HP/AppData/Local/Programs/Python/Python313/python.exe d:/Cybersecurity_Proj
ects/Threat_Intel_Aggregator/aggregator.py
[+] Aggregation complete
[+] Total IOCs processed: 8
PS D:\Cybersecurity_Projects\Threat_Intel_Aggregator>
```

Figure 6: Core implementation logic of the Threat Intelligence Aggregator.

LEARNING OUTCOMES

Through this project, the following skills were developed:

- Understanding of threat intelligence concepts
- Practical IOC parsing and normalization
- Cross-source correlation techniques
- Python scripting for cybersecurity applications
- SOC-oriented reporting and documentation

CONCLUSION

The Threat Intelligence Aggregator successfully demonstrates how multiple threat intelligence feeds can be combined, analyzed, and prioritized using simple yet effective logic. By correlating indicators across sources, the system reduces noise and highlights high-risk threats.

This project reflects real-world blue-team practices and provides a strong foundation for further development in threat intelligence automation and SOC operations.

Bibliography and References

1. Scarfone, K., & Mell, P. (2012). *Guide to Intrusion Detection and Prevention Systems (IDPS)*. National Institute of Standards and Technology (NIST), Special Publication 800-94.
2. Stallings, W. (2018). *Network Security Essentials: Applications and Standards.* Pearson Education.
3. Behl, A., & Behl, K. (2017). *Cyberwarfare: The Changing Landscape of International Security*. Oxford University Press.
4. Unified Mentor. *Cyber Security Internship Project Guidelines and Documentation.* (Internal internship reference material)
5. Python Software Foundation. *Python 3 Documentation.* https://docs.python.org/3/
6. Mandiant. *Threat Intelligence and Indicators of Compromise (IOC) Overview*. https://www.mandiant.com/resources
7. MITRE Corporation. *Common Cyber Threat Intelligence Concepts.* https://attack.mitre.org/