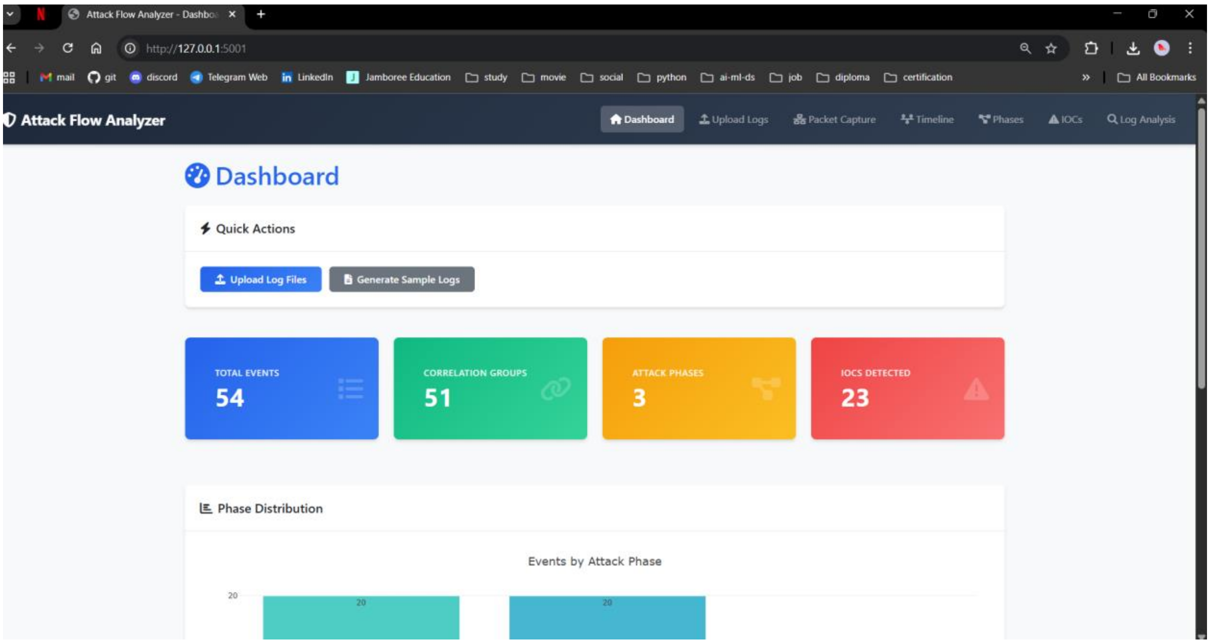
PES UNIVERSITY
Department of Computer Science and Engineering
M.Tech 1st SEM 2025

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



Dashboard

Quick Actions

Upload Log Files | Generate Sample Logs

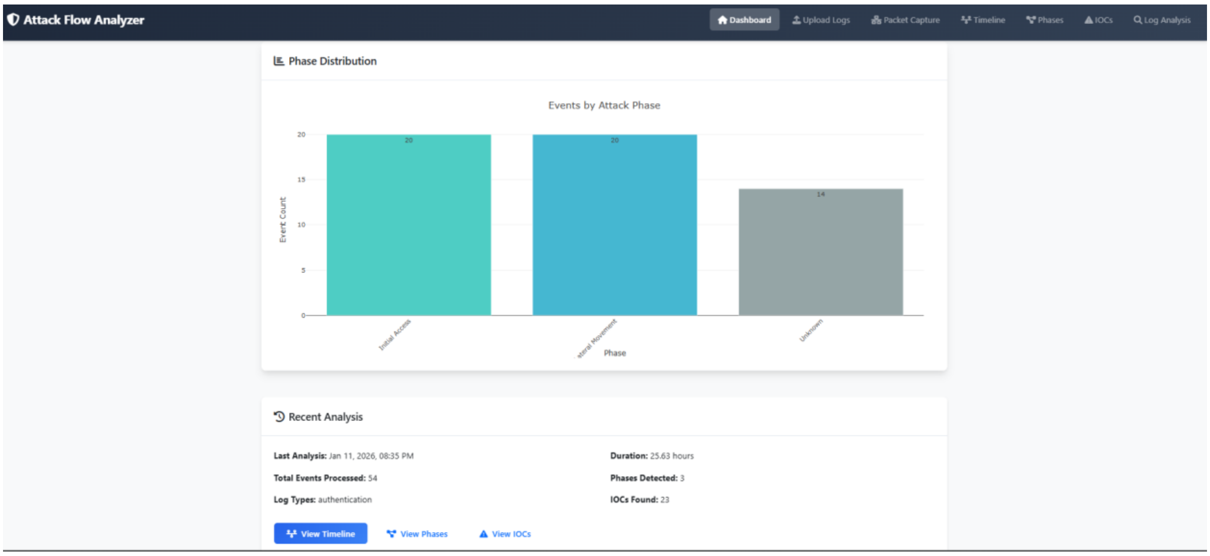
TOTAL EVENTS 54

CORRELATION GROUPS 51

ATTACK PHASES 3

IOCS DETECTED 23

Phase Distribution



Recent Analysis

Last Analysis: Jan 11, 2026, 08:35 PM

Total Events Processed: 54

Log Types: authentication

Duration: 25.63 hours

Phases Detected: 3

IOCs Found: 23

View Timeline | View Phases | View IOCs

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

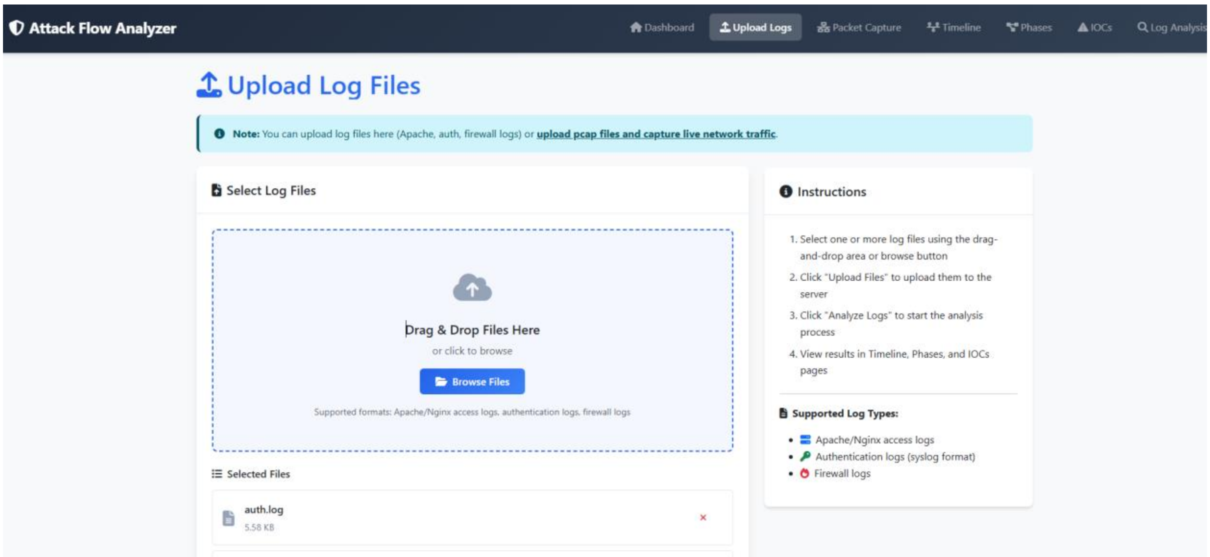
- Small to Medium Enterprises (SMEs): Automate incident response without expensive SIEM tools.
- Security Operations Centers (SOCs): Quickly analyze logs, detect attack phases, and extract IOCs.
- Network Monitoring: Track real-time attacks using packet capture and timeline visualization.
- Incident 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]



Upload Log Files

Note: You can upload log files here (Apache, auth, firewall logs) or upload pcap files and capture live network traffic.

Select Log Files

Drag & Drop Files Here

or click to browse

Supported formats: Apache/Nginx access logs, authentication logs, firewall logs

Selected Files

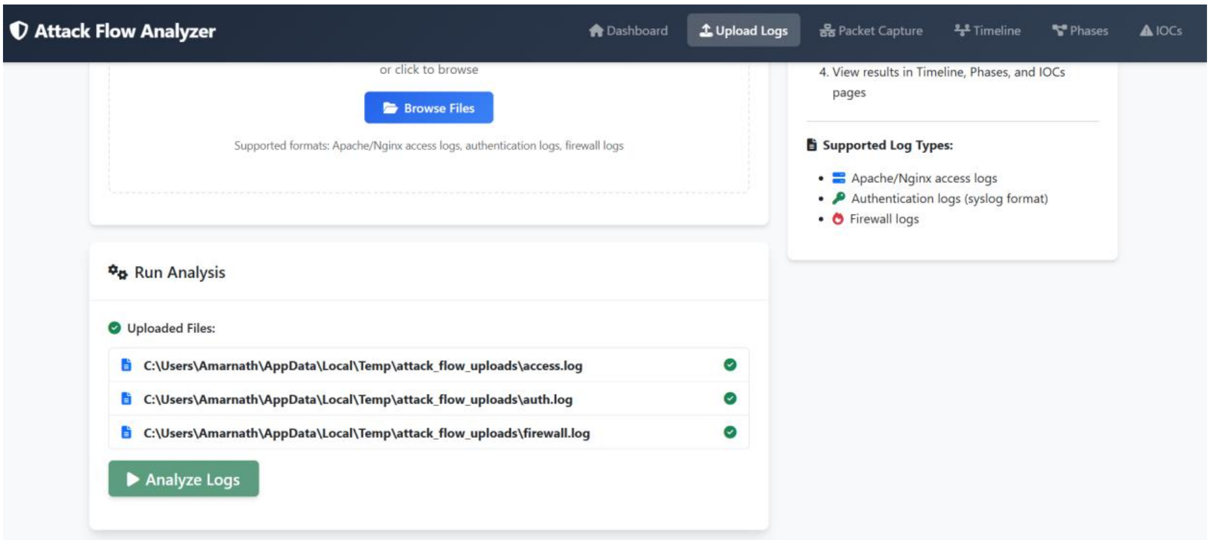
auth.log
5.58 KB

Instructions

1. Select one or more log files using the drag-and-drop area or browse button
2. Click "Upload Files" to upload them to the server
3. Click "Analyze Logs" to start the analysis process
4. View results in Timeline, Phases, and IOCs pages

Supported Log Types:

- Apache/Nginx access logs
- Authentication logs (syslog format)
- Firewall logs



or click to browse

Supported formats: Apache/Nginx access logs, authentication logs, firewall logs

4. View results in Timeline, Phases, and IOCs pages

Supported Log Types:

- Apache/Nginx access logs
- Authentication logs (syslog format)
- Firewall logs

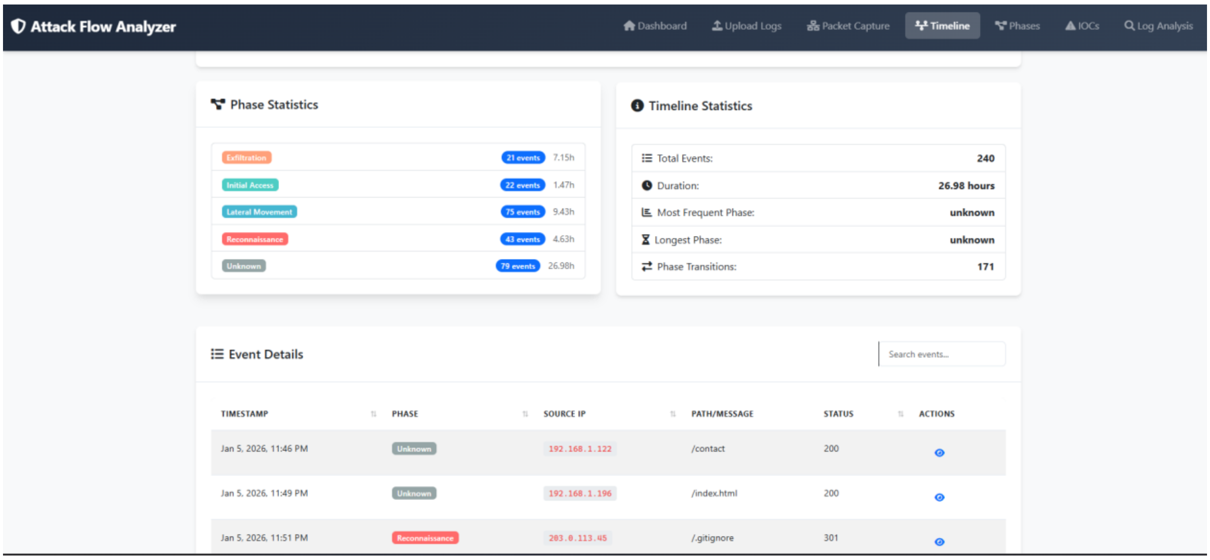
Run Analysis

Uploaded Files:

```
C:\Users\Amarnath\AppData\Local\Temp\attack_flow_uploads\access.log ✓
C:\Users\Amarnath\AppData\Local\Temp\attack_flow_uploads\auth.log ✓
C:\Users\Amarnath\AppData\Local\Temp\attack_flow_uploads\firewall.log ✓
```

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]



Phase Statistics

Exfiltration	21 events	7.15h
Initial Access	22 events	1.47h
Lateral Movement	75 events	9.43h
Reconnaissance	43 events	4.63h
Unknown	79 events	26.98h

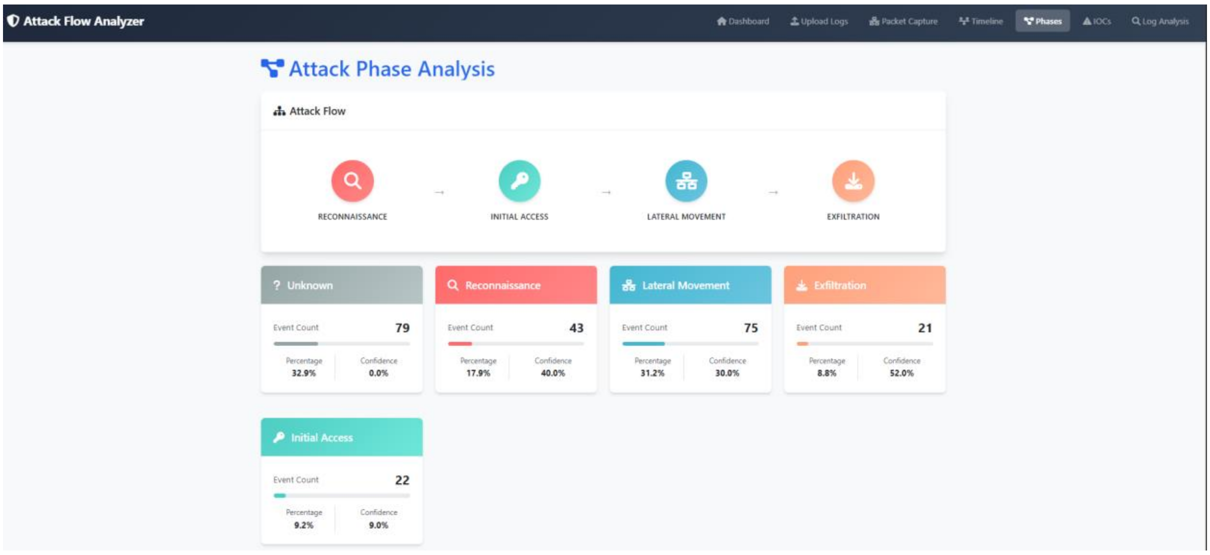
Timeline Statistics

Total Events:	240
Duration:	26.98 hours
Most Frequent Phase:	unknown
Longest Phase:	unknown
Phase Transitions:	171

Event Details

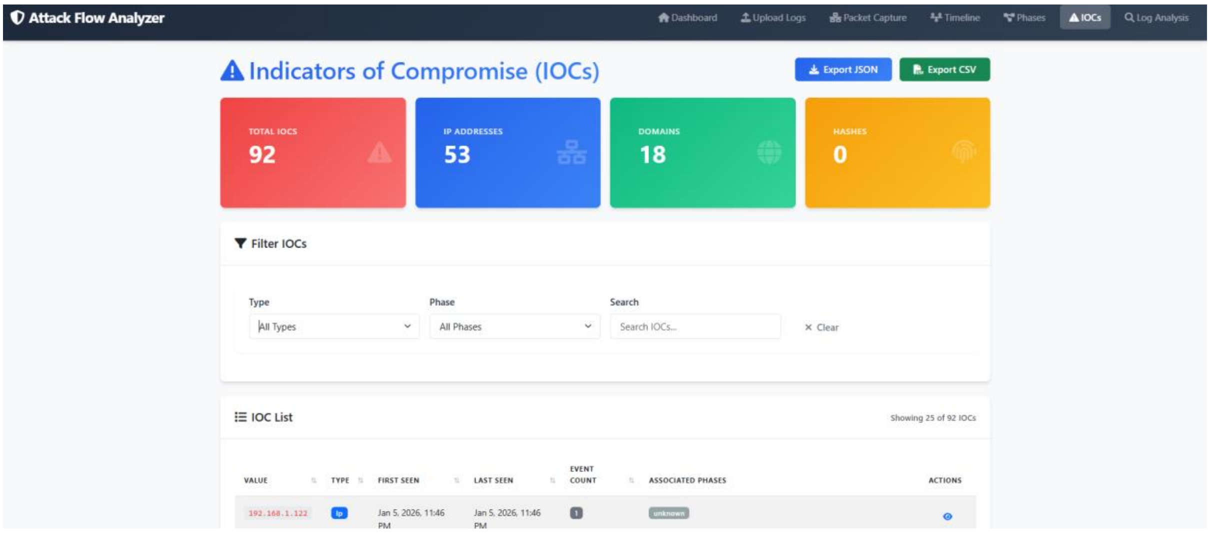
Search events...

TIMESTAMP	PHASE	SOURCE IP	PATH/MESSAGE	STATUS	ACTIONS
Jan 5, 2026, 11:46 PM	Unknown	192.168.1.122	/contact	200	
Jan 5, 2026, 11:49 PM	Unknown	192.168.1.196	/index.html	200	
Jan 5, 2026, 11:51 PM	Reconnaissance	203.0.113.45	/.gitignore	301	



Phase	Event Count	Percentage	Confidence
Unknown	79	32.9%	0.0%
Reconnaissance	43	17.9%	40.0%
Lateral Movement	75	31.2%	30.0%
Exfiltration	21	8.8%	52.0%
Initial Access	22	9.2%	9.0%

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]



Indicators of Compromise (IOCs)

TOTAL IOCS 92

IP ADDRESSES 53

DOMAINS 18

HASHES 0

Filter IOCs

Type: All Types

Phase: All Phases

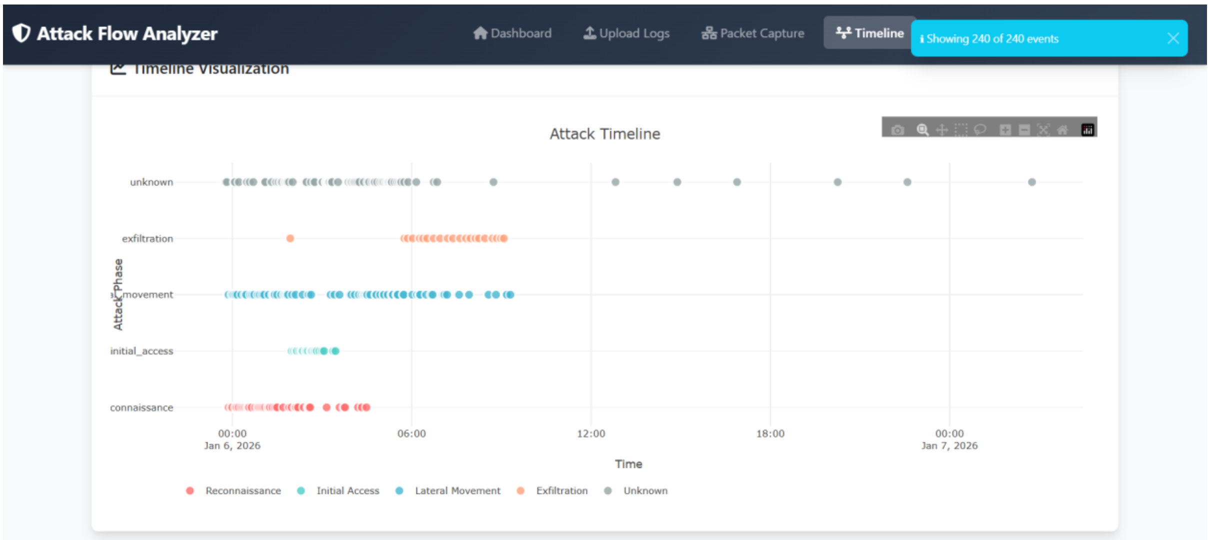
Search: Search IOCs...

× Clear

IOC List

Showing 25 of 92 IOCs

VALUE	TYPE	FIRST SEEN	LAST SEEN	EVENT COUNT	ASSOCIATED PHASES	ACTIONS
192.168.1.122	ip	Jan 5, 2026, 11:46 PM	Jan 5, 2026, 11:46 PM	1	unknown	



ℹ Showing 240 of 240 events

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.

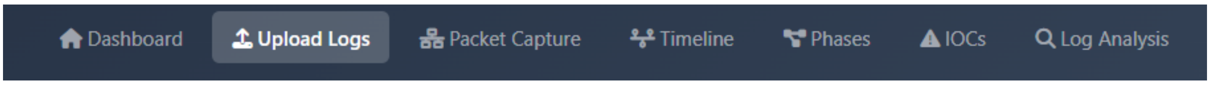
Implementation Details:

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.

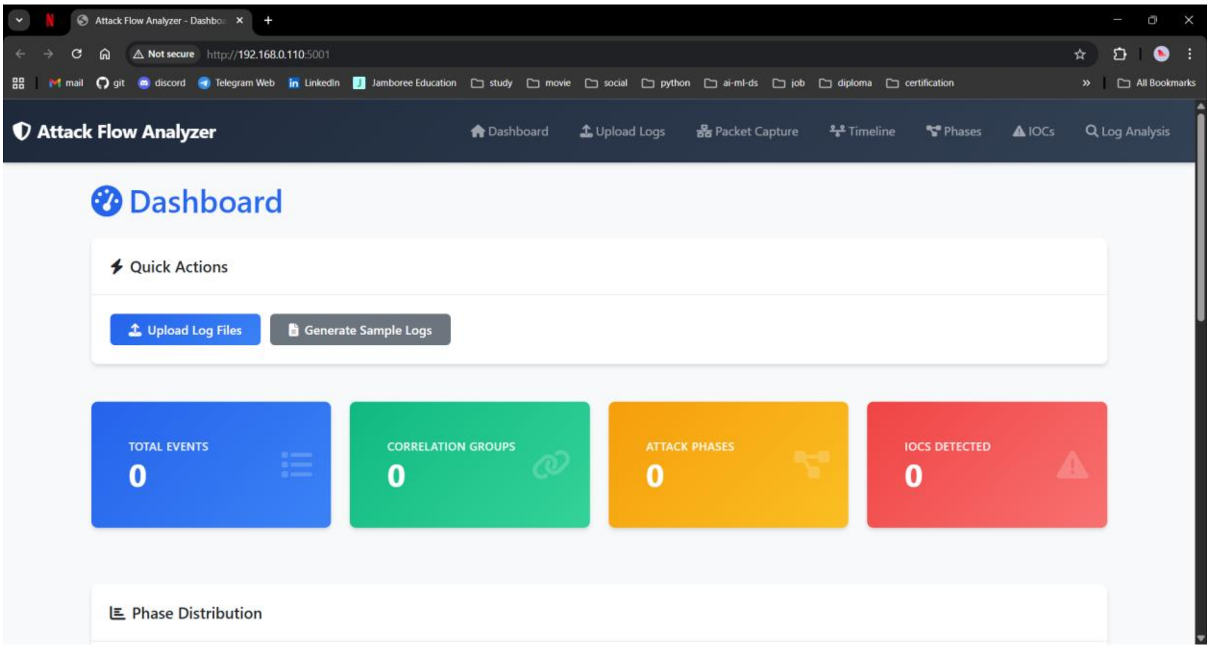


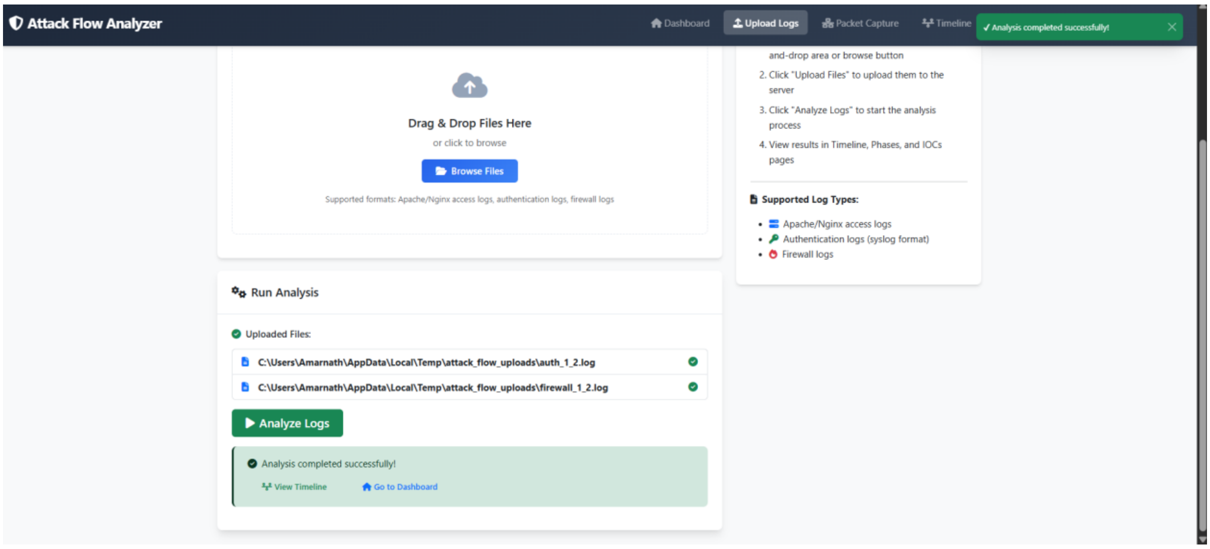
Workflow

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.





✓ Analysis completed successfully!

Drag & Drop Files Here

or click to browse

Supported formats: Apache/Nginx access logs, authentication logs, firewall logs

and-drop area or browse button

2. Click "Upload Files" to upload them to the server
3. Click "Analyze Logs" to start the analysis process
4. View results in Timeline, Phases, and IOCs pages

Supported Log Types:

- Apache/Nginx access logs
- Authentication logs (syslog format)
- Firewall logs

Run Analysis

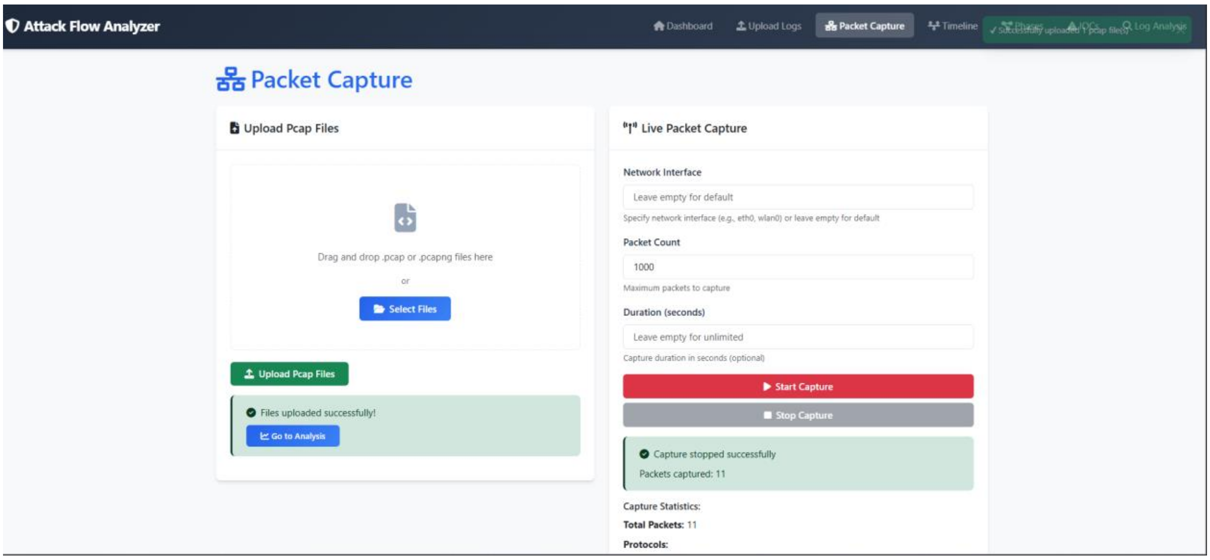
Uploaded Files:

C:\Users\Amarnath\AppData\Local\Temp\attack_flow_uploads\auth_1_2.log ✓

C:\Users\Amarnath\AppData\Local\Temp\attack_flow_uploads\firewall_1_2.log ✓

Analysis completed successfully!

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.



✓ Successfully uploaded 1 pcap file(s)

Packet Capture

Upload Pcap Files

Drag and drop .pcap or .pcapng files here

or

Files uploaded successfully!

Live Packet Capture

Network Interface: Leave empty for default

Specify network interface (e.g., eth0, wlan0) or leave empty for default

Packet Count: 1000

Maximum packets to capture

Duration (seconds): Leave empty for unlimited

Capture duration in seconds (optional)

Capture stopped successfully

Packets captured: 11

Capture Statistics:

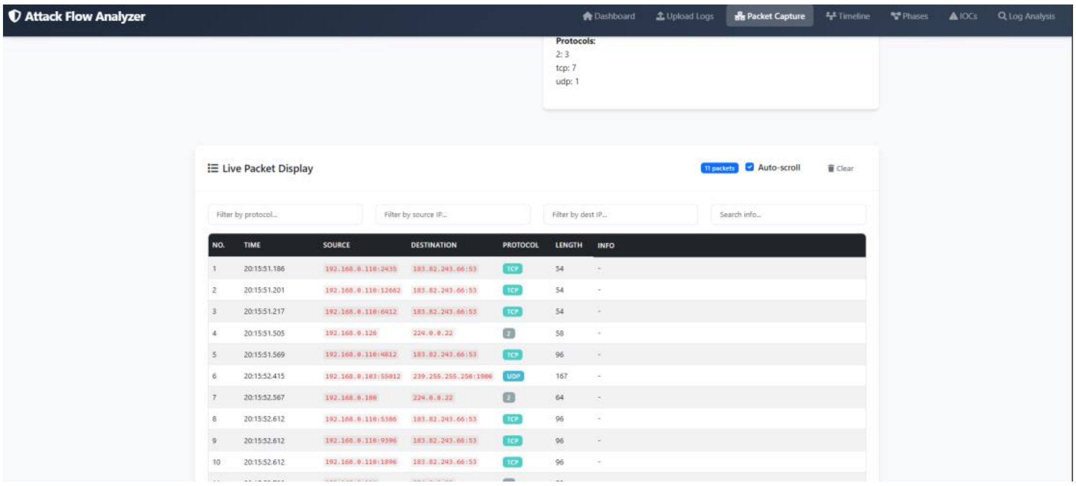
Total Packets: 11

Protocols:

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.



Protocols:
2: 3
tcp: 7
udp: 1

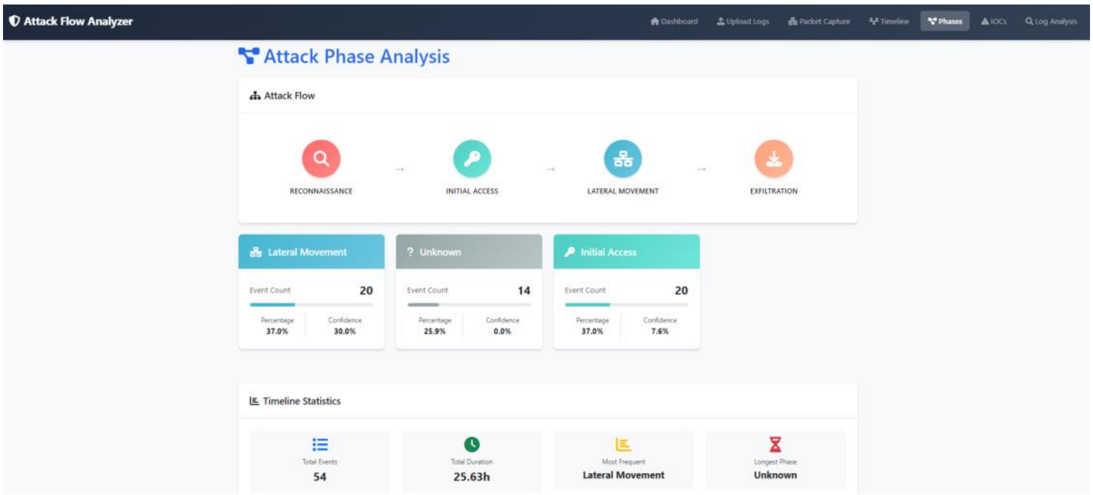
Live Packet Display

11 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	20:15:51.186	192.168.0.110:2435	183.82.243.66:53	TCP	54	-
2	20:15:51.201	192.168.0.110:12662	183.82.243.66:53	TCP	54	-
3	20:15:51.217	192.168.0.110:6412	183.82.243.66:53	TCP	54	-
4	20:15:51.505	192.168.0.126	224.0.0.22	2	58	-
5	20:15:51.569	192.168.0.110:4812	183.82.243.66:53	TCP	96	-
6	20:15:52.415	192.168.0.103:55012	239.255.255.250:1900	UDP	167	-
7	20:15:52.567	192.168.0.108	224.0.0.22	2	64	-
8	20:15:52.612	192.168.0.110:5386	183.82.243.66:53	TCP	96	-
9	20:15:52.612	192.168.0.110:9396	183.82.243.66:53	TCP	96	-
10	20:15:52.612	192.168.0.110:1896	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.



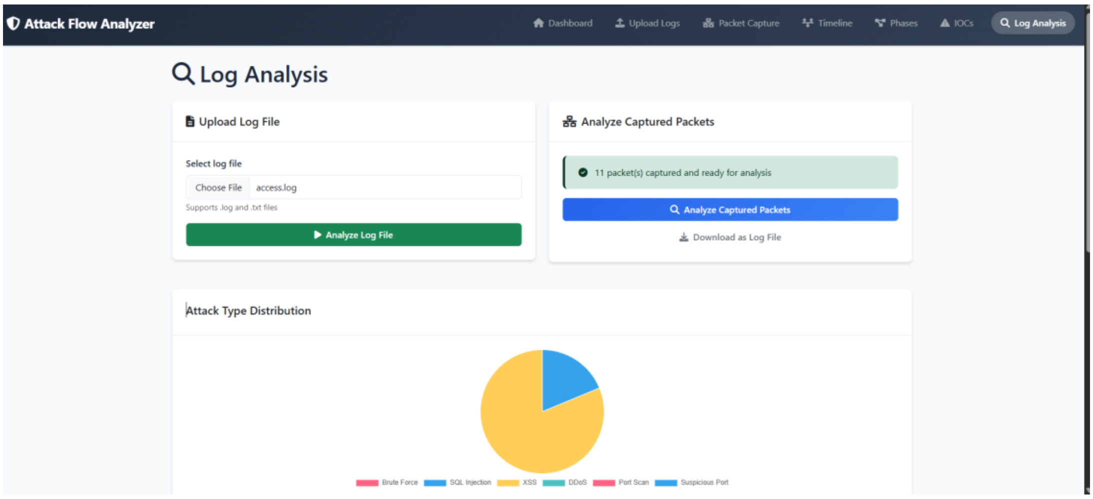
Attack Phase Analysis

Attack Flow

Lateral Movement	Unknown	Initial Access
Event Count 20	Event Count 14	Event Count 20
Percentage 37.0%	Percentage 25.9%	Percentage 37.0%
Confidence 30.0%	Confidence 0.0%	Confidence 7.6%

Timeline Statistics

Total Events	Total Duration	Most Frequent	Longest Phase
54	25.63h	Lateral Movement	Unknown



Log Analysis

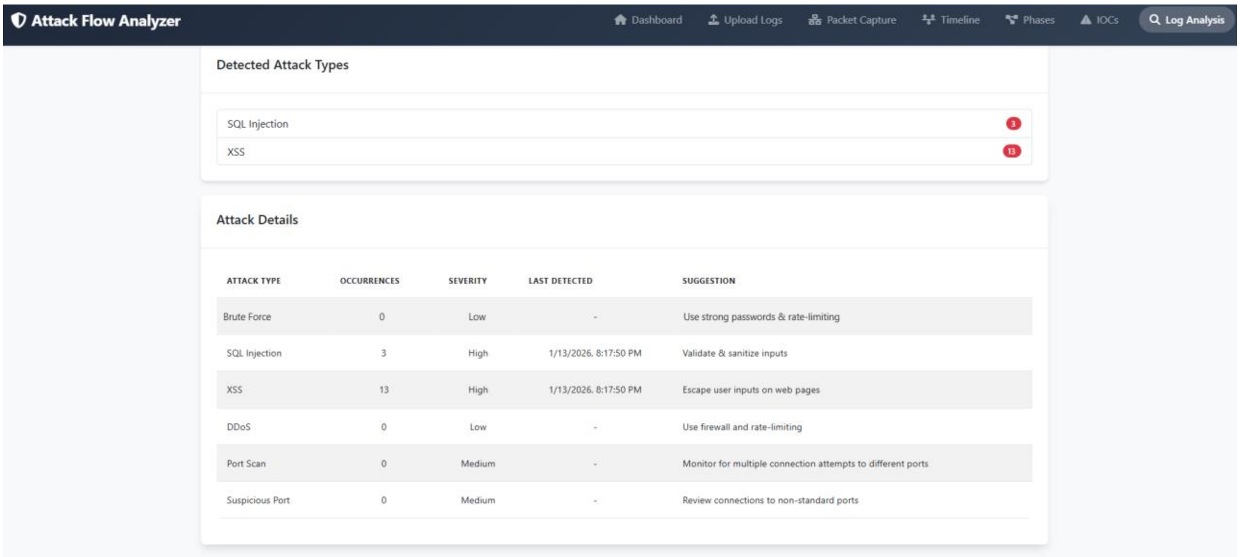
Upload Log File

Select log file
Choose File access.log
Supports .log and .txt files
Analyze Log File

Analyze Captured Packets

11 packet(s) captured and ready for analysis
Analyze Captured Packets
Download as Log File

Attack Type Distribution



Detected Attack Types

SQL Injection	3
XSS	13

Attack Details

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.

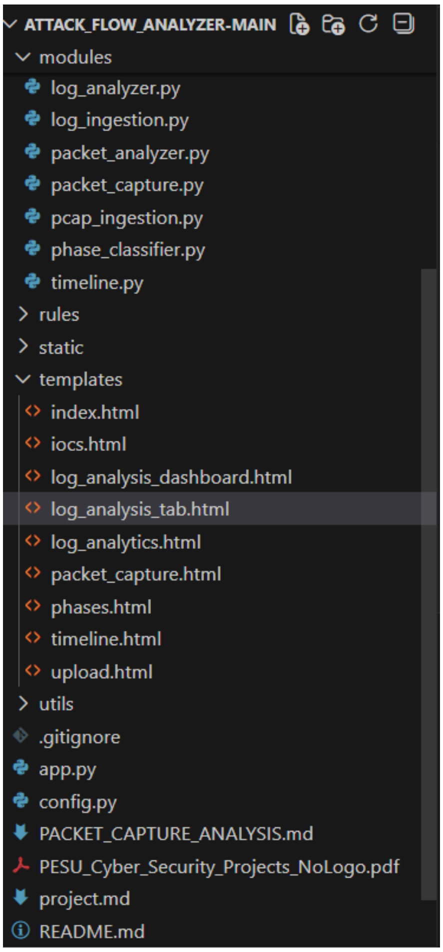
Conclusion

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



```
∨ ATTACK_FLOW_ANALYZER-MAIN
  ∨ modules
     log_analyzer.py
     log_ingestion.py
     packet_analyzer.py
     packet_capture.py
     pcap_ingestion.py
     phase_classifier.py
     timeline.py
  > rules
  > static
  ∨ templates
     index.html
     iocs.html
     log_analysis_dashboard.html
     log_analysis_tab.html
     log_analytics.html
     packet_capture.html
     phases.html
     timeline.html
     upload.html
  > utils
  .gitignore
  app.py
  config.py
  PACKET_CAPTURE_ANALYSIS.md
  PESU_Cyber_Security_Projects_NoLogo.pdf
  project.md
  README.md
```

- **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.