

**PES UNIVERSITY**  
**Department of Computer Science and Engineering**  
**M.Tech 1<sup>st</sup> SEM 2025**

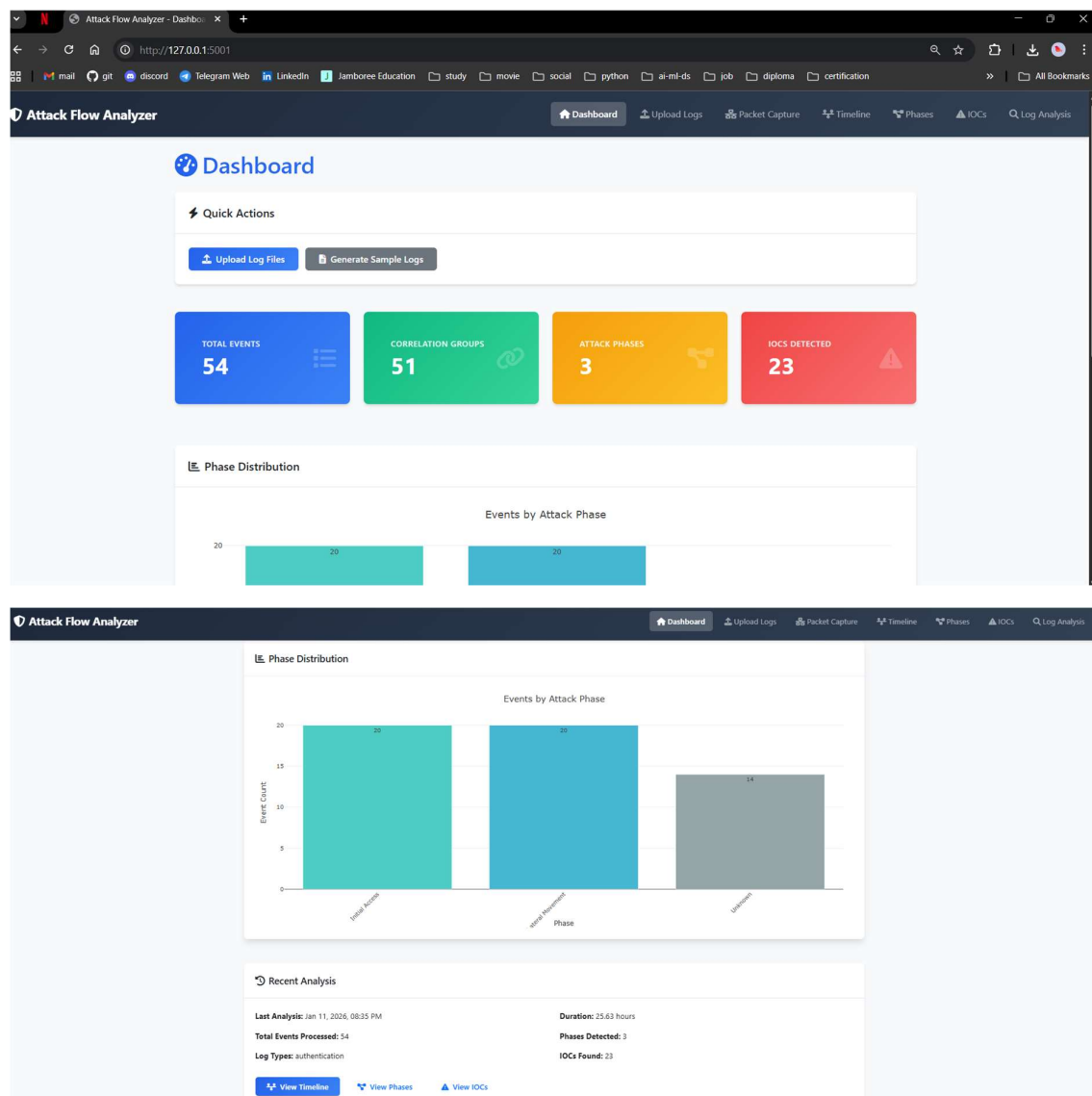
**CYBER SECURITY MINI PROJECT REPORT**

**Title: Incident Response Attack-Flow Analyzer**

**Team Members:**

1. Anarghya B - PES1PG25CS009
2. G Krishna Teja - PES1PG25CS025
3. Niveditha Amarnath - PES1PG25CS045

**Git Link:** [https://github.com/Krish022004/attack\\_flow\\_analyzer](https://github.com/Krish022004/attack_flow_analyzer)



## Project 14: Incident Response Attack-Flow Analyzer

### Concept

Modern cyber-attacks seldom appear as single-step events; rather, they come as multi-stage attack flows-including reconnaissance, exploitation, lateral movement, and data exfiltration. Traditional log monitoring systems mostly support isolated alerts, which make it hard for security analysts to get insight into the sequence, intent, and progression of an attack.

The need is addressed through an attack flow analyzer that correlates system logs, firewall logs, authentication logs, and web server logs to interpret complete attack narratives. Instead of merely looking at the log entries as individual events, the platform analyzes these events over time, patterns, and IOCs to determine the attack types, classify them into attack phases, and present them visually and in a structured format.

To enhance situational awareness, the system offers:

- Distribution of Attack Types on graphical representation to indicate the detected attacks
- Timeline visualization for the different phases of the attacks
- IOC retrieval involving IP addresses, port numbers, and malicious payloads
- Recommendations of action regarding how the attack type identified can be mitigated

By enabling the interpretation of log data in the context of attack Intelligence, the Attack Flow Analyzer facilitates the speedy detection of threats by the security analyst as well as his decisions based on these attacks. The log analysis project, as stated, can be used to improve response techniques in the current cyber security scenario.

### Input

The Attack Flow Analyzer accepts log data and network captures as input for security analysis. The supported input types are:

1. System Log Files
  - Authentication logs (e.g., auth.log)
  - Firewall logs (e.g., firewall.log)
  - Web server logs (e.g., access.log, error.log)
  - General system logs (Format: .log, .txt)
2. Network Packet Capture Files
  - Packet capture files containing network traffic (Format: .pcap, .pcapng)
3. User Inputs
  - Selection and upload of one or more log files through the web interface
  - Triggering the analysis using the Analyze Logs button

Outputs

## **Output**

After processing the input logs, the system generates the following outputs:

- Attack Summary
- Total number of detected attacks
- Severity level of each attack
- Last detected timestamp for every attack type
- Attack Type Distribution.

## **Main Modules**

1. Log Upload Module  
Uploads different log files for analysis.
2. Log Parsing Module  
Reads logs and extracts important fields like IP, time, and action.
3. Attack Detection Module  
Identifies attacks such as brute force, SQL injection, and DDoS.
4. Attack Flow / Timeline Module  
Arranges events in time order to show attack progression.
5. IOC Extraction Module  
Extracts Indicators of Compromise like IPs and URLs.
6. Visualization Module  
Shows results using charts and graphs.
7. Recommendation Module  
Suggests security measures to prevent detected attacks.
8. Reporting Module  
Generates summary and analysis reports.
9. Web Dashboard Module  
Connects frontend and backend using Flask

## **Learning Component**

- Gained practical knowledge of incident response and attack lifecycles.
- Learned to parse and analyze multiple log formats and extract IOCs.
- Developed skills in event correlation, rule-based classification, and timeline visualization.
- Gained experience in Flask web development and building interactive dashboards.
- Learned basics of network packet analysis using Scapy and PyShark.

## **Use Case / Purpose**

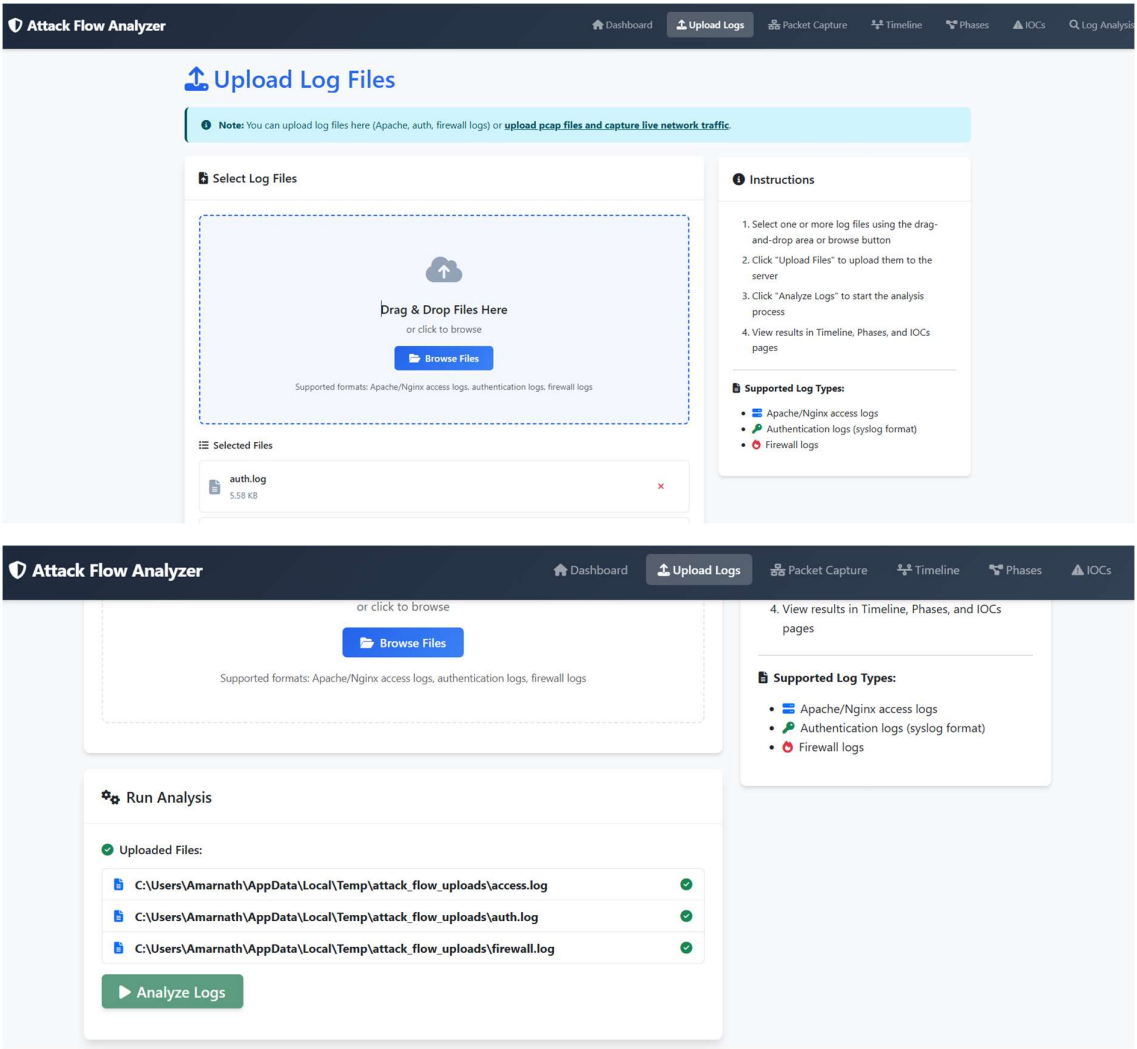
- Small to Medium Enterprises (SMEs): Automate incident response without expensive SIEM tools.
- Security Operations Centers (SOCs): Quickly analyze logs, detect attack phases, and extract IOCs.
- Network Monitoring: Track real-time attacks using packet capture and timeline visualization.
- Incident Investigation: Correlate events across systems to reconstruct attack flows for forensic analysis.

Abstract

The project addresses the challenge of manual incident response analysis by developing an automated log analysis tool that reconstructs end-to-end attack lifecycles from multiple log sources. The system automatically correlates events from authentication logs, web server logs, firewall logs, and network packet captures to classify attack phases and extract Indicators of Compromise (IOCs). The solution provides security analysts with a comprehensive web-based dashboard featuring interactive timelines, phase classification, and IOC export capabilities, enabling efficient incident response without requiring commercial SIEM tools

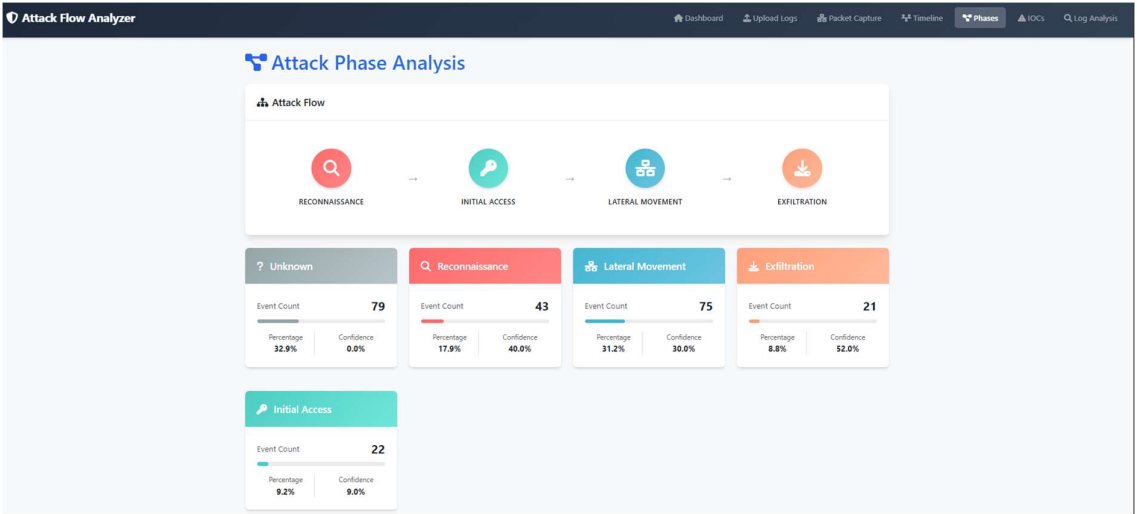
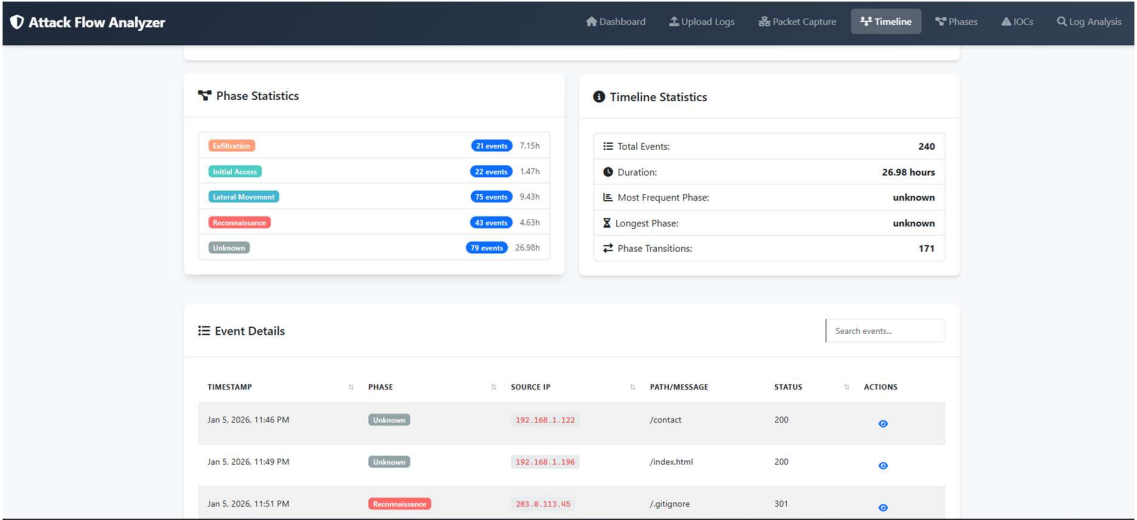
Objectives and Solution

**Objective 1 – Solution:** Multi-source log ingestion and parsing from diverse formats (Apache/Nginx access logs, authentication logs, firewall logs, and .pcap/.pcapng files). Implemented automatic log type detection, timestamp normalization, and regex-based pattern matching for efficient parsing. [Image: Log upload interface showing multiple file types supported]



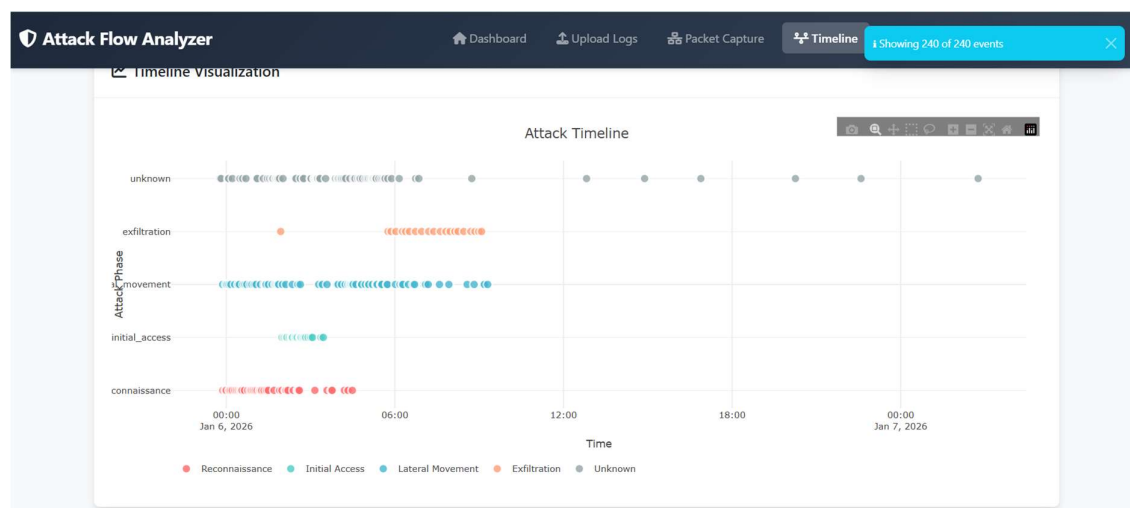
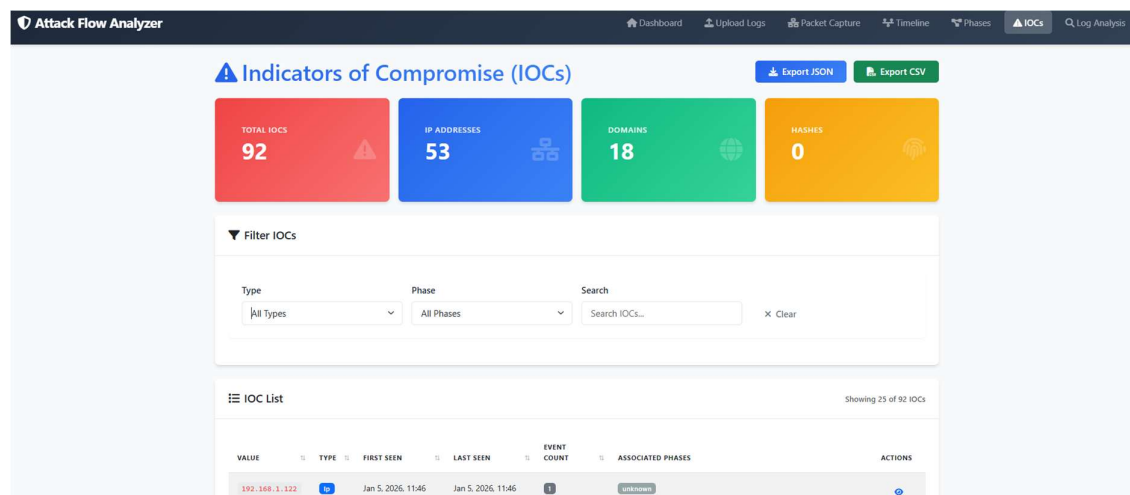
**Objective 2** – Solution: Automated attack phase classification using rule-based system mapping events to MITRE ATT&CK phases (Reconnaissance, Initial Access, Lateral Movement, Exfiltration). Implemented weighted confidence scoring and pattern matching for accurate classification.

[Image: Attack phases dashboard showing phase distribution and classified events]



**Objective 3** – Solution: Interactive timeline visualization and IOC extraction with export functionality. Built Plotly-based timeline showing chronological attack progression and automatic extraction of IPs, domains, URLs, hashes, and user agents with JSON/CSV export.

[Image: Interactive timeline visualization and IOC export interface]



## Problem Statement

Traditional incident response requires security analysts to manually analyse multiple log sources from different systems, making the process time-consuming, error-prone, and inefficient. Analysts struggle to correlate events across different log formats, identify attack progression through various phases, and extract actionable Indicators of Compromise (IOCs). The lack of automated tools for small to medium organizations forces reliance on expensive commercial SIEM solutions, creating a gap in accessible incident response capabilities.

## Proposed Solution

The project provides a web-based Attack-Flow Analyzer that automates incident response. It ingests and parses multiple log formats, correlates events by user, IP, and session, classifies them into attack phases using rule-based patterns, and extracts IOCs. The system displays interactive timelines, phase distributions, and real-time packet analysis on a single dashboard, enabling efficient and accessible security investigations.

The project solves this problem through a comprehensive web-based attack flow analyser that:

1. Ingests and parses multiple log formats automatically
2. Correlates events by user, IP address, and session using hash-based algorithms,
3. Classifies events into attack phases using rule-based pattern matching with confidence scoring
4. Generates interactive timelines showing attack progression
5. Extracts and categorizes IOCs automatically
6. Provides real-time packet capture and analysis capabilities.

The solution integrates all components into a single Flask-based web dashboard, making it accessible and user-friendly for security analysts

### **Tools & Technologies Used**

- Programming Language: Python 3.8+
- Web Framework: Flask 3.0+
- Packet Analysis: Scapy, PyShark, Wireshark
- Visualization: Plotly
- UI Framework: Bootstrap 5
- Libraries: python: dateutil, validators, flask-socketio
- Development Tools: Git, VS Code

### **Algorithms / Techniques Used:**

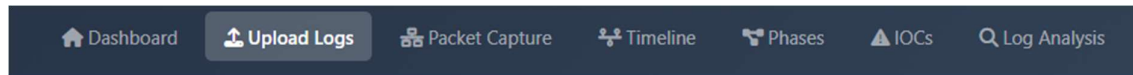
- Regex-based Pattern Matching: For log parsing and IOC extraction using compiled regular expressions for  $O(n)$  complexity
- Hash-based Correlation:  $O(1)$  lookup using dictionary/hash maps to group events by user, IP, and session identifiers
- Time-window Correlation: Groups events within configurable time windows (default 5 minutes) for session-based correlation
- Rule-based Classification: Weighted scoring algorithm that matches event patterns against phase-specific rules, calculating confidence scores (0.0-1.0) for each attack phase
- Event Sorting:  $O(n \log n)$  merge sort for chronological timeline construction
- IOC Deduplication: Set-based deduplication with metadata tracking (first seen, last seen, event count)
- Protocol-aware Packet Parsing: Layer-based packet dissection for TCP, UDP, ICMP, DNS, and HTTP protocols.

### **Implementation Details:**

#### **Modules**

1. Log Ingestion: Parses Apache/Nginx, auth, and firewall logs with automatic format detection.
2. Correlation Engine: Groups events by user, IP, and session using hash maps and time windows.

3. Phase Classifier: Applies rule-based classification with confidence scoring.
4. Timeline Builder: Sorts events chronologically and groups them by attack phase.
5. IOC Extractor: Extracts IPs, domains, URLs, hashes, and user agents using regex.
6. Packet Capture Module: Captures and analyzes packets in real-time with Scapy.
7. Web Dashboard: Displays analysis, timelines, phase distribution, and IOC export in Flask.

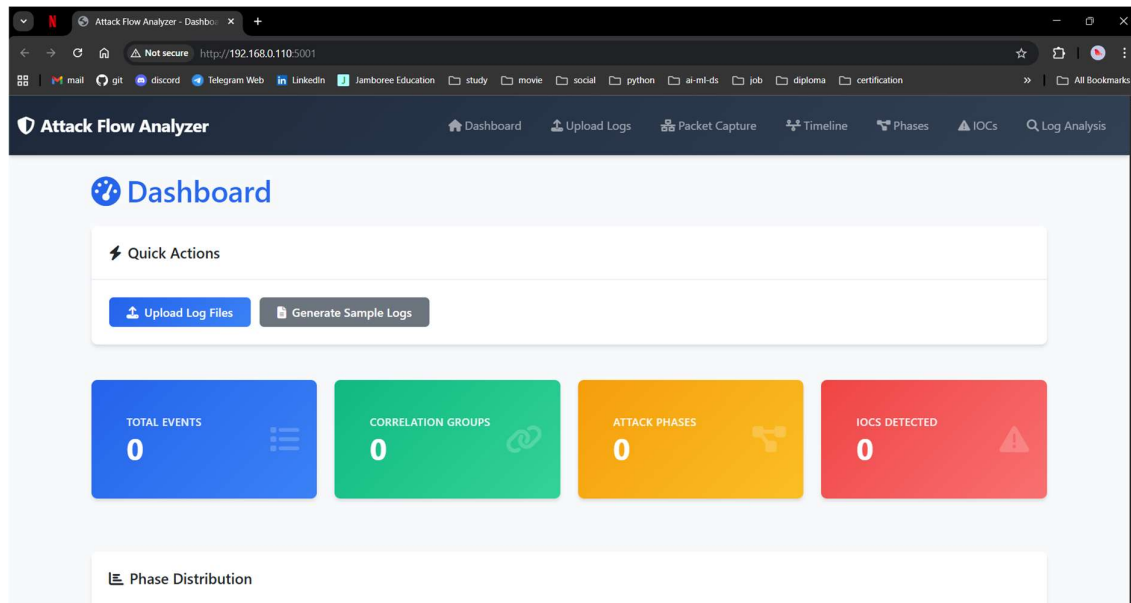


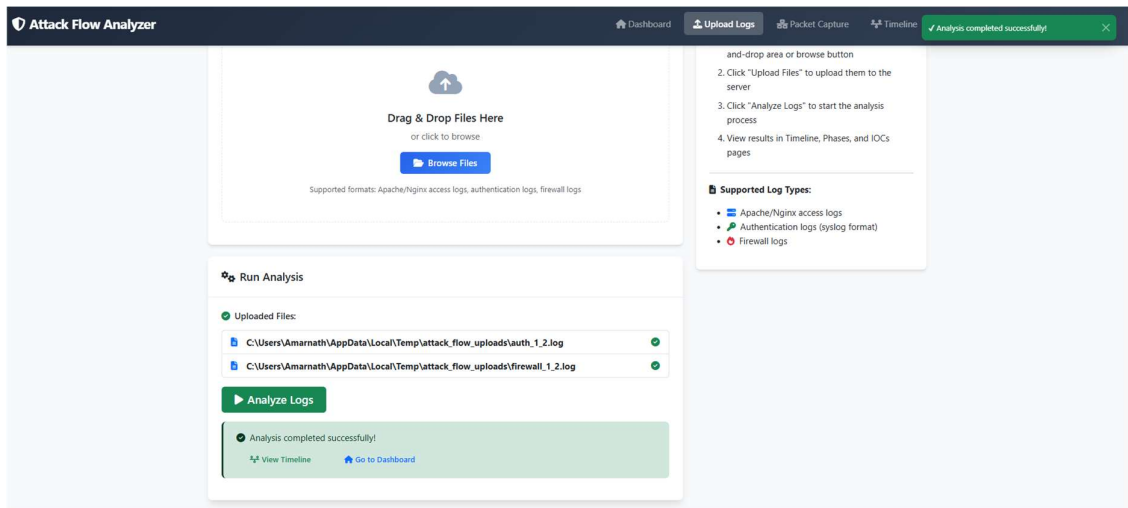
## Workflow

Upload logs → Parse → Correlate events → Classify phases → Build timeline → Extract IOCs → Display results on dashboard.

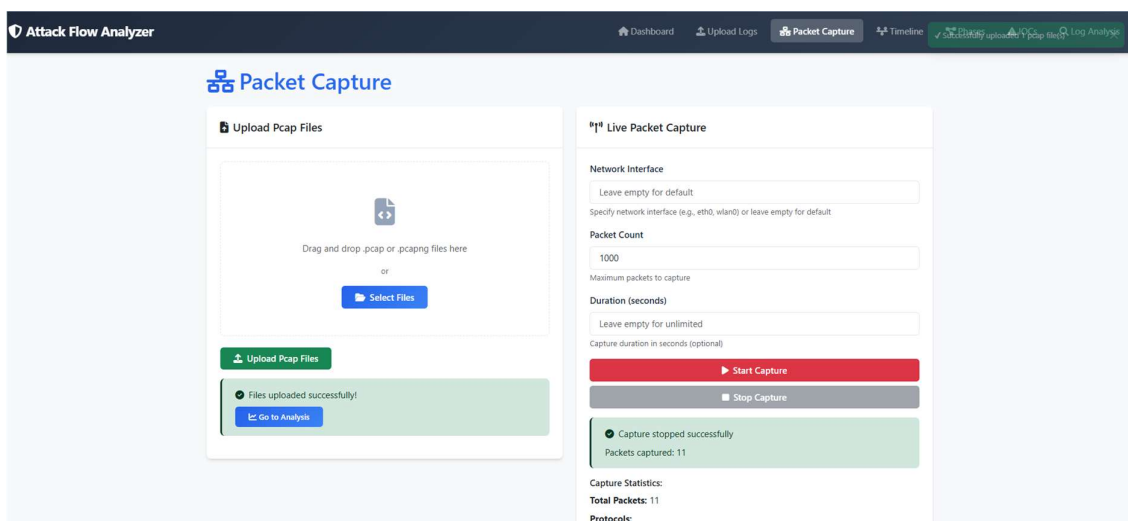
## Results

The system successfully processes multiple log formats and correlates events across different sources, enabling accurate classification of attack phases. Interactive timelines provide a clear visualization of attack progression, while Indicators of Compromise (IOCs) are automatically extracted with metadata and can be exported in JSON or CSV formats. Related event groups are displayed for better context, and real-time packet analysis allows monitoring live network activity. Overall, the tool effectively reconstructs end-to-end attack flows, making incident response faster and more efficient.





With the Upload logs: The user to upload multiple log files from different sources (Apache/Nginx, authentication, firewall logs, and packet captures). The system automatically detects the log type and normalizes timestamps for consistent analysis.



**Timeline Tab:** Displays an interactive chronological view of all events, Shows how attacks progress over time and allows users to zoom into specific events or periods, Helps analysts visualize the sequence of attack phases.

**IOC Tab:** Automatically extracts Indicators of Compromise such as IP addresses, domains, URLs, file hashes, and user agents, Provides export functionality in JSON or CSV format for further analysis or reporting, Ensures analysts can quickly access actionable threat data.

**Packet Capture Tab:** Captures network packets in real-time using Scapy, Allows users to analyze live traffic or pre-captured packet files (.pcap/.pcapng), Displays protocol details for TCP, UDP, ICMP, DNS, and HTTP packets, supporting deeper network analysis.

Attack Flow Analyzer

Dashboard Upload Logs Packet Capture Timeline Phases IOCs Log Analysis

Protocols:  
2: 3  
tcp: 7  
udp: 1

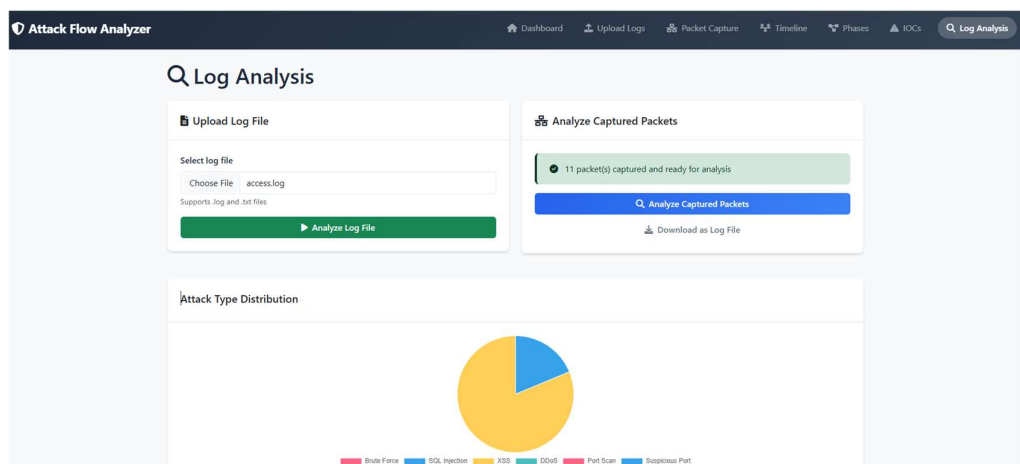
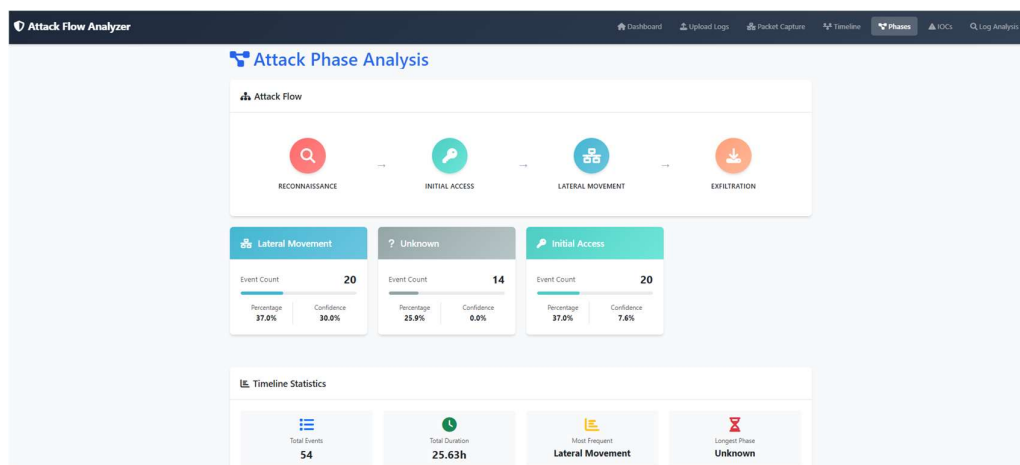
Live Packet Display

17 packets Auto-scroll Clear

Filter by protocol... Filter by source IP... Filter by dest IP... Search info...

NO.	TIME	SOURCE	DESTINATION	PROTOCOL	LENGTH	INFO
1	2015-51-186	192.168.8.110:2455	183.82.243.66:53	TCP	54	-
2	2015-51-201	192.168.8.110:12662	183.82.243.66:53	TCP	54	-
3	2015-51-217	192.168.8.110:4912	183.82.243.66:53	TCP	54	-
4	2015-51-305	192.168.8.126	229.8.8.22	2	58	-
5	2015-51-569	192.168.8.110:4812	183.82.243.66:53	TCP	96	-
6	2015-52-415	192.168.8.183:50612	239.255.255.250:1998	UDP	167	-
7	2015-52-567	192.168.8.100	229.8.8.22	2	64	-
8	2015-52-612	192.168.8.110:9386	183.82.243.66:53	TCP	96	-
9	2015-52-612	192.168.8.110:9396	183.82.243.66:53	TCP	96	-
10	2015-52-612	192.168.8.110:1896	183.82.243.66:53	TCP	96	-

Attack Phases Tab: Classifies events into MITRE ATT&CK phases (Reconnaissance, Initial Access, Lateral Movement, Exfiltration), Displays phase distribution charts and lists of events under each phase with confidence scores, Assists analysts in understanding which parts of the attack lifecycle were executed.



Attack Flow Analyzer																																							
<div> Dashboard Upload Logs Packet Capture Timeline Phases IOCs Log Analysis </div>																																							
<div> Detected Attack Types </div> <div> <div>SQL Injection3</div> <div>XSS13</div> </div>																																							
<div> Attack Details </div> <table> <tr> <th>Attack Type</th><th>Occurrences</th><th>Severity</th><th>Last Detected</th><th>Suggestion</th></tr> <tr> <td>Brute Force</td><td>0</td><td>Low</td><td>-</td><td>Use strong passwords &amp; rate-limiting</td></tr> <tr> <td>SQL Injection</td><td>3</td><td>High</td><td>1/13/2026, 8:17:50 PM</td><td>Validate &amp; sanitize inputs</td></tr> <tr> <td>XSS</td><td>13</td><td>High</td><td>1/13/2026, 8:17:50 PM</td><td>Escape user inputs on web pages</td></tr> <tr> <td>DDoS</td><td>0</td><td>Low</td><td>-</td><td>Use firewall and rate-limiting</td></tr> <tr> <td>Port Scan</td><td>0</td><td>Medium</td><td>-</td><td>Monitor for multiple connection attempts to different ports</td></tr> <tr> <td>Suspicious Port</td><td>0</td><td>Medium</td><td>-</td><td>Review connections to non-standard ports</td></tr> </table>					Attack Type	Occurrences	Severity	Last Detected	Suggestion	Brute Force	0	Low	-	Use strong passwords & rate-limiting	SQL Injection	3	High	1/13/2026, 8:17:50 PM	Validate & sanitize inputs	XSS	13	High	1/13/2026, 8:17:50 PM	Escape user inputs on web pages	DDoS	0	Low	-	Use firewall and rate-limiting	Port Scan	0	Medium	-	Monitor for multiple connection attempts to different ports	Suspicious Port	0	Medium	-	Review connections to non-standard ports
Attack Type	Occurrences	Severity	Last Detected	Suggestion																																			
Brute Force	0	Low	-	Use strong passwords & rate-limiting																																			
SQL Injection	3	High	1/13/2026, 8:17:50 PM	Validate & sanitize inputs																																			
XSS	13	High	1/13/2026, 8:17:50 PM	Escape user inputs on web pages																																			
DDoS	0	Low	-	Use firewall and rate-limiting																																			
Port Scan	0	Medium	-	Monitor for multiple connection attempts to different ports																																			
Suspicious Port	0	Medium	-	Review connections to non-standard ports																																			

## Limitations

- Supports only common log formats (Apache/Nginx, syslog, generic firewall); other formats need custom parsers.
- Rule-based classification may miss novel or unknown attack patterns.
- Limited to text-based logs and standard packet formats (.pcap, .pcapng).
- IOC extraction depends on log/packet content; encoded or obfuscated IOCs may be missed.
- Live packet capture requires administrator/root privileges.
- Single-threaded log processing may be slow for very large files.

## Future Enhancements

- Add machine learning-based phase classification for improved accuracy.
- Support additional log formats like Windows Event Logs and proprietary systems.
- Enable real-time log streaming from remote sources.
- Integrate with SIEM systems (Splunk, ELK stack).
- Implement STIX/TAXII export for threat intelligence sharing.
- Add user authentication and role-based multi-user support.
- Include advanced packet analysis and network flow visualization.
- Automate IOC enrichment using threat intelligence feeds.

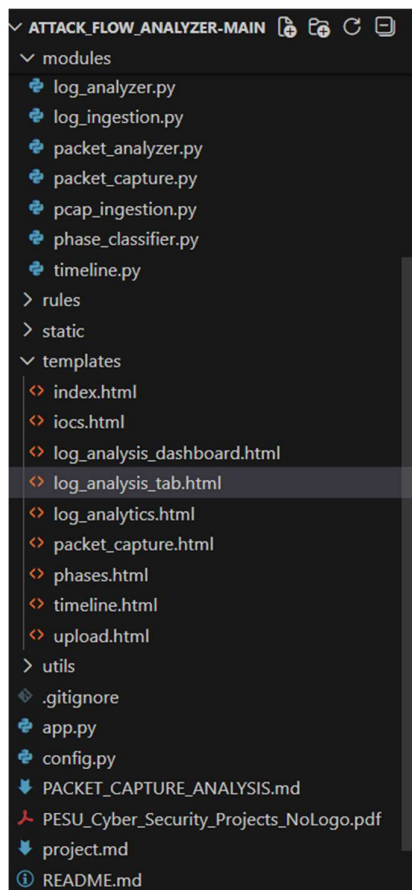
## Conclusion

The Incident Response Attack-Flow Analyzer successfully automates log analysis, attack phase classification, and IOC extraction, providing security analysts with a comprehensive view of attack lifecycles. The web-based dashboard with interactive timelines and real-time packet capture enhances incident response efficiency and accessibility. This project demonstrated practical skills in log parsing, event correlation, network packet analysis, and web application development, forming a solid foundation for future improvements such as machine learning-based classification and support for additional log formats.

## References

1. MITRE ATT&CK Framework – <https://attack.mitre.org/>
2. Flask Documentation – <https://flask.palletsprojects.com/>
3. Scapy Documentation – <https://scapy.readthedocs.io/>
4. PyShark Documentation – <https://kiminewt.github.io/pyshark/>
5. Plotly Python Documentation – <https://plotly.com/python/>
6. Apache/Nginx Common Log Format – <https://httpd.apache.org/docs/current/logs.html>
7. Python Regular Expressions – <https://docs.python.org/3/library/re.html>
8. PES University Cyber Security Project Compendium

## Folder Structure



- **code/** – Contains main application, modules, templates, static files, utilities, and rules.
- **Dataset or logs/** – Sample logs for testing and demonstration.
- **report/** – Project report in PDF format.