Shri Ramswaroop Memorial University,

Barabanki, Lucknow

Case Study: End-to-End SIEM Pipeline—Detecting and Responding to Attacks Using Splunk and ELK Stack

B.Tech (Computer Science, Cybersecurity Specialization),

Department of Computer Science

Submitted to:

Santanu Sasmal

Submitted by:

Nidhi Rai

Roll no. 202210101180019

Date

11/17/2025

Project Focus

Cybersecurity, Security Operations (SOC), Data Analytics, Log Management

Keywords

SIEM, Splunk, ELK Stack, Threat Detection, Correlation, Logstash, SPL, SOAR

Abstract

A **Security Information and Event Management (SIEM) system** is a critical cornerstone for bolstering digital resilience and achieving **unified security** in modern enterprises. This project details the design, implementation, and comparison of an **End-to-End SIEM Pipeline** using the proprietary **Splunk Enterprise Security** and the open-source **ELK Stack (Elasticsearch, Logstash, Kibana)**. The goal was to establish a scalable log analysis platform capable of overcoming challenges related to **large volumes of logs** and the **difficulty in correlating events across diverse systems**.

A hands-on lab environment was established to simulate real-world cyberattacks, including **Nmap scans, brute-force attempts, SQL injections, and Metasploit exploitation**, replicating an SOC analyst workflow. The methodology focused on architecting a complete log flow, covering collection (via **Splunk Forwarders/Beats**), processing (via **Logstash**), indexing (via **Elasticsearch/Splunk Indexer**), and visualization (via **Kibana/Splunk Dashboards**).

Custom detection logic, utilizing **Search Processing Language (SPL)** and specialized filtering, successfully identified all simulated threats, providing **real-time event monitoring**. Comparative analysis reveals that **Splunk Enterprise Security** has a higher Analyst Rating (**93**) and excels in **UEBA** and user-friendly dashboards, while the **ELK Stack** (Analyst Rating **82**) is praised by users (90% 'excellent' sentiment) for its **scalability** and **cost-effectiveness** due to its open-source core. The project concludes that both platforms enable proactive defense, with future enhancements recommending **SOAR integration** for automated response.

INDEX

1. Introduction

1.1 Context and Background

A SIEM system aggregates and analyzes event data from various sources to identify security threats. SIEM platforms have become indispensable, pulling in log data from across environments, surfacing threats, and helping teams respond quickly. This capability is critical because IT environments generate **extensive logfiles** that record the minutiae of daily operations, and without a centralized solution, this volume of data quickly becomes unwieldy.

Splunk Enterprise Security is a robust SIEM solution, tailored for large enterprises, and praised for features like real-time event monitoring, threat intelligence integration, and comprehensive security capabilities. Splunk uses a proprietary search language called **Search Processing Language (SPL)** for traversing and executing contextual queries against large data sets.

The **ELK Stack** (Elasticsearch, Logstash, Kibana) is a consolidated log analysis and visualization platform. **Elasticsearch** is the distributed search/analytics engine used for indexing, **Logstash** is the data processing pipeline, and **Kibana** is the data visualization interface. More recently, **Beats** was added to the stack as a lightweight method to collect data from various sources.

1.2 Problem Statement

Effective cybersecurity is challenged by the lack of **real-time detection of attacks**. Organizations struggle due to:

• **Large Volumes of Logs:** Logs are generated from diverse sources (servers, firewalls, applications), making the data difficult to manage.

• **Correlation Difficulty:** It is challenging to correlate events across systems to identify multi-stage attack campaigns.

• **Delayed Visibility:** Security compromise often progresses until damage occurs because of a lack of visibility into suspicious activity.

The project addresses the need for a scalable pipeline that can efficiently normalize logs and rapidly detect common attack patterns, strengthening the overall security posture.

2. Methodology and Technical Implementation

2.1 SIEM Pipeline Architecture (Conceptual Flow)

The end-to-end SIEM pipeline requires event data to move through a structured workflow. This architecture is designed to replicate a **real-world SOC analyst workflow**.

Stage	Function	Splunk Component	ELK Stack Component
Log Generation	Creating security events (e.g., failed logins, scans).	Victim/Target Machine.	Linux VMs (Kali, Ubuntu) for attacker/server.
Collection	Shipping logs from endpoints to the processing stage.	**Splunk Forwarder**.	**Beats** (e.g., Filebeat for logs).
Processing	Filtering, transforming, and enriching (e.g., with threat intelligence).	`props.conf/transforms.conf`	**Logstash** (the data processing pipeline).
Indexing	Storing and organizing processed data for fast retrieval.	**Splunk Indexes**.	**Elasticsearch** (the distributed search engine).

Visualization	Providing graphical views for analysis.	**Splunk Dashboards**.	**Kibana Dashboards**.
Detection	Applying correlation rules to trigger alerts.	**SPL Queries**.	**ELK Queries/Rules** (e.g., via ElastAlert).

2.2 Lab Setup and Simulated Attacks

The project used a virtualized lab setup. The **ELK Stack** was hosted on an **Ubuntu Server VM**, and an **attacker VM (Kali Linux)** was used for running offensive tools like **Nmap, Metasploit, and John the Ripper**. **Filebeat** was installed on the target machine to forward logs to **Logstash**.
The project configured custom alerts to detect simulated attacks, which included:

1. **Nmap scans** and **excessive port access attempts.**
2. **Brute-Force Attacks** (John the Ripper) and **Repeated Failed Logins**.
3. **SQL injection attempts**.
4. **Metasploit Exploitation** (reverse shell attempts and privilege escalation).

3. Technical Implementation and Code Examples

3.1 Splunk Detection Queries (Search Processing Language—SPL)

Splunk searches are crucial for proactive defense and detecting suspicious behavior.

A. Brute Force Login Detection (Use Case 2)

This query searches secure logs (`sourcetype=secure*`) for failed authentication events and aggregates them by user and source IP:

```
index=* sourcetype=secure*
| search "failed password" OR "authentication failure"
| rex "(?<user>[\w.-]+)"
| rex "(?<src_ip> ([3, 11, 26, 38-43]{1,3}\.){3}[3, 11, 26, 38-43]{1,3})"
| stats count as failed_attempts by user, src_ip
| where failed_attempts > 5
```

• **Detection Logic:** Filters for users or IPs that show **more than five failed login attempts**, which indicates a possible brute force attack.
• **Response:** Block the attacking IP address or enable MFA for targeted accounts.

B. Port Scanning Detection (Use Case 3)

This query identifies source IPs that attempt to access multiple, different ports, indicating reconnaissance activity:

```
index=* sourcetype=secure-2
| search "Failed password"
| rex "from (?<src_ip>\d+\.\d+\.\d+\.\d+) port (?<dest_port>\d+)"
| stats count by src_ip, dest_port
| stats count as port_attempts by src_ip
| where port_attempts > 5
| sort - port_attempts
```

• **Detection Logic:** Filters for source IPs that have attempted more than five different port accesses, then sorts by attempt count.

• **Response:** Immediate action includes blocking the identified IP addresses and implementing IDS/IPS.

3.2 ELK Implementation Example: Logstash and Data Processing

Logstash is responsible for collecting, transforming, and enriching data before indexing.

A. Logstash Configuration (Beats Input/Elasticsearch Output)

A basic `logstash.conf` file defines input from Beats on port 5044 and outputs to Elasticsearch, indexing to `threat intelligence:`

```
input {
    beats { port: 5044 }
}
output {
    elasticsearch {
        hosts => ["localhost:9200"]
        index => "threat-intelligence"
    }
}
# Logstash is started via: sudo service logstash start [41, 42]
```

B. Threat Intelligence Data Processing Logic (Python Example)

This Python snippet demonstrates the logic used to transform raw threat intelligence data, analogous to processing and enrichment filters within Logstash:

```
# Example 1: Threat Intelligence Data Processing

# Define sample raw threat intelligence data
threat_intelligence_data = [
    { "id": "TID-123", "type": "IP", "data": "192.168.1.1" },
    { "id": "TID-456", "type": "DOMAIN", "data": "example.com" }
]

# Function to process and transform the data
def process_threat_intelligence(data):
    # Transform the data structure
```

```
    transformed_data = {}
    transformed_data["log_message"] = data["data"]
    transformed_data["threat_id"] = data["id"]
    transformed_data["threat_type"] = data["type"]
    return transformed_data

# Process the data
processed_threat_intelligence_data = []
for threat_intelligence in threat_intelligence_data:
    processed_threat_intelligence = process_threat_intelligence(threat_intelligence)
    processed_threat_intelligence_data.append(processed_threat_intelligence)
# The resulting processed data is ready for indexing in Elasticsearch [43].
```

4. Results: Real-Life Threat Detection and Advanced Use Cases

The pipeline demonstrated successful detection of basic threats (brute force, port scan) and was validated against high-level, real-world SIEM use cases.

4.1 Case Example: Correlating Advanced Persistent Threats (APTs)

SIEMs detect APTs by correlating long-term, seemingly benign events into a broader threat picture. APT campaigns often unfold over weeks or months.

• **Threat Scenario:** Detecting slow-moving intruders stealing sensitive intellectual property.

• **SIEM Action:** The SIEM correlates unusual DNS queries, unauthorized access attempts to databases, and small, repeated outbound data transfers over non-standard ports.

• **Business Value:** This correlation reveals the multi-stage nature of the APT, enabling the detection of sophisticated threats and providing **enhanced support for retrospective investigations and forensic timelines**.

4.2 Case Example: Cloud Visibility and Misconfiguration Detection

Modern SIEMs must integrate with cloud platforms to provide unified visibility into environments like AWS and Azure.

• **Threat Scenario:** Unauthorized privilege escalation or misconfiguration in AWS.

• **SIEM Action:** The SIEM ingests **AWS IAM access logs** and **S3 access logs** (e.g., via a Logstash S3 input configuration for CloudTrail logs). It then detects the **creation of a new IAM user with administrator privileges** during non-business hours, followed by access to sensitive S3 buckets.

• **Business Value:** Enables **centralized monitoring** of cloud assets and identities, facilitating the detection of misconfigurations and unauthorized access.

4.3 Case Example: Automated Response via SOAR Integration

SIEM integration with SOAR (Security Orchestration, Automation, and Response) technology allows automated response workflows. Splunk offers native SOAR capability (rated 100/100 support for automation of security response workflows).

• **Threat Scenario:** Identifying and containing active ransomware activity (e.g., mass file encryption).

• **SIEM Action:** The SIEM detects signs of ransomware, such as **mass file encryption attempts** and **registry modifications**. This triggers the SOAR platform.

• **Response:** The SOAR platform **automatically isolates the affected machine** from the network, sends notifications, and creates a case in the ticketing system.

• **Business Value:** Provides **minimized response time** and containment of threats, reducing analyst fatigue and human error.

4.4 Case Example: Monitoring User Behavior (UEBA)

Modern SIEMs track user behavior to identify anomalies, often leveraging **UEBA (User and Entity Behavior Analytics)**.

• **Threat Scenario:** Insider threat or compromised credentials.

• **SIEM Action:** The SIEM detects a user who typically accesses systems during business hours from a local IP now **accessing a critical finance server at 3 a.m. from a foreign IP**. This deviation triggers a medium-severity alert.

• **Business Value:** Enables **detection of suspicious user behavior patterns** and provides an early warning for compromised credentials.

5. Comparative Analysis and Findings

The comparison highlights the primary differences between the two leading SIEM platforms.

5.1 Analyst and User Ratings Comparison

Metric	Splunk Enterprise Security	Elastic Security (ELK)	Details/Source
Analyst Rating	**93** (Highest among comparable tools)	82	Based on SelectHub's 400+ point analysis.
User Sentiment Rating	87% ('great')	**90% ('excellent')**	Based on aggregate user reviews.
UEBA Support	**100/100** (Tier 1: Fully supported out-of-the-box)	60/100 (Tier 3: Requires custom development/partner integrations)	Splunk leads significantly in native behavioral analytics.
SOAR Support	60/100 (Tier 2: Supported with workarounds/add-ons)	**100/100** (Tier 1: Fully supported out-of-the-box)	Elastic fully supports SOAR natively.
Dashboards & Reporting	**100/100** (Tier 1 Support)	75/100 (Tier 1 Support 71%)	Splunk dashboards are highly refined and intuitive.

Learning Curve	Moderate/Steep	Surprisingly Flat/Moderate	ELK is often cited as easier to learn for log processing.

5.2 Key Findings Summary

The sources confirm that both platforms are powerful, enterprise-grade log management and analysis platforms.

• **Splunk Strengths:** Splunk provides **advanced threat detection** capabilities and comprehensive security capabilities. It has a more user-friendly interface focused on **search-based analytics** and **easy-to-use SPL queries**. Users find it highly effective for real-time monitoring and incident response.

• **Splunk Weaknesses:** The primary barriers are the **high proprietary licensing cost**, the **complex initial setup**, and the **learning curve** required for specialized analyses.

• **ELK Strengths:** The ELK Stack is based on open-source components, offering **superior scalability** for large datasets. It is highly customizable and supports standard **RESTful APIs and JSON** for extensibility. It is known for effective threat detection and behavioral analytics.

• **ELK Weaknesses:** The **complex initial setup** and configuration can be challenging, requiring dedicated expertise. Although the software is free, the **total cost of ownership (TCO) can be substantial** due to hardware and storage requirements for expansive infrastructures.

6. Conclusion and Future Work

6.1 Conclusion

The project successfully demonstrated the implementation of a robust, end-to-end SIEM pipeline using both Splunk and the ELK Stack, proving that both architectures can achieve the goal of **unified threat detection, investigation, and response (TDIR)**. All stages of the pipeline—from log collection via **Beats/Forwarders** to alert generation via **Kibana/SPL**—were validated through simulated attacks and applied to real-world threat detection use cases (APTs, cloud misconfiguration).

The choice between platforms depends heavily on organizational strategy:
• **Splunk** is the leader for organizations prioritizing **out-of-the-box advanced features** (like **UEBA**), **ease of use** (SPL), and strong **enterprise readiness**, despite the high premium cost.
• **The ELK Stack** (Elastic Security) is the ideal choice for budget-minded organizations requiring **unmatched scalability**, **flexibility**, and a customizable open-source core, provided they invest in the necessary technical expertise for complex configuration and ongoing maintenance.

6.2 Future Enhancements

Future iterations of this SIEM project should focus on enhancing **automation** to move beyond detection and into robust incident remediation.
1. **SOAR Integration:** Integrate the SIEM with **SOAR** technology (supported natively by Splunk and fully supported by Elastic) to **automatically execute playbooks** in response to high-severity alerts (e.g., blocking an IP detected in a port scan or isolating a host upon ransomware detection).
2. **Expanded Endpoint Detection:** Integrate dedicated **Endpoint Detection and Response (EDR) telemetry** to allow the SIEM to correlate endpoint behavior (like suspicious PowerShell processes) with broader network and authentication data, improving alert confidence.
3. **Advanced Alerting:** For the ELK Stack, automate detection rules using tools like **ElastAlert**.

7. References

- **ELK Documentation: Elastic.co/docs.**
- **Splunk SIEM Resources: [IBM Knowledge Center].**
- **Splunk SPL Examples: [GitHub - root4oz/SIEM-Analysis].**
- **ELK Lab Implementation: [GitHub - 00112244/SIEM-Implementation-with-ELK-Stack].**
- **Logstash Configuration: [Codez Up]**