**Project Design Documentation**

Last updated: June 13th, 2025

**1. Project Title & Version Control**

Project Title: Automated SIEM Correlation and Attacker Timeline Reconstruction

Version: DRAFT

Date: 08/06/2025

Change Log: Initial draft of project scope, goals, and implementation plan. To be updated weekly with milestone progress.

This project aims to develop an intelligent SIEM-based system capable of automatically correlating log events, identifying Indicators of Compromise (IOCs), and generating structured attacker timelines. The project builds on a simulated virtual SOC lab and enhances detection, analysis, and reporting capabilities without manual log inspection.

**2. Project Summary (2–3 sentences)**

This project aims to develop an intelligent SIEM-based system capable of automatically correlating log events, identifying Indicators of Compromise (IOCs), and generating structured attacker timelines. The project builds on a simulated virtual SOC lab and enhances detection, analysis, and reporting capabilities without manual log inspection.

# 3. Problem Statement / Use Case

Security analysts often spend significant time manually parsing logs to understand the sequence of an attack. This project addresses the need for automated correlation and timeline generation by detecting chained events and extracting meaningful insights. The system will be used in a virtual lab but reflects real-world SOC workflows in small-to-medium enterprises lacking advanced threat-hunting tools

# 4. Goals and Objectives

 Build a modular system to correlate multi-stage attack events using log data.

 Automatically extract IOCs and generate human-readable attack timelines.

 Integrate automated alerting or response for matched IOC chains or behavioral sequences.

# 5. Key Features / Functions

 Centralized log collection from Windows (Sysmon) and Linux (auditd).

 Rule-based or script-based event correlation using user, IP, and time windows.

 Automated timeline generator outputting Markdown/CSV/JSON formats.

 IOC parsing from logs and enrichment capability.

 Optional: Auto-response actions like IP blocking or alerting

# 6. Tech Stack and Tools

 **Languages:** Python, Bash (support scripts)

 **SIEM:** Wazuh + Elasticsearch + Kibana (or Timesketch optional)

 **Data Collection:** Sysmon (Windows), auditd (Linux), Filebeat

 **Threat Frameworks:** MITRE ATT&CK, Sigma (optional for rules)

 **Automation:** Python subprocess, Wazuh active response

 **Documentation:** Markdown, GitHub, PDF reports

# 7. Architecture / Workflow Diagram

**Algorithm Summary:**

1. Logs are collected via Filebeat/Wazuh.
2. Python scripts parse logs for IOCs and group related events.
3. Events are correlated based on attacker behavior patterns (IP, user, time proximity).
4. Output is converted into structured timeline format.
5. IOC detection can trigger auto-response scripts.

# 8. Timeline / Weekly Milestones

|  |  |
| --- | --- |
| Week | Outcome |
| **Week 1** (Aug 25–28) | Define project scope, finalize attack scenarios, build lab architecture diagram |
| **Week 2** (Sept 1–4) | Enable logging, normalize log data, build first version of IOC extractor |
| **Week 3** (Sept 8–11) | Simulate attacks, manually tag logs, create initial attacker timelines |
| **Week 4** (Sept 15–18) | Build and test correlation logic (Python or Wazuh rules) |
| **Week 5** (Sept 22–25) | Develop automated timeline generator, optionally visualize in Kibana |
| **Week 6** (Sept 29–Oct 2) | Implement basic auto-response + compile project documentation & report |

# 9. Risks and Risk Mitigation

| **Risk** | **Mitigation Strategy** |
| --- | --- |
| Limited VM resources for full simulation | Focus on endpoint-to-SIEM correlation instead of lateral movement |
| Complex logs difficult to parse | Start with simple attack chains and build parsing logic iteratively |
| Script or rules produce false positives | Use manual validation in early stages; define time-window constraints properly |
| Tool or integration issues (Timesketch, etc.) | Keep core deliverables independent of optional tools |

.

# 10. Evaluation Criteria

 Timeline generation outputs attacker actions in the correct order

 At least 2 multi-stage attack scenarios are automatically correlated

 IOC detection triggers a scripted response (e.g., alert or block)

 Documentation of attack chain includes TTP alignment with MITRE ATT&CK

# 11. Future Considerations

 Expand to handle lateral movement and insider threat scenarios.

 Integrate OpenCTI or MISP for external threat enrichment.

 Build a web-based frontend for browsing timelines and detection insights.

 Scale to handle real-time event streaming and high log volume.