

**Jordan University of Science and Technology (JUST)**

**Faculty of Computer and Information Technology (FCIT)**

**Project Title**

**Custom SIEM System for Small to Medium Enterprises Focus on Open-source tools**

**Prepared By:**

**Majed Abdallah Alabed**

**Sara Khaled Darweesh**

**Tala Khaled Saleh**

**Takwa Abdel-Monem Shatnawi**

**Supervised By:**

**Dr. Heba Alawneh**

**Project Submitted in Partial Fulfilment of the Requirements for the Degree of Science in Cybersecurity**



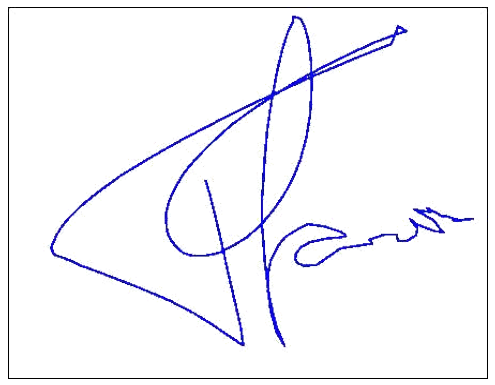
**Submitted: January 2025**

**Declaration of Originality**

This document has been written entirely by the undersigned project team members. The source of every quoted text is cited, and there is no ambiguity in where the quoted text begins or ends. The source of any illustration, image, or table that is not the work of the team members is also clearly cited. We know that using non-original text or material or paraphrasing or modifying it without proper citation violates the university’s regulations and is subject to legal action.

Names and Signatures of team members:

| Name 1:Majed Abdallah Alabed | Name 2:Sara Khaled Darweesh | Name 3:Tala Khaled Saleh | Name 4:Takwa Abdel-Monem Shatnawi |
| --- | --- | --- | --- |
| Student ID:151151 | Student ID:152297 | Student ID:153217 | Student ID:155862 |
| Signature 1: | Signature 2: | Signature 3: | Signature 4: |

****

**Acknowledgments**

First, we thank Allah Almighty for giving us strength and guidance in our work. We would also like to extend special thanks to our supervisor Dr. Heba Alawneh for her support and helpful advice. Her encouragement and dedication helped us improve our project. Special thanks to our families for their endless support and encouragement. Finally, we thank our friends for their companionship, which has been a great source of support throughout this journey.

**Table of Contents**

[Declaration of Originality i](#_heading=h.30j0zll)

[Acknowledgments](#_heading=h.1fob9te) ii

[Table of Contents iii](#_heading=h.3znysh7)

[List of Figures v](#_heading=h.2et92p0)

[List of Tables vi](#_heading=h.tyjcwt)

[Abbreviations vii](#_heading=h.3dy6vkm)

[Abstract viii](#_heading=h.1t3h5sf)

[Chapter 1](#_heading=h.4d34og8) Introduction 1

[1.1](#_heading=h.2s8eyo1) Overview 1

[1.2](#_heading=h.17dp8vu) Problem Statement 2

[1.3](#_heading=h.3rdcrjn) Significance of Study 3

[1.4](#_heading=h.26in1rg) Study Objectives 6

[1.5](#_heading=h.lnxbz9) Study Contribution 7

[1.6](#_heading=h.35nkun2) Outline of the Report 8

[Chapter 2](#_heading=h.1ksv4uv) Project Plan 9

[2.1](#_heading=h.44sinio) Project Deliverables 10

[2.2](#_heading=h.2jxsxqh) Project Tasks 10

[2.3](#_heading=h.z337ya) Roles and Responsibilities 12

[2.4](#_heading=h.3j2qqm3) Risk Assessment 13

[2.5](#_heading=h.1y810tw) Cost Estimation 18

[2.6](#_heading=h.4i7ojhp) Project Management Tools 18

[Chapter 3](#_heading=h.1ci93xb) Literature Review and Related Work 20

[3.1](#_heading=h.3whwml4) Related Work 20

[3.2](#_heading=h.2bn6wsx) Knowledge Gap 21

[Chapter 4](#_heading=h.qsh70q) Requirements Specification 23

[4.1](#_heading=h.3as4poj) Stakeholders 23

[4.2](#_heading=h.1pxezwc) Platform Requirements 24

[4.3](#_heading=h.49x2ik5) Functional Requirements 27

[4.4](#_heading=h.2p2csry) Non-Functional Requirements 29

[4.5](#_heading=h.147n2zr) Other Requirements 29

[Chapter 5](#_heading=h.3o7alnk) System Design 30

[5.1](#_heading=h.23ckvvd) Architectural Design 31

[Chapter 6](#_heading=h.1mrcu09) Conclusions and Future Work 32

6[.1](#_heading=h.46r0co2) Conclusions 32

6[.2](#_heading=h.2lwamvv) Future Work 33

[References](#_heading=h.111kx3o) 34

[Appendices](#_heading=h.3l18frh) 35

[Appendix A](#_heading=h.206ipza) 35

**List of Figures**

## [Figure 5.1: Architecture of system.](#_heading=h.ihv636)

[**Figure 5.1.2: Use Case Diagram**](#_heading=h.immk4tr7vufs)

[**Figure 5.1.3: Component Diagram**](#_heading=h.sdsawsofn8am)

[**Figure 5.1.4: Activity Diagram**](#_heading=h.vke5bkr778jj)

[**Figure 5.1.5: Sequence Diagram**](#_heading=h.unsx8wgfmi6)

**List of Tables**

[Table 2.2: project tasks](#_heading=h.2jxsxqh)

[Table 2.3: Roles](#_heading=h.z337ya)

[Table 2.4.1: Organization Context Assessments](#_heading=h.chc067dt48c7)

[Table 2.4.2: Hardware Assets](#_heading=h.gwfr5llkyi5s)

[Table 2.4.3: Software Assets](#_heading=h.gbkenyibwzsa)

[Table 2.4.4: Information Assets](#_heading=h.p2kz8plovamy)

[Table 2.4.5: Detailed risk: Analysis](#_heading=h.6foh34p8v07x)

[Table 2.4.6: Detailed risk: Design](#_heading=h.t1ncoq5qglki)

[Table 2.4.7: Detailed risk: Implementation](#_heading=h.aropb2hey6lf)

[Table 2.4.8: Detailed risk: Risk treatment](#_heading=h.t7ia5yp75zx3)

[Table 2.4.9: Detailed risk: Control](#_heading=h.cqcrb8oqdjiz)

[Table 2.5: Cost](#_heading=h.1y810tw)

[Table 4.1: Stakeholders](#_heading=h.3as4poj)

[Table 4.2.1: Hardware Requirements (Server-Side](#_heading=h.n5knde6ea63n))

[Table 4.2.2: Software Requirements (Server-Side)](#_heading=h.gepv4454q8hw)

[Table 4.2.3: Network Requirements (Server-Side)](#_heading=h.k575trnj0q8)

[Table 4.2.4: Hardware Requirements (Client-Side)](#_heading=h.kplnuz827i1b)

[Table 4.2.5: Software Requirements (Client-Side)](#_heading=h.fkjlbelt8ipy)

[Table 4.3: Functional Requirements](#_heading=h.49x2ik5)

[Table 4.4: NonFunctional Requirements](#_heading=h.2p2csry)

[Table 4.5: Others](#_heading=h.147n2zr)

[Table 5.1: Architectural Design](#_heading=h.23ckvvd)

**Abbreviations**

List the abbreviations you have used in your project if there are any, and what they stand for.

**SIEM** Security Information and Event Management

**SMEs** Small and medium enterprises

**SIM**  Security Information Management

**SEM** Security Event Management

**AI** Artificial Intelligence

**IDS**  Intrusion Detection System

**IPS**  Intrusion Prevention System

**CSV**  Comma-separated value

**TIP** Threat Intelligence Platform

**UEBA** User and Entity Behavior Analytics

**UML** Unified Modeling Language

**JSP** Java Server Pages

**ELK** Elasticsearch Logstash Kibana

**Custom SIEM System for Small to Medium Enterprises Focus on Open-source tools**

**By**

**Majed Abdallah Alabed**

**Sara Khaled Darweesh**

**Tala Khaled Saleh**

**Takwa Abdel-Monem Shatnawi**

**Supervisor**

**Dr. Heba Alawneh**

**Abstract**

As we see in this era the cyberattack has been raised,cybersecurity become a critical concern for the organisation of all sizes,medium to small enterprise face unique challenge due to the limited resource making it difficult to implement a effective SIEM systems soutlion and its essential tools to detecting to cyber threats in real time, and the SIEM solution that's existing in the market seems to expensive and complex of focus on the large organization that's mean leaving the smaller organization undeserved.

In our project we aim to design and implement a cost-effective, scalable, and user-friendly SIEM solution focused on the needs of small to medium enterprises.The solution will focus on integrating open source tools, automating threat detection , and providing actionable insights to enhance security operations .our project seeks to bridge the gap between affordability and effective cybersecurity,offering a practical and accessible approach to threat management.

# Introduction

## 1.1 Overview

**1.1.1 Importance of the project:**

Medium to small enterprises now become target to the cyberattacks,because they often lack the advanced security measures the larger organization,as many SIEM solution they designed focus on the large enterprise which make them too expensive for the small to medium enterprise.that leaves a gap where small to medium enterprise vulnerable to attacks.

The importance on our project is to solve the gaps by provide a SIEM solution for low to medium enterprise that are practical and easy accessible.that we will focus to affordability ,ease of use ,scalability,the project focus on to give power to small to medium enterprise to enhance there security in cybersecurity posture without extensive resource.this help protect the small to medium enterprise to block the entry points for the attacker as we know the small enterprise is a target.

**1.1.2 Motivation:**

The motivation for this project is to protect the small to medium enterprises during the growing of cybersecurity attack and make a effective cybersecurity solutions that are accessible for these enterprises and our teams is ride to be a blue team guys to protect all the assets for enterprise and make us get a lot of information and type of the attacks that we gonna protect or mitigations additionally our project is giving opportunity to explore the integrations of opens source tools and methodologies in building a SIEM solution . This is not only to reduce costs but to promote flexibility and customization allowing the solution to be adapted to the needs of different organization.The project is also driven by the goal of advancing cybersecurity by developing a framework that may act as a basis for further study and advancement in the sector.

**1.1.3 Scientific and Technical Background:**

A SIEM system is a centralized platform the collects and analyze log data from various sources,such as firewall,servers, applications,and endpoints,to detect and respond to security incidents it combines **two functions:**

**Security Information Management (SIM):** focuses on long term storage,analysis and reporting of data.

**Security Event Management (SEM):** provide real time monitoring,correlation,and alerting of security events.

**-The following are the main parts of a SIEM system:**

**Data collection:** combining log information from many sources.

**Transforming:** unprocessed log data into a consistent format for analysis is known as normalization and parsing.

**Finding patterns and connections:** between occurrences in order to identify possible risks is known as a correlation engine.

**Alerting and Reporting:** Sending thorough reports for additional investigation and alerting security personnel of questionable activity.

**1.1.4 Objectives**:

The primary objective of this project is to build and implement a SIEM solution that will meet all the specific needs of small to medium enterprises.

**The project will focus on:**

**Affordability:** we will use open source tools to reduce costs.

**Scalability:** if the enterprises grow the data will increase we will ensure that the system can handle this data.

**Ease of use:** providing a user friendly interface and natural workflows for security teams.

**Effectiveness:** enabling real time threat detection and response to enhance security.

By achieving these objectives the project aims to provide valuable information to the field of cybersecurity and will help to address the challenges faced by limited resource organizations.

## 1.2 Problem Statement

**1)** **Precise description of the problem**

Small and medium enterprises (SMEs) are often easy targets for cyberattacks, because they have some challenges in implementing the solutions, due their small and limited budget, limited techniques, and lack of the resources. Meanwhile the cyberthreats such as insider threats are becoming more common.

Our project aims to address these gaps by developing an open-source, cost-effective SIEM system tailored for SMEs. The solution will focus on insider threats, real-time threat monitoring, and customizable alert systems, enabling businesses to strengthen their defenses without incurring high costs.

**2)Target Audience**

* **Small and Medium enterprises:** organizations with limited technical resources and budget that require affordable, easy-to-use cybersecurity solutions.
* **It and Security Teams :** Responsible for monitoring threats, alerts, and incidents in real time, and they will gain value from user-friendly tools that minimize manual work while enhancing the cyber threat detection.
* **Non-Technical Teams:**

**-HR Teams:** Can monitor insider threats.

**-Marketing and social media Teams:** Can secure their platforms against thinker property theft.

**3) Expected Outcomes**

1. Enhanced Cybersecurity Measures.

2. Increased Usability and Accessibility.

3. Cost Saving.

4. Operational Efficiency.

5. Scalability for Future Growth.

## 1.3 Significance of the project

**1.3.1 Opening Statement**

This project aims to empower small to medium-sized enterprises (SMEs), particularly non-IT organizations like marketing companies, to address a critical gap in the cybersecurity field. SMEs are becoming the prime targets for cyber threats, including data breaches, and insider threats, due to limited cybersecurity expertise and training.

**1.3.2 Current Challenges**

1. Configuration Complexity: As a result, it is possible to distinguish a configuration of an SIEM system to the needs of an enterprise in question in the course of the implementation phase. Scoping which data source will be incorporated in the output, fine-tuning correlation rules and in fact, tuning a particular alert, calls for immediate consideration of every minor detail associated with it.Controlling which kind of data is to be merged into the output, coding of correlation rules, the fine tuning of alerts, each and every aspect requires a lot of focus.n. Determining which data source is to be incorporated into the output, fine-tuning correlation rules, and actually tuning the alerts requires special consideration of every little detail that goes with it. From deciding which data sources are to be integrated with the output to configuring correlation rules and tuning the alerts, attention to each and every detail is a must.
2. Noise in Event Data: SIEM platforms typically process vast amounts of logs, which can lead to significant noise, for example: large amounts of non-threat-related data that confuse actual security threats. This noise can potentially cause critical warnings to be overlooked among false positives.
3. Underprepared SMEs: Small to medium enterprises often have a gap in IT generally, making them vulnerable to cyberattacks.
4. Alert Fatigue: It occurs when security teams are overwhelmed by a high amount of alerts, many of which may be false positives. This can lead to slower response time, overlooked alerts, and increased risk of missing actual threats.Such problems can reduce the effectiveness of the SIEM.
5. Integration Hurdles: Integrating SIEM tools seamlessly with existing tools and systems can be a challenge. The lack of compatibility can hinder the SIEM's ability to provide the full view of security events.

**1.3.3 Contribution to the Field`**

1.Threat Detection and Response: According to Microsoft the most common use case for a SIEM solution is threat detection and response. It can help in uncovering and responding to even some of the most complex threats, such as insider threats, advanced persistent threats, and multidomain attacks.

2.Customizable for Non-Technical Enterprises: Make it easy to engage with the SIEM without so much training by providing interactive dash and automated reports.

**1.3.4 Practical Applications and Benefits**

1.Cost Effective: Using open-source solutions in addressing modern security challenges and compliance with regulatory requirements. This provides an offer for the organizations and the academic community that are seeking cost-effective security solutions.

2.Advanced Visibility: The deployed SIEM agent within the organization’s network is able to correlate data spanning an organization’s entire attack surface, endpoint, and network data, as well as firewall logs and antivirus events. This capability offers a comprehensive view of data.

3.Efficient Log Handling: SIEM earned its niche based on the speed at which it aggregates related security incidents into prioritized alarms.Logs are coming from different sources and they are sending it here it is doing the correlation and the analysis of the logs.ed alerts. The logs are directed to a common logging storage from different sources, it performs the correlation and the analysis of the logs.

4.Better Threat Detection and Response: Each SIEM system has a correlation engine that helps analyze data to look for threats and trigger an alert when programmed to do so. Meanwhile, the engine is also watching logs; the Threat Intelligence Platform (TIP) is designed to detect and mitigate known threats. Also, User and Entity Behavior Analytics (UEBA) apply the use of machine learning to identify insider threats.

5.Compliance Support: Instead of manually compiling data from various hosts within the IT network, SIEM automates the process, reducing the time by making automated reports and providing a full view of what's happening.

**1.3.5 Alignment with Industry Trends**

1.Rise in Targeted SME Attacks: Attackers are targeting SMEs as entry points to larger supply chains. This project responds to the growing demand of affordable and effective cybersecurity tools for smaller businesses.

## 1.4 Project objectives

**1.Enhance Real-time Analysis:** Modern SIEM provides you with real-time analytics whereby your company’s SIEM administrators can increase the efficacy of security alerts in many ways, including confirming security alerts and events.

**2.Effective Threat Detection:** By enabling continuous monitoring and analysis of network activities. SIEMs utilize predefined signatures and behaviors, as well as anomaly detection, to spot recognized threats and identify new ones. It empowers organizations to mitigate cyber risks by discovering vulnerabilities earlier and minimizing damage caused by breaches.

**3.Efficient Incident Response Mechanisms:** Incident Response in a SIEM signifies the ability to rapidly respond to detected security incidents. When a SIEM identifies unusual or potentially harmful activities, the incident response function immediately and takes action.

**4.Address Industry-Security Needs**: Provide a solution to the challenges that the marketing companies face, including: handling sensitive client and consumer data and protecting intellectual property like designs and marketing campaigns.

**5.Provide Cost-Effective Security for SMEs:** Present a scalable and affordable solution that minimizes the financial burden for SMEs while delivering a good grade of security capabilities.

**6.Promote Cybersecurity Awareness for Non-IT SMEs:** People tend to trust that cybersecurity is dealt with in other parts of the company or by other people (such as third party services they may use as part of their business infrastructure). In the context of the general observations, individuals within SMEs don’t only need the awareness, but must also accept their responsibility in maintaining robust defences against cyber risks.

**7.Ensure Scalability for Future Growth:** Design the SIEM solution to scale with organizational growth. By enabling your SIEM systems to scale efficiently, you can manage the daily flow of security data generated, ensuring that no potential threats go unnoticed. A scalable SIEM solution not only helps in real-time threat detection and incident response but also offers flexibility in adapting to evolving security landscapes.

## 

## 1.5 Project Contribution

**1.5.1 Project Novelty**

This project provides a cheap SIEM solution targeting the small business companies and departments.businesses, it is aimed to meet a significant gap where it is needed. Unlike traditional, expensive SIEM systems, and as such is relatively cheap, easy to implement and targets responding to some general issues small companies troubleshoot such as detecting insider threats, providing a real solution to SME’s for improving their security posture.

**1.5.2 Project Audience**

The project, benefits many groups within an organization:

* **IT and Security Teams:** it gives real time means for threat identification and threat handling.
* **Cloud Users:** Get better data protection.
* **HR Teams:** can track and even prevent insider-related threats.
* **Executives and Insurers:** Get brief and unambiguous risk information that can inform decisions.
* **Marketing, Social Media, and Design Teams:** Be safe from these (theft of intellectual property).
* **Legal and PR Teams:** Apply SIEM reports to orchestrate and minimize breaches while keeping reputational losses to the minimum to ensure that small businesses have robust and appropriate protection for a relatively cheap price.

**1.5.3 Project Model Novelty**

ELK and other tools are used in a new way in the project. These tools are often regarded as being appropriate for large enterprises only, but, in this project, both are made simple.Adapts them to suit personal needs in order to make them easy to use. By doing this, it brings an affordable plan into existence.Effective and affordable solution that stands between powerful SIEM systems and simple tools for analysis security and limited budgets.

**1.5.4 Project Structure of the pipeline**

The structure of the project breaks the conventional pipeline design approach and is considered to be unique.on the effective fine-tuning of the correlation rules and the detection algorithms in order to eliminate false positives – a typical issue of most conventional SIEM solutions. It refines detection.

This work also covers the rules and the most complex methods that are used to enhance the reliability of the system. It also can easily work with other security tools, such as firewalls, IDS, and IPS to form integrated solution that consolidates all security operations and accelerates the methods.

This is a welcome innovation for small business since there is this smooth pipeline.

## 1.6 Outline of the report

**Chapter 1: introduction**

In this chapter we discuss the challenges that faces SMEs. The most important

challenge in the small and medium -sized businesses is limited budgets, lack of

technical tools, and this chapter highlights how we need to solve these issues by

providing low-cost, open-source solutions to deal with the insider threats.

**Chapter 2: project plan**

This chapter outlines the planning stage, describing the project deliverables, required.tasks, and timelines. It includes analyses of roles and responsibilities, risk assessments, and cost estimations. and it explains the tools used for project management and collaboration.

**Chapter 3: Literature Review and Related Work**

This chapter talks about past research and projects in cybersecurity. It explains the issues with current solutions, like being too costly, hard to use in real life, and not using AI much. It also shows how the new idea fixes these problems and offers something better.

**Chapter 4: Requirements Specification**

This chapter specifies the technical and non-technical requirements of the system. It identifies key stakeholders and describes their interactions with the system. Functional, non-functional, and other specific requirements, and data storage formats.

**Chapter 5:** **System Design**

This chapter explains how the system is planned and organized. It includes pictures to show how everything fits together and works. Important parts like the database, user interfaces, and how data flows are described in detail.

# 

# Project Plan

## 2.1 Project Deliverables

The SIEM solution project aims to provide SMEs with a user-friendly, cost-effective system for threat detection. Deliverables focus on essential tools, guides, and resources to Ensure efficient implementation and effective monitoring.

**1. Source Code:** Customizable scripts written in Python for phishing and insider threat detection, Provided with configuration files to tailor the system to specific needs.

**2. Executable Files:** Easy-to-use installation files and scripts that simplify system setup, supporting both Windows and Linux environments.

**3. Documentation:** Comprehensive guides to support users and developers:

* **User Guide:** Step-by-step instructions for using and maintaining the SIEM system
* **Developer Guide:**Technical resources for customizing or upgrading the system.
* **Troubleshooting Guide:** Solutions for common issues.

**4. Database:** Pre -configured databases optimized for storing and analyzing .system logs and threat data. The databases are designed to be easily imported and include options for both SQL-based and NoSQL-based systems.

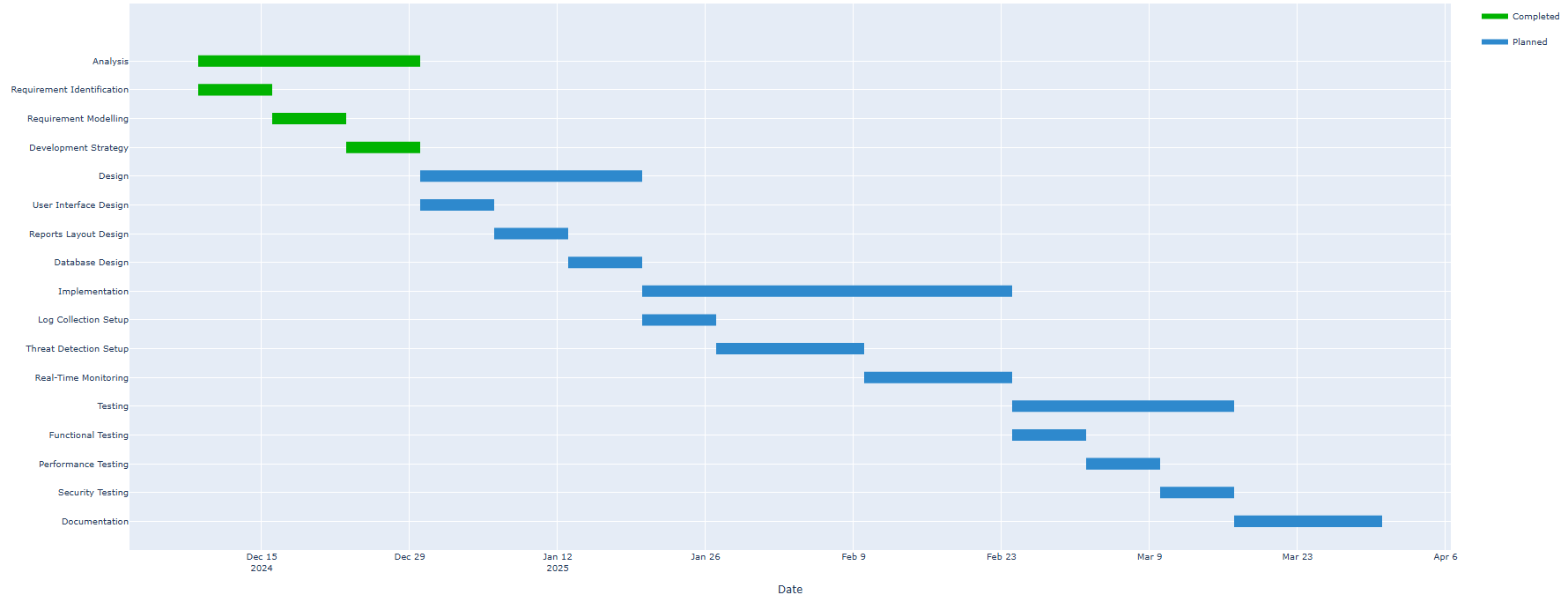
**5. Dataset**: Complete and collected datasets featuring insider threat scenarios. Provided in widely used formats such as CSV and Excel, they are compatible with machine learning tools.

**6.Testing Tools:** set of tools for validating system functionality, including predefined scenarios for assessing phishing detection and insider threat identification accuracy.  
**7. Dashboards:** web-based dashboards designed for real-time monitoring of the system. They display key insights such as detected phishing threats, alerts for suspicious activities, and overall system performance insights.

## 2.2 Project Tasks

**Table 2.2**

| **#** | **Task Name** | **Description** | **Time Duration** | **Dependencies** |
| --- | --- | --- | --- | --- |
| 1 | Analysis | Identification, modelling, and development approach of requirements. Scalability analysis, security, and fact-finding techniques. | 3 weeks | None |
| 1.1 | Requirement Identification | Identify requirements based on what stakeholders need | 1 week | None |
| 1.2 | Requirement Modelling | Model the requirements through  data flow diagram, use case  diagram or other tool to show how  structure of the SIEM system  (server-client), interfaces,  hardware/software specification. | 1week | Task 1.1  (Requirement Identification) |
| 1.3 | Development Strategy | Define the hardware, software tools for  The SIEM system  ( wazuh, ELK , postgreSQL,etc) | 1 week | Task 1.2  (Requirement Modelling) |
| 2 | Design | Design user interfaces, report layouts, and database schema. | 3 weeks | Task 1.3  (Development Strategy) |
| 2.1 | User Interface Design | It will depend on the SIEM solution that we will use for security analysts to interact with the SIEM system. | 1 week | Task 2 |
| 2.2 | Reports Layout Design | We will design the reports layout ,template, focusing on threat analysis , risk reporting , incident response, user behaviour | 1 week | Task 2 |
| 2.3 | Database Design | Design the database schema to store logs and  event and alerts | 1 week | Task 2 |
| 3 | Implementation | We will develop the system (event collect, processing ,pipeline) and implement important security measures (RBAC, encryption) | 5 weeks | Task 2 |
| 3.1 | Log collection setup | Implement the log collection tools via syslog , Collect logs in real-time from various sources and store them in centralized repository | 1 week | Task2 |
| 3.2 | Threat Detection Setup | Analyze log patterns using predefined rules and machine learning models to identify threats | 2 weeks | Task 3.1 |
| 3.3 | Real-Time Monitoring | Collect real-time data, correlate events, Tigger alerts based on predefined rules | 2 weeks | Task 3.2 |
| 4 | Testing | Do a a functional testing , performance testing system alerts, to make sure that the system work as we expected | 3 weeks | Task 3 |
| 4.1 | functional testing | Test the log collection ,threat detection | 1 week | Task 3 |
| 4.2 | performance testing | To ensure that system performance work as we expected and meet the requirements | 1 week | Task 4.1 |
| 4.3 | security testing | Perform penetration testing if we can to validate the security for SIEM | 1 week | Task 4.1 |
| 5 | Documentation | We will write important documentation  Including user guides , technical design documents | 2 weeks | Task 4 |



## 

## 2.3 Roles and Responsibilities

This table includes the work of each member in the group and the things each of them have searched for the most in the duration of this project.

**Table 2.3**

| **Name** | **Roles and Responsibilities** |
| --- | --- |
| Majed Abdallah Khaleel Alabed | Searched the options of open source tools that can be used for SME, key SIEM implementation challenges and what frameworks are best to integrate to enhance the SIEM. |
| Sara Khaled Faraj Darweesh | Performed a deep search about how to detect and filter spam emails effectively using ML techniques/tools by using AI-generated phishing emails and searched how to identify the insider threats involved in any organization and what ways are used to protect against these threats. |
| Tala Khaled Saleh Saleh | Studied how to implement a Wazuh based SIEM that improves the apparel sector’s threat monitoring and detection system, the most effective solution that is used to fix the alert fatigue problem by using machine learning to filter the false alerts and how to implement an intelligent-based SIEM that provides email alerts. |
| Takwa Abdel-Mon’em Mohammad Shatnawi | Covered how to integrate AI-based supervised classification with SIEM to enhance phishing detection, how the merge of AI and data lakes drive smarter threat detection in SIEM and searched the development of ML using neural networks for phishing. |

## 

## 

## 

## 

## 2.4 Risk Assessment

## 1.Organization Context Assessments:

**Table 2.4.1**

| **Context elements** | **Description** | **threats** | **Likelihood** | **Impact** | **Risk level** | **Mitigation strategy** |
| --- | --- | --- | --- | --- | --- | --- |
| Business Objective | Medium to low enterprise SIEM  implementation | **Security policy**  **violations** | **Possible** | **High** | **Critical** | we need a clear security objectives and policies |
| Scope | Security monitoring and incident response | **Scoop**  **creep** | **Likely** | **High** | **Medium** | good discussion about scope with stakeholders |
| Legal/Regulatory | Compliance Requirements for Enterprise Security | **Compliance violations** | **Almost**  **certain** | **High** | **Critical** | Regular compliance audits and updates |

## 

## 2.Asset-Based Risk Assessments:

## 2.1 Hardware Assets

**Table 2.4.2**

| **Risk Description** | **Threats** | **likelihood** | **Impact** | **Risk level** | **Mitigation**  **strategy** |
| --- | --- | --- | --- | --- | --- |
| **Hardware failure** | Component malfunction/  Overheating | Possible | High | **High** | **Redundant systems/**  **Temperature monitoring** |
| **Physical access violations** | Theft/ Unauthorized modifications | Unlikely | High | **Medium** | **Access controls/**  **Security cameras** |
| **Equipment failure** | Natural disasters/  Environmental factors/  Manufacturing defects | Possible | High | **High** | **Regular maintenance/**  **Environmental controls** |

## 

## 2.2 Software Assets

**Table 2.4.3**

| **Risk Description** | **Threats** | **likelihood** | **Impact** | **Risk level** | **Mitigation strategy** |
| --- | --- | --- | --- | --- | --- |
| **Software bugs** | Code vulnerabilities/  Memory leaks. | Almost Certain | High | **Critical** | **Regular updates** |
| **Misconfiguration** | Security settings errors/Insecure protocols enabled/  Default configurations left unchanged. | Almost Certain | High | **Critical** | **Configuration management** |
| **System crashes** | Resource exhaustion/  Memory corruption. | Possible | High | **High** | **System monitoring** |

## 

## 2.3 Information Assets

**Table 2.4.4**

| **Risk Description** | **Threats** | **likelihood** | **Impact** | **Risk level** | **Mitigation strategy** |
| --- | --- | --- | --- | --- | --- |
| **Data corruption** | Hardware malfunctions/  Malicious code | Possible | High | **High** | **Data integrity checks** |
| **Unauthorized access** | Credential theft/  Privilege escalation/  Weak authentication/ | Likely | High | **Critical** | **Access controls** |
| **Data loss** | Hardware failures/  Accidental deletion/  Natural disasters/  Ransomware | Possible | High | **High** | **Backup systems** |
| **Data breach** | SQL injection/  Network intrusion/  Advanced Persistent Threats | Possible | Critical | **Critical** | **Security**  **controls** |

## 

## 3.Detailed risk analysis by project phase:

## 

## phase1:Analysis

**Table 2.4.5**

| **Task** | **Vulnerabilities** | **Threats** | **Likelihood** | **Impact** | **Risk level** | **Controls** |
| --- | --- | --- | --- | --- | --- | --- |
| Requirement identification | Incomplete system | Incomplete requirements | **Likely** | **High** | **High** | Structured gathering |
| Requirement modelling | Design errors | Implementation errors | **Possible** | **High** | **High** | Expert review |
| Development strategy | Tool selection errors | Project delays | **Possible** | **High** | **High** | stakeholders evaluation |

## 

## 

## 

## :Design

**Table 2.4.6**

| **Task** | **Vulnerabilities** | **Threats** | **Likelihood** | **Impact** | **Risk level** | **Controls** |
| --- | --- | --- | --- | --- | --- | --- |
| UI design | Usability problems | User rejection | **Unlikely** | **Medium** | **Medium** | User testing |
| Reports layouts | Information gaps | Decision delays | **Possible** | **Medium** | **Medium** | Template validation |
| Database design | Performance issues | System slowdown | **Likely** | **High** | **High** | we will do a performance testing |

## 

## phase3:Implementation

**Table 2.4.7**

| **Task** | **Vulnerabilities** | **Threats** | **Likelihood** | **Impact** | **Risk level** | **Controls** |
| --- | --- | --- | --- | --- | --- | --- |
| Log collection | Collection gaps | Missing alerts | **Almost Certain** | **High** | **Critical** | validate the collection of logs |
| Threat detection | Detection gaps | False positives | **Likely** | **High** | **Critical** | Rule setting /  do a testing scenarios |
| Real-Time Monitoring | Performance issues | the detection will be delay /  system overload | **Likely** | **High** | **Critical** | Do a performance testing/  testing how data will load |

## 

## 

## 

## phase4:Risk treatment strategies

**Table 2.4.8**

| **Risk Level** | **Risk treatment** | **Application** | **Priority** |
| --- | --- | --- | --- |
| **Critical** | **Risk Avoidance** | Software bugs/Compliance violations/  Log collection gaps/Real-time monitoring issues | **Critical** |
| **High** | **Risk Mitigation** | Hardware failures/System crashes/  Database performance/Requirement gaps | **High** |
| **Medium** | **Risk Transfer** | Physical access violations/  Environmental factors/  UI issues | **Medium** |
| **Low** | **Risk Acceptance** | Non-critical reporting delays/  Minor UI issues | **High** |

## 

## phase5:Control implementation Framework

**Table 2.4.9**

| **Control type** | **Controls** | **Priority** | **Monitoring Frequency** |
| --- | --- | --- | --- |
| **Management controls** | Security policies and procedures/  Risk assessment framework/  Change management | **High** | **Monthly** |
| **Operational controls** | security awareness/  incident response procedures/  System maintenance/  Configuration management/  Backup procedures | **High** | **Weekly** |
| **Technical control** | Access control systems/  Encryption mechanisms/  Performance monitoring/  Log collection validation//  Threat detection rules | **Critical** | **Daily** |

## 2.5 Cost Estimation

Here in our company, our SIEM project includes some free products that make the cost less sensitive; thus, suitable for small businesses. These are the operating system Linux distro’s such as Ubuntu, SIEM software like Wazuh and the ELK stack, databases such as PostgreSQL & MySQL, web server like Apache HTTP server.Other utilities of log management (Syslog etc) backup solutions (i.e Duplicate etc) are also free.The costs are further kept low by various open source choices for the Open source JVM which is the Java runtime environment (for example Open JDK).Table 2.5 will cross-list all the hardware it needs to actualize our project.g them excellent choices for small businesses. These include the operating system (Linux distributions like Ubuntu), SIEM software options like Wazuh and the ELK suite, database management systems like PostgreSQL and MySQL, and web servers like Apache. Also tools of log management (Syslog etc) Backup solutions (i.e. Duplicate etc) are also free. Open-source options for the Open source JVM (Java runtime environment (e.g., Open JDK) also keep the costs further. Table 2.5 will enumerate all the hardware that it takes to realize our project.

**Table 2.5**

| **Component** | **Specification** | **Cost** |
| --- | --- | --- |
| **CPU** | Quad core processor | 100 JOD |
| **Ram** | 16GB | 40 JOD |
| **Storage** | 1TB SSD OR HDD | 40 JOD |
| **Network interface** | Gigabit Ethernet (1Gbps) | 20 JOD |
| **Power Supply** | Redundant Power Supply | 100 JOD |
| **Firewall** |  | 130 JOD |
| **Bandwidth** | 1GB | 355 JOD/month |
| **Training** | Yearly | 120 JOD |
| **Total** | -------- | 795 JOD |

## 2.6 Project Management Tools

Project Requirements To help the SIEM project to be developed efficiently in a small organization, the following tools will be used:

**1. SIEM Software:** ELK cloud is the primary software for being a SIEM platform.

**2. Environment Operating System:** Linux Ubuntu / Windows 10

**3.Google Drive:** we will use this for storage of documents, sharing them and collaboration. It helps centralize project documentation, meeting notes, and other resources.

**4.Lucidchart. com:** for creating diagrams.

**5.Discord:** used for coordination meetings and communication during projects and teamwork.

**6.Git(GitHub):** for version control, to manage the development workflow, tracking modifications, and for collaborative developement. We will host the project on GitHub repositories.

**7.Microsoft Word:** Create report containing the project websites and other research related.

**8.Google Scholar:**determine potential works and research related to the project topic.

**9.Sysmon:** By using sysmon, we can send the log info of all our network devices to one centralized place.

**10.Outlook:** we used it to send alerts.

# Literature Review and Related Work

## Related Work

Common studies on Security Information and Event Management analysis and on machine learning software and applications are suggestive of high potential in making cybersecurity better. Various studies have looked at how threat detection could be enhanced by incorporating SIEM with AI as well as minimizing the generation of false alarms and provide affordable solutions for SMEs.

In this case, one important research was carried out investigating how the incorporation of AI endow tools with SIEM tools in the detection of phishing attacks. The study established that using machine learning, the detection rate was improved, leading to fewer false alarms[1]. Another study created a neural network with a capability of recognizing more than 80 percent of phishing mails and at the same time also failing to signify genuine mail as being hazardous.These studies tended to use small and possibly skewed datasets and were evaluated solely in laboratory settings, making it a bit hard to apply the outcomes in real world situations.[2]

Other research works concentrated on integrating AI into big data storage systems within SIEM solutions, to improve the chances of identifying new attacks, and emerging malware such as zero-day threats. By doing so, Large datasets analysis became real-time, which improved the identification of intricate threats. The current implementations of threat intelligence, mainly plugging AI into prior SIEM implementations, was challenging, and previous versions generated excessive false positives. Another study explored how Wazuh-based SIEM systems might offer additional economical SIEM solution to particular fabrics such as apparel. Although these systems were cheaper and could generate automatic reports they had limited capability in machine learning and efficient monitoring of the user activities.For that matter, some studies suggested the use of enhanced machine learning algorithms to enhance filtering out of unimportant alerts and handling of incidents.dies have explored how integrating SIEM with artificial intelligence (AI) can enhance threat detection, reduce false alarms, and offer affordable solutions, especially for small and medium-sized businesses (SMEs).

A significant published paper discussed the integration of AI tools with SIEM systems for a better detection of Phishing attacks. The study also discovered that alert detection rate was enhanced by machine learning yet noted a decline in false positives[1]. In one study, researchers evolved a neural network model for identifying phishing emails that could recognize over 80% of the phishing emails without labeling legit ones as either bad Another study experimented with a dataset that was relatively small and unbalanced, and the effectiveness of the algorithms has been tested only in ideal conditions.[2]

Other researchers in the past also centered their work around the integration of IA with big data archival systems in SIEM solutions in order to more effectively identify emerging threats like zero day threats. They said this approach allowed analysis of Large datasets in real time and enhanced the ability to identify numerous and diverse threats. Introducing AI into currently established SIEM systems was challenging, and previous versions contributed to excessive false alarm settings. In another study carried out, there was an analyzed on how Wazuh-based SIEM systems could deliver low cost solutions that were especially suitable for certain industries such as the apparel industry. These systems were inexpensive to implement and capable of generating automated reports, but they did not have powerful machine learning features as well as the necessary level of tracking the user’s actions.

In attempts to reduce the problem of excessive and uninformative alerts, other research works suggested more sophisticated ML models for handling false alerts, while containing the corresponding hindrances to valid incidents.s. The above models where developed for the purpose of refining the methods of alerts ornotification, and to make their application more easy. Nevertheless, some of these frameworks still require that they be tested on real environments.[5] While primary research on this problem in the context of employing recognised guidelines, such as NIST and ISO/IEC 27001, for designing the SIEM system proved useful in this regard, they lacked information on the present-day concerns regarding AI-backed cyber threats.

In the specific case of SMEs, research established open-source SIEM tools as encouraging and inexpensive solutions. While these tools gave some insights on how to do this , none had the more sophisticated machine learning to reduce false positives [7]. Prior works on machine learning for spam detection pointed out the possibility of a number of process automations, but did not explain the actual timeline of email scanning or how such tools can be incorporated into SIEM systems.[8] Studying AI solutions to phishing detection also yielded substantial findings In most cases, however, such studies were still in a conceptual stage and not implemented practically.

Insider threat research aimed to understand how learning techniques could identify user activity and raise alarms about high risk activities. Such methods were good when it came to pattern recognition, however, for real time performance they were suboptimal because the results could only be computed based on previous data processing. Some of the problem areas include high false positive rate, integration of state of the art artificial intelligence, and lack of concern on how the cost of this product would be made reasonably affordable to SMEs: there is also no concern on how this product would need to be in real time, how the attendees’ behavior would be analyzed, or how it would need to adhere to local or international set guidelines. ML based approaches for detection of spam pointed out that there are possibilities of automation of different processes but did not refer to the real-time email scanning or how these tools could be integrated into SIEM systems.[8] Similar work in detecting phishing through AI was also heralded but much of them was still theoretical and had not undergone applied usage.

Insider threat detection conducted research on how machine learning could take an analysis of user activity and determine what are high risk activities. These methods were used as ‘supervised learning’ to identify unfamiliar trends; still, these methods do not allow for real-time operation and are based on analysis of historical data. This limitation is particularly significant for today’s improved versions of SIEM systems which need to monitor systems in real time to remain beneficial.

**Key Gaps in Research:**

Higher false positives, issues in incorporating future technologies, and neglect to small and medium-sized businesses were issues seen often. Timely features, utilization of the users’ behavior analysis, and the regulation issue were discussed less frequently.liance with regulations. Solving them will be relevant to building upgraded and more widespread SIEM solutions.

## Reference

### [1] Ferreira, G. (2020). Enhancing Phishing Defense Mechanisms with Information Security Event Management (SIEM) and AI-based Supervised Classifiers.

## [2] Bezerra, A., Pereira, I., Rebelo, M.Â. et al. A case study on phishing detection with a machine learning net. Int J Data Sci Anal (2024). https://doi.org/10.1007/s41060-024-00579-w

## [3] Marri, R., Varanasi, S., & Chaitanya, S. V. K. (2024). Integrating Security Information and Event Management (SIEM) with Data Lakes and AI: Enhancing Threat Detection and Response. Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023, 6(1), 151-165.

## [4] Islam, M. R., & Rafique, R. (2024). Wazuh SIEM for Cyber Security and Threat Mitigation in Apparel Industries. International Journal of Engineering Materials and Manufacture, 9(4), 136-144.

## [5] Ban, T., Takahashi, T., Ndichu, S., & Inoue, D. (2023). Breaking alert fatigue: Ai-assisted siem framework for effective incident response. Applied Sciences, 13(11), 6610.

## [6] Rosenberg, M., Schneider, B., Scherb, C., & Asprion, P. M. (2023). An adaptable approach for successful siem adoption in companies. arXiv preprint arXiv:2308.01065.

## [7]Manzoor, J., Waleed, A., Jamali, A. F., & Masood, A. (2024). Cybersecurity on a budget: Evaluating security and performance of open-source SIEM solutions for SMEs. Plos one, 19(3), e0301183.

## [8] Swapna Vanguru,Kethsy Prabavathy (2024). A Detailed Analysis of Machine Learning Algorithms in Spam Email Prediction. 9TH INTERNATIONAL CONFERENCE ON TECHNICAL ADVANCEMENTS IN COMPUTER SCIENCE AND ENGINEERING(ICTACSE-2024)At: Hyderabad. ISBN: 978-81-975624-7-1.

## [9] Eze, C. S., & Shamir, L. (2024). Analysis and prevention of AI-based phishing email attacks. *Electronics*, *13*(10), 1839.‏

## [10] Subhani, A., Khan, I. A., & Zubair, A. (2021). Review of insider and insider threat detection in the organizations. *Journal of Advanced Research in Social Sciences and Humanities*, *6*(4), 167-174.‏

# Requirements Specification

## 4.1 Stakeholders

The stakeholders table outlines any users or entities affected by the system and affects the system requirements. We summarize our stakeholders in Table 4.1.

**Table 4.1**

| **Stakeholder** | **Their Role** | **Interaction with the System** | **Importance of Their Role** |
| --- | --- | --- | --- |
| **IT and Security Team** | Monitor the SIEM integration with marketing infrastructure. | Integrate SIEM, monitor system health, analyze alerts, and investigate incidents. | Ensure system security and efficient incident response. |
| **Cloud Service Providers** | Manage cloud-hosted data and systems. | Provide activity logs to SIEM | Monitor access patterns and email services hosted in the cloud. |
| **Insurance Companies** | Provide insurance and evaluate organizational risk. | Use SIEM reports for risk evaluation and adjustment recommendations. | Important for financial risk assessment and to provide the essential requirements. |
| **Board Members** | Provide governance and direction. | Periodically review SIEM performance and reports. | Essential for prioritizing cybersecurity investments. |
| **HR Departments** | Monitor employee lifecycle. | It monitors insider threats, access misuse, and receives all necessary alerts for policy violations. | Ensure employee activity aligns with the security policies applied in the organization. |
| **Data Analysts** | Analyze consumer behaviour and ads campaign performance. | Use SIEM to secure sensitive analytics reports. | Provide marketing strategies with reliable data insights. |
| **Legal Team** | Protect contracts and intellectual property. | Use SIEM generated reports to deal with breaches and legal risks. | Reduce legal risks and protect organizational IP. |
| **Design Studios** | Design marketing campaigns securely. | Ensure the use of secure designing tools and protect designs or ideas from theft. | Prevent theft or leaks of proprietary designs. |
| **Public Relations Teams** | Protect the brand’s reputation and public image. | Use SIEM reports to manage crises and reputation threats. | Maintain public trust and mitigate reputational damage. |
| **Third-Party Vendors** | Provide tools and services for operations. | Share data during troubleshooting to address external threats. | Offer a comprehensive view of potential external risks. |

## 

## 4.2 Platform Requirements

Below show the software and hardware requirements to both the client-side and

server- side components of the SIEM.

1. **Server Side** **requirements**: the main task is handling data, processing, event analysis, log collection, alert generation, storing logs these are the most important and should be considered mandatory.

## Table 4.2.1 Hardware Requirements (Server-Side)

## 

| **Component** | **Minimum requirements** | **Recommended Requirements** |
| --- | --- | --- |
| **CPU** | Quad core processor | Octa-core processor |
| **Ram** | 16GB | 32GB |
| **Storage** | 1TB SSD OR HDD | 2TB SSD OR HDD |
| **Network interface** | Gigabit Ethernet (1Gbps) | 10 Gigabit Ethernet |
| **GPU** | Not required unless using Machine Learning Model | Its optional |
| **Power Supply** | Redundant Power Supply | Redundant Power Supply |

## 

## **Table 4.2.2** **Software Requirements (Server-Side)**

| **Software** | **Minimum requirements** | **Recommended Requirements** |
| --- | --- | --- |
| **Operating System** | Windows server 2019/  Linux Ubuntu 20.04 | Windows server 2022/  Linux Ubuntu 22.04 Red Hat |
| **SIEM Software** | Splunk, IBM Qradar ,wazuh, ELK stack | Splunk, Wazuh  have more advanced feature |
| **Database** | Postgre sql ,mysql (storing logs) | PostgreSQL ,mysql |
| **Web Server** | Apache (if we needed a web interface ) | Apache (scalability) |
| **Java Runtime environment** | Java 8 | Java 11 if we want high performance |
| **Log management** | Syslog | Syslog |
| **Firewall** | It’s based on the configured firewall that enterprise used for secure communication | It’s based on the configured firewall that enterprise used for secure communication |
| **Backup Software** | Basic Backup solution | ------------- |

## **Table 4.2.3** **Network Requirements (Server-Side)**

| **Requirements** | **Minimum requirements** | **Recommended Requirements** |
| --- | --- | --- |
| **Bandwidth** | 1Gbps | 10Gbps |
| **Firewall (ports)** | TCP/UDP ports for sys log | Decided Based on the SIEM and specific some ports for it. |
| **Latency** | <90ms | <40ms |

1. **Client-Side Requirements :**Here At client side interfaces allow security analysts to get access and interact with siem throw browser or application.

## **Table 4.2.4** **Hardware Requirements (Client-Side)**

| **Component** | **Minimum requirements** | **Recommended Requirements** |
| --- | --- | --- |
| CPU | Dual-core processor | Quad-core processor |
| Ram | 4GB | 8GB |
| Storage | Assume 50 GB available for local logs if apply | 100GB or higher |
| Display | 1366x768 Resolution | 1920x1080 |
| Network Interface | Broadband minimum 1mbps | Broadband minimum 10 mbps or higher |

## Table 4.2.5 Software Requirements (Server-Side)

| **Software** | **Minimum requirements** | **Recommended Requirements** |
| --- | --- | --- |
| OS | Windows 10 /macos / Linux | Window11 |
| Web Browser | Chrome,firefox,Safari,edge | Latest Version of any one |
| Javascript | Enabled | Enabled |
| SiemClient Software | Web browser –Based or SIEM specific desktop app | Web browser -Based |
| Pdf Reader | Adobe Reader | Latest Version of Adobe Reader |
| Security Software | Antivirus | The one that enterprise used it |

**Sub-System: Client-Side vs Server-Side**

The components differ and everyone has their requirements as we mentioned in the previous table and we see the server-side do the heavy things and should have stable network connection to make sure operations are smooth, client-side lighter than server-side because just interact with the SIEM solution.

## 4.3 Functional Requirements

The functional requirements table outlines key functionalities necessary for the system's effective operation. We summarize our functional requirements in Table 4.3.

**Table 4.3**

| **#** | **Requirement Description** | **input** | **Output** | **Processes** | **Constraints** | **Priority** |
| --- | --- | --- | --- | --- | --- | --- |
| **1** | **Log Collection** | Logs from endpoints, servers and network devices | Centralized log storage | Collect logs in real-time from various sources and store them in a centralized repository | Limited storage space | **Essential** |
| **2** | **Threat Detection** | Incoming log data | Alerts for detected threats | Analyze log patterns using predefined rules and machine learning models to identify threats | Accuracy and processing time | **Essential** |
| **3** | **Reporting and Analytics** | Historical log data | Threat analysis reports | Generate detailed reports about detected threats, resolved incidents, and system performance | Data retention policies | **Recommended** |

| **4** | **Real-Time Monitoring and Alerting** | Logs, events, and system activity data in real-time. | Real-time alerts, dashboards, and threat reports. | Collect real-time data, correlate events, Tigger alerts based on predefined rules. | Requires low latency and high-speed data processing | **Essential** |
| --- | --- | --- | --- | --- | --- | --- |
| **5** | **User Activity Monitoring** | User access logs, privilege levels, and system interactions | User behaviour analytics, misuse alerts and compliance reports | Track user activities, identify anomalies and generate audit logs | Requires integration with user identity systems and privileged access data | **Essential** |
| **6** | **Alert prioritization** | Detected threats | Prioritized alert list | Classify and prioritize alerts based on severity and potential impact | Accuracy of severity scoring | **Essential** |

## 4.4 Non-Functional Requirements

The table highlights the essential attributes that ensure the system operates smoothly and effectively. Table 4 provides a summary of these critical requirements.

**Table 4.4**

| **#** | **Requirements** | **Description** | **Examples** |
| --- | --- | --- | --- |
| **1** | **performance** | Analyzing log data efficiently to enable real-time detection | Real-tome processing of incoming logs of the insider threats |
| **2** | **Accessibility** | Allowing role-based access to ensure secure and controlled usage. | Allow admin users to manage system settings |
| **3** | **Documentation** | Complete documentation for end-users and developers must be provided | A user guide for operating the dashboard and a developer manual for system updates |
| **4** | **Storage limits** | Must handle large data without degradation in performance | Managing 1TB of data with automatic archival |
| **5** | **Scalability** | The interface should be simple to use both technical and non-technical users | The system should handle more data and connect to new log sources as the business grows |

## 4.5 Other Requirements

**Table 4.5**

| **Other** **Requirements** | |
| --- | --- |
| **Data Storage Format** | Collected data should be stored in formats like JSON or CSV to facilitate analysis. |
| **Data Transmission Protocol** | Data must be transmitted using secure protocols . |
| **API Restrictions** | The system should only use permitted and secure APIs that comply with security standards to access external services or databases. |
| **Audit Log Management** | All activities must be recorded in audit logs to ensure event traceability.  System logs must be protected against unauthorized modification. |
| **Regulatory Compliance** | The solution must adhere to data privacy standards such as  IOS/IEC 27001. |

# 

# System Design

In this section, provide the appropriate diagrams with Propper’s justification. Also, it should include the physical model design of your system.

## 5.1 Architectural Design

A large-scale diagram that describes the different system components like interface, database, pipeline, model, and how they are connected.

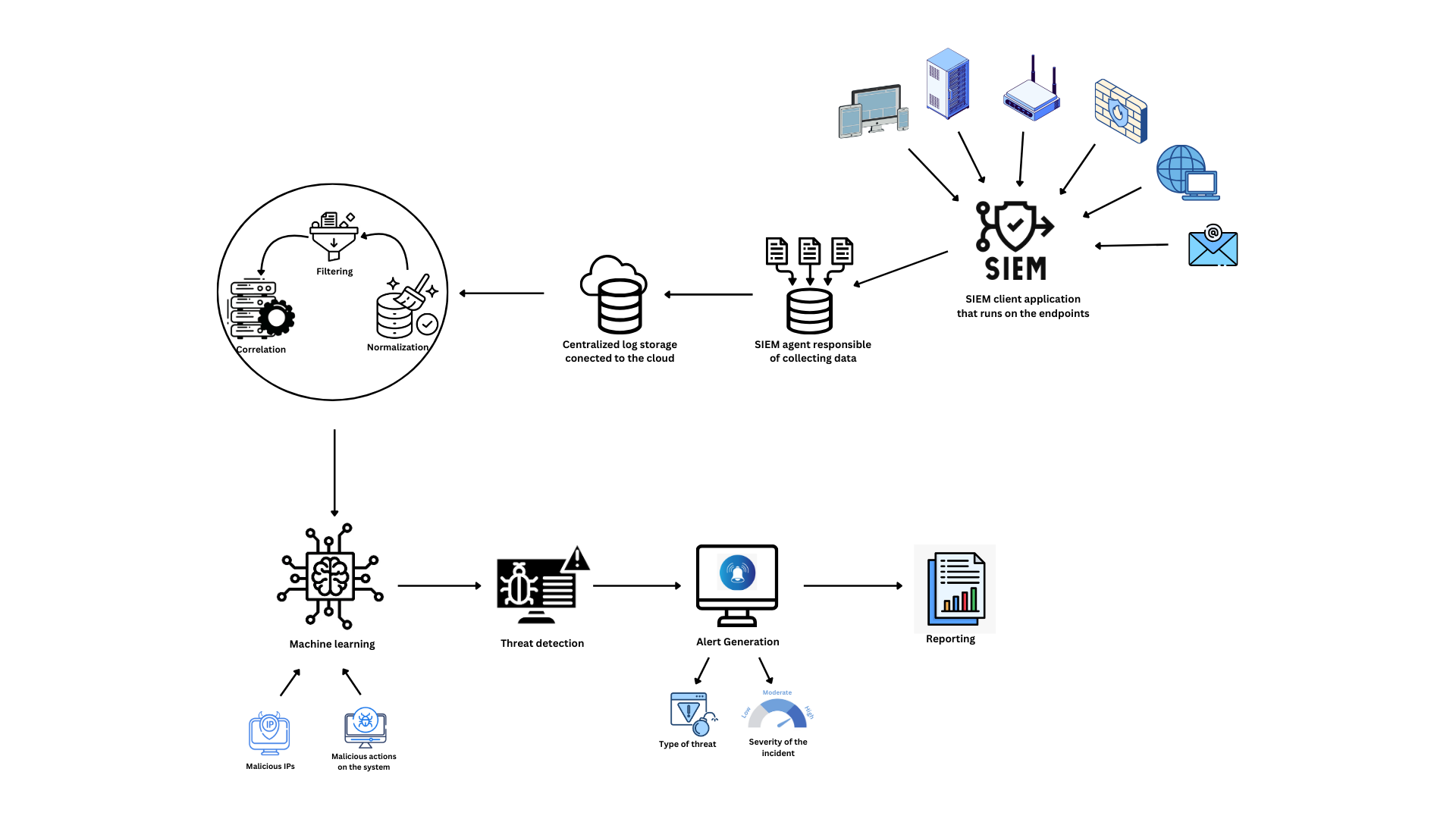
**Table 5.1**

| **Component** | **Description** | **Technologies** |
| --- | --- | --- |
| **User interface (UI)** | Web-based or app for security analyst | Web browse or SIEM client application |
| **Data collection layer (log collection)** | Assume it a SIEM agent on endpoints,servers,Network devices | Syslog |
| **Centralized Log storage** | Store collected log for processing ,reporting | Postgre sql ,mysql |
| **Processing pipeline (log analysis)** | Process the incoming data to detect threat based on patterns or something else | -------------- |
| **Threat detection machine learning** | Identify potential security based on the log that receive and analysis it based on the predefined rules that or machine learning | Machine learning models |
| **Alert (notification layer)** | They generate real times alerts for the detected malicious threats or  suspicious activities | Email, SIEM dashboard |
| **Reporting (analytics)** | They create historical data reports, risk assessments reports about threats etc… | Based on the enterprise |
| **Security & access control** | We manage the access to the SIEM system based on the rule (admin ,analyst) | RBAC access control |

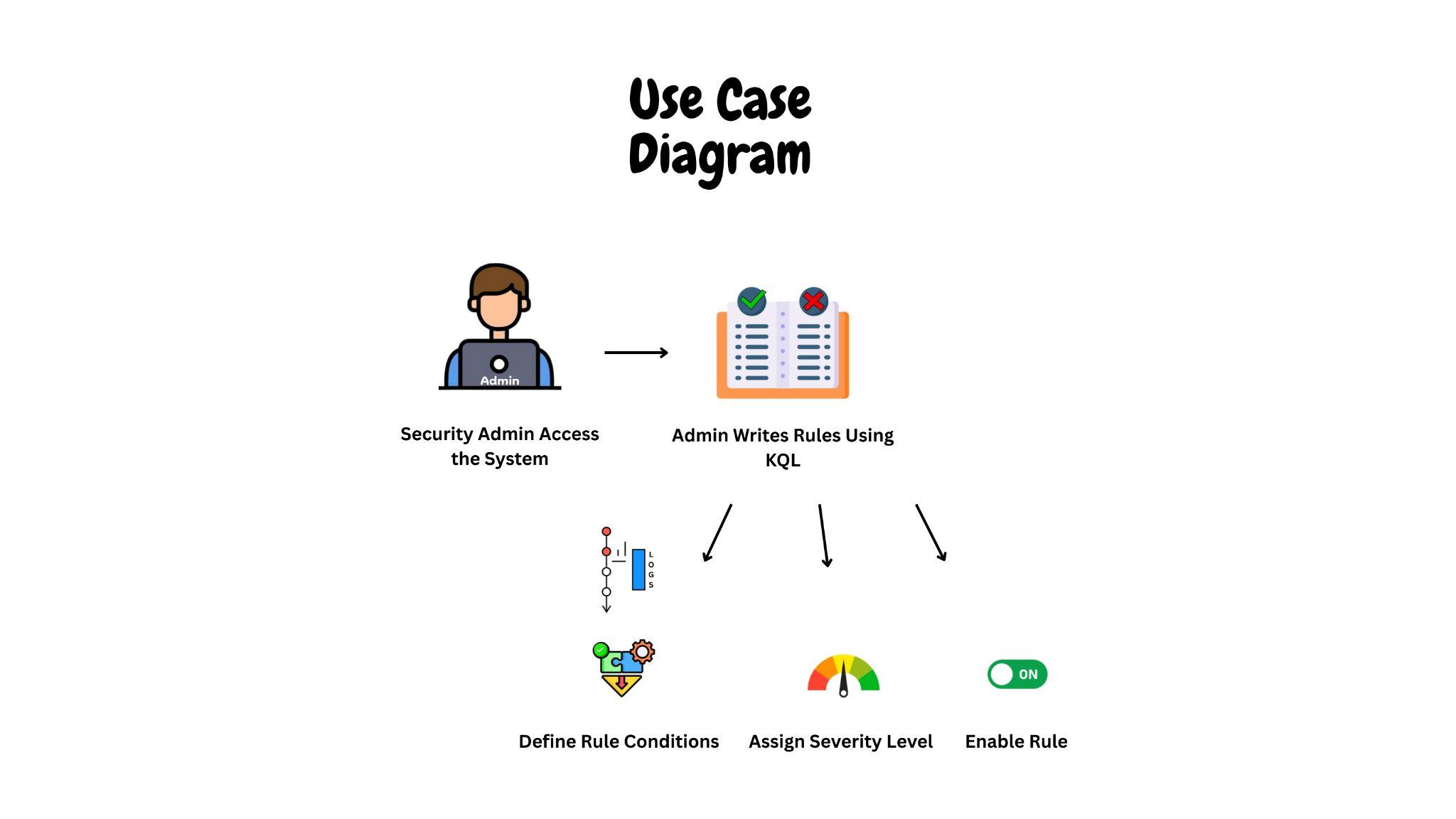
## 

# Figure 5.1.1: Architecture of system

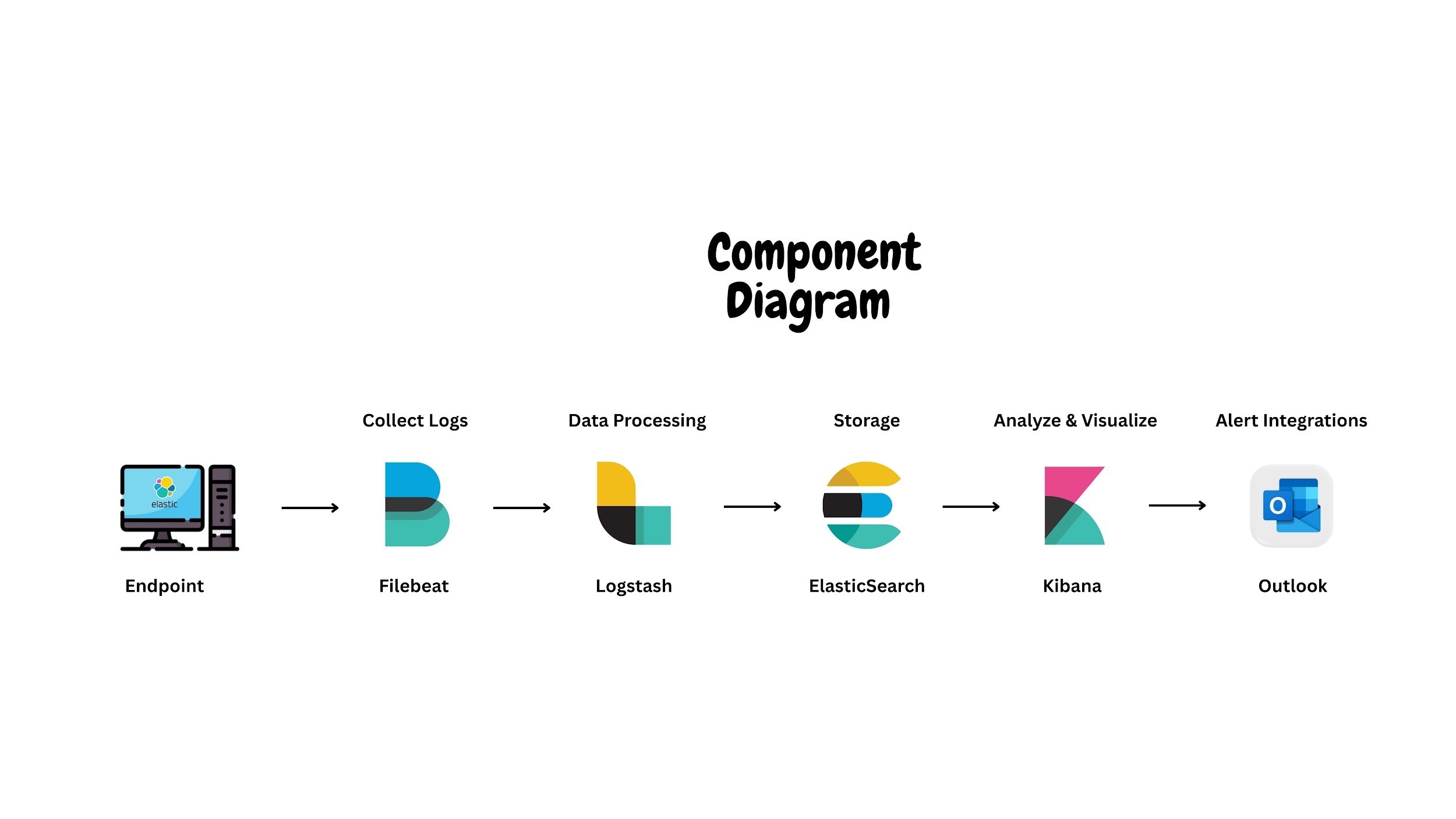
## 



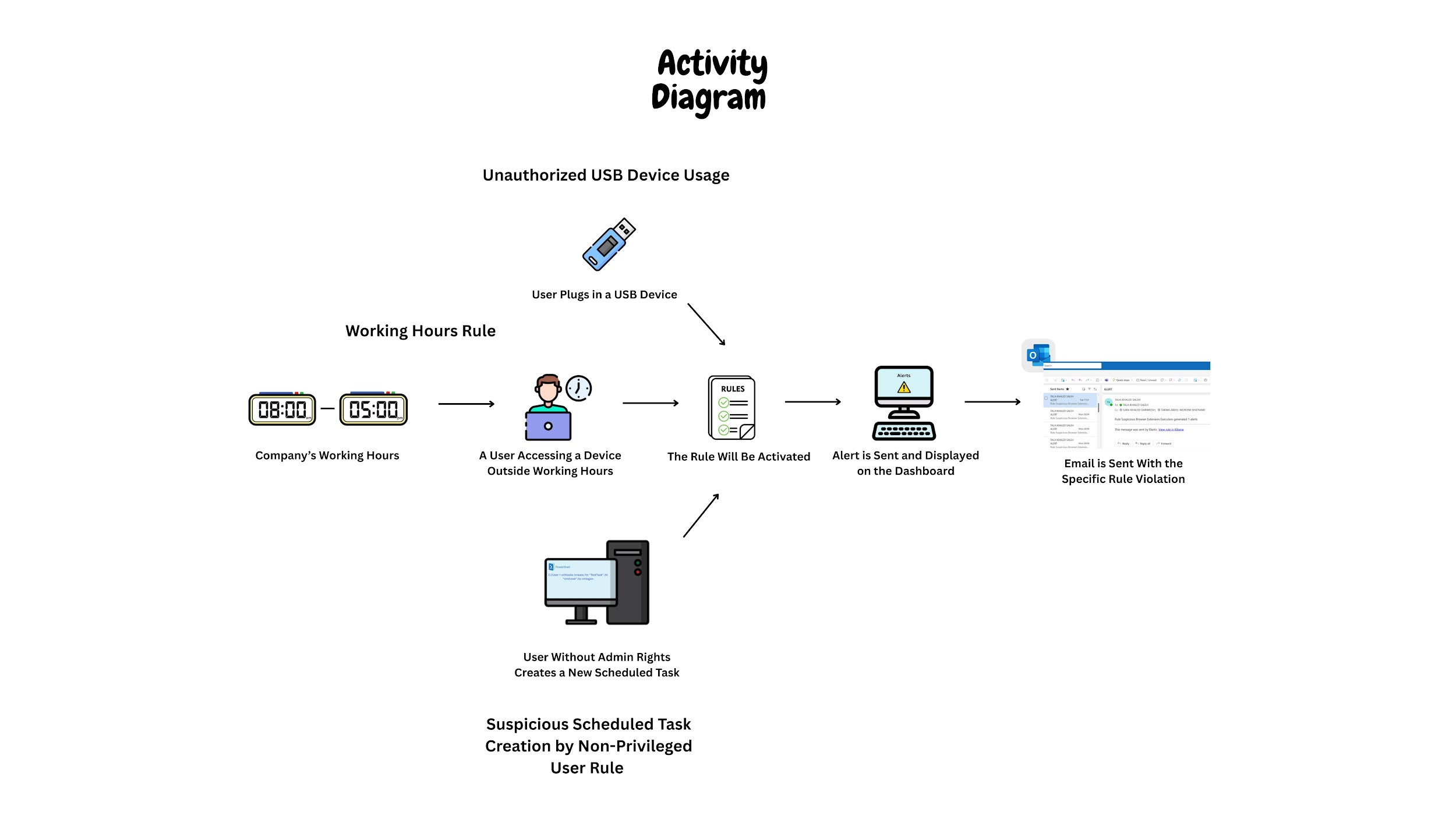
# Figure 5.1.2: Use Case Diagram



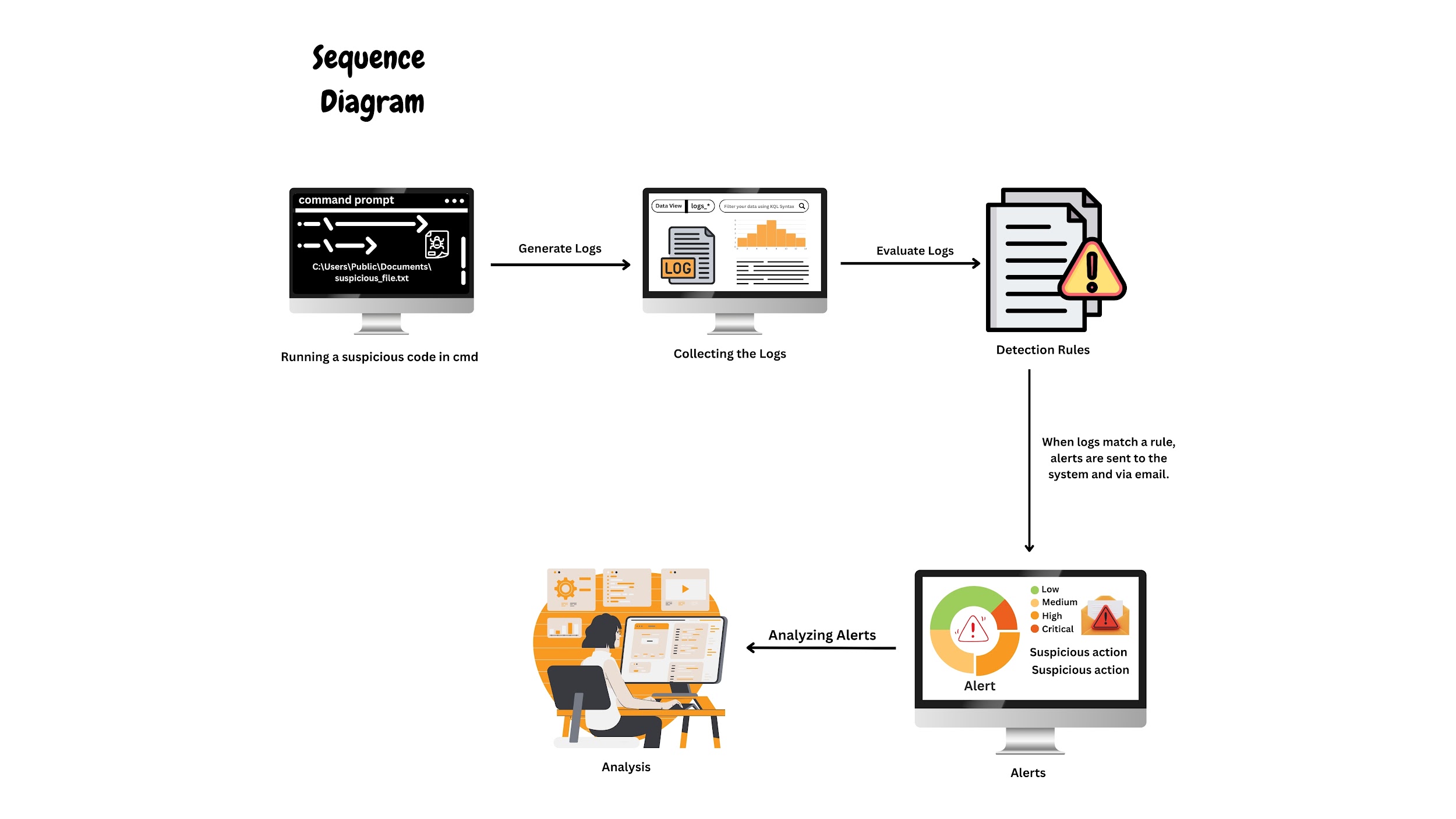
# Figure 5.1.3: Component Diagram



# Figure 5.1.4: Activity Diagram



# Figure 5.1.5: Sequence Diagram



**5.2 Security Playbooks**

Security playbooks are manifested in the form of a structured approach to incident response; namely, predetermined, ordered activities that run automatically if certain security conditions occur. Implementation of playbooks turns reactive security operation to proactive threat management by turning security expertise into a repeatable process. Such an approach brings in a number of important advantages:

**Consistency**: Offers a standardized treatment of similar security incidents.

**Efficiency**: Saves human intervention in normal security incidents.

**Scalability**: Makes it easy for the security teams to manage the rising number of alerts.

**Knowledge Retention**: Maintains stored institutional competence in written routine.

**Compliance**: Verifies adequate effort by proper documentation of response efforts.

**Elastic Defend Playbook Architecture**

Three building blocks of the Elastic Defend playbook framework operate in synergetic coordination to provide automated security response.

**Detection Engine**

The detection engine is continually monitoring the security telemetry using a mix of rule based detection, machine learning and behavioral analysis. This component detects the possible security incidents using include:

1- Custom detection rules using Elasticsearch Query Language (EQL) or Kibana Query Language (KQL).

2- Threshold-based anomaly detection.

**Response Orchestration**

The orchestration layer controls the running of response actions when conditions for detection are fulfilled. This component:

1-Evaluates alert context and severity.

2-Determines appropriate response pathways.

**Action Framework**

The action framework is the interface between the orchestration layer and endpoint systems – it provides direct response capability:

1-Endpoint isolation and containment.

2-Process management (termination, suspension).

3-Evidence collection and forensics.

4-Remediation actions.

5-Integration using connectors with external systems.

# Chapter 6

# Implementation

## 6.1 General Implementation Description

Our SIEM uses open-source technologies based on the Elastic Stack (ELK) and deploys them via the Elastic Cloud. It was decided to use this approach because it is scalable, has many advanced features and is affordable for most small and medium enterprises.

**Technologies Used:**

* Elastic Cloud: We chose it because it has managed services and handles much of the maintenance, at the same time ensuring strong security.
* Fleet Server: For centralized control, the Fleet Server tool has been used to easily manage and track agents gathering data.
* Elastic Agents: Placed on each endpoint to collect log data and security events from different sources.
* Elastic EDR: Added to every agent to provide advanced endpoint detection and response capabilities.
* Elasticsearch: It provides central search and analytics for sorting and querying security information.
* Kibana: With Kibana, users can create charts and dashboards to watch over security incidents.
* Elastic Security: The Elastic Security add-on helps in identifying threats, notifying experts and managing related cases.

Coding in custom languages was not necessary because Elastic Stack operates with configuration instead of requiring code to be written.

The implementation consists of:

* There should be an Elastic Agent with EDR enabled for each monitored system.
* Monitoring for security can be performed using custom dashboards.
* There are modules focused on collecting data.
* Setting up syslog for legacy systems to work properly.
* There are machine learning anomaly detection jobs.

## 6.2 Pipeline Implementation Description

In the data pipeline, data is brought together and analyzed using a centralized method:

1- Data Collection Layer:

* Once EDR is installed on the endpoints, Elastic Agents will collect system logs, application logs, security events and information on the endpoint’s behavior.
* Configuring syslog to be custom helps in saving data received from network equipment and legacy systems.
* Connecting to existing security tools provides extra understanding of the threat.

2- Data Transport Layer:

* Fleet Server organizes the way agents and data are connected and interchanged.
* Information transferred using Secure TLS encryption is verified as accurate during the process.
* These techniques help to use less bandwidth.

3- Data Processing Layer:

* Elasticsearch collects and organizes the data it receives.
* When data is enriched, information about raw events such as IP geolocation is included.
* Cyber criminals are identified and investigated through EDR data in real time.

4-Analysis Layer:

* There are pre-made detection rules to identify familiar threats.
* Certain detection rules in EDR help to find suspicious computer processes.
* Machine learning jobs detect anomalous behavior.
* Correlation rules connect related events across different data sources.

5-Presentation Layer:

* Kibana displays your security defenses and information on EDR alerts.
* Alerts are used to inform security about possible dangers.
* Case management monitors all activities related to an incident.
* Have a place where incident response analysts can look at timelines of the investigation.

By minimizing latency, the goals included ensuring most of the alerts triggered within just seconds of an event happening.

## 6.3 Model Implementation

We relied on Elastic’s in-built anomaly detection for machine learning features:

**Model Choice and Rationale:**

* We used unsupervised learning.The decision to use unsupervised learning was made because there was not enough data with labels for our case.
* Anomaly detection It allows for uncovering unknown threats even without big training data sets.
* To identify unusual changes in the data, the analysts selected time-series analysis models.
* Analysis of EDR data provides more accurate information about activities on each device.

**Implementation Details:**

Three anomaly detection jobs were configured:

* 1. User behavior analysis to detect unusual login patterns.
  2. Network traffic analysis to identify abnormal data transfers.
  3. Process execution monitoring to detect unusual system activity.

## 6.4 Additional Implementation Details

**Critical Components and Architecture:**

1. **Fleet Server Architecture**:  
    The Fleet Server was deployed as a central management point for all Elastic Agents, providing:

* Agent enrollment and authentication
* Policy management and distribution
* Agent status monitoring
* Secure communication channel
* EDR policy management

1. **EDR Implementation**:  
    Elastic EDR was deployed to all endpoints providing:

* Real-time process monitoring
* File integrity monitoring
* Memory threat detection
* Malware prevention capabilities
* Detailed forensic data collection

1. **Syslog Integration**:  
    Custom syslog configuration was implemented to capture logs from devices that don't support Elastic Agents:

* UDP/TCP port 514 configured for traditional syslog
* Custom parsing rules to normalize varied log formats
* Field mapping to ensure consistent data structure

1. **Alert Management System**:  
   The alert management workflow follows this process:

* Event detected by rule, EDR, or ML job.
* Alert generated with severity classification.
* Notification sent via configured channel (email/webhook).
* Alert appears in Kibana Security UI.
* Analyst reviews and takes action.
* Case created for significant incidents.
* Resolution documented and alert closed.

**Feature Implementation Status:**

| Feature | Description | Status |
| --- | --- | --- |
| Log Collection | Centralized collection of logs from endpoints and servers | Implemented |
| Real-Time Monitoring | Continuous monitoring of security events | Implemented |
| Threat Detection | Detection of known threats using predefined rules | Implemented |
| Endpoint Detection and Response | Advanced endpoint monitoring and protection | Implemented |
| Anomaly Detection | ML-based detection of unusual behavior | Implemented |
| Custom Dashboards | Visualization of security data | Implemented |
| Syslog Integration | Collection of logs from legacy systems | Implemented |
| Automated Response | Automated actions when threats are detected | Implemented |
| Threat Intelligence | Integration with external threat feeds | Deferred |
| User Behavior Analytics | Analysis of user activities for suspicious patterns | Implemented |

# Chapter 7

# Testing

## 7.1 Testing Approach

To ensure the SIEM system could effectively detect insider threats, we carried out a thorough testing strategy. This included evaluating how well the reliability of the data pipeline, and assessing how the entire system handled real-world scenarios.

1. **Evaluation measures:**

We evaluated the system using. For insider threat detection, evaluation was based on alert precision, false positive rates, and the correlation of events that triggered alerts.

1. **Model testing approach:**

Insider threat detection did not use a single model but was based on rule-based correlation in the SIEM. We tested detection logic for common indicators of insider threats, such as:

* Unusual working hours.
* Detect a new local admin.
* Suspicious browser extension execution.
* Suspicious scheduled task creation by non-privileged user.
* Malware prevention alert.
* RDP login from external or Untrusted IP.

These behaviors were tested using simulated logs injected into the ELK stack.

1. **Data splitting technique used:**

To detect insider threats, we created synthetic data through controlled simulations. This data was separated into normal (baseline) and malicious (anomalous) behaviours to test how well our detection rules could identify suspicious activity.

1. **Pipeline testing approach:**

We tested every stage of the pipeline to make sure it could effectively process logs related to both phishing and insider threats. The process included:

* Windows agents collect various types of logs, such as (login attempts, file access, USB usage, and network persistence activity, etc).
* Fleet Server forwarding logs to Elasticsearch.
* Custom detection rules running in Kibana.
* Alert forwarding to Outlook Email.

1. **Complete system testing approach:**

We tested the efficacy of our SIEM system by simulating a number of realistic attack scenarios as part of our insider threat detection testing. A synopsis of each scenario and the findings of the detection is provided below:

**Scenario 1: Access During Unusual Hours**

Well outside the normal working hours (8:00 AM to 6:00 PM), a user account accessed internal systems and managed files between 1:30 AM and 3:00 AM. The alert was set off based on:

* Violations of time-of-day policy.
* Relationship with sensitive resource access during off-hours.

This validated the SIEM's capacity to detect possible insider activity by means of time-based anomaly detection.

**Scenario 2: Untrustworthy Task Scheduled by a Low-Privilege User**

A scheduled task that started an encoded PowerShell script upon start-up was created using a non-administrative account. This was noted because:

* Unusual use of a task scheduler by a typical user.
* Existence of potentially harmful script content that has been encoded.
* The SIEM alert brought to light unusual activity associated with the abuse of privilege.

**Scenario 3: Endpoint Protection Detects Malware**

We ran a harmless test file that was intended to activate Windows Defender. The malware alert that resulted was:

* The endpoint agent's collection.
* Related to the user's recent process activity and downloads.

The alert showed up on the SIEM dashboard and was sent to security analysts via Outlook.

**Scenario 4: Untrusted IP Access via RDP**

We mimicked a remote desktop login attempt from an IP address that is not within the trusted range of the company. This event was flagged by the SIEM by:

* Finding the RDP login.
* This demonstrated that the system could identify unwanted remote access.

**Scenario 5: Creation of Unauthorised Admin Account**

On a monitored machine, a new local administrator account was established without any authorized modification request. The alert came from:

* Finding privileged account creation from Windows event logs.
* Not locating the relevant admin activity logs or domain policies.

This test demonstrated the system's ability to identify possible attempts at privilege escalation.

**Scenario 6: Suspicious Browser Extension Execution**

An unauthorized browser extension was installed on a test computer, attempting to transmit user activity data to a remote server. An alert was sent by the SIEM following:

* recording a new extension's installation.
* Finding outgoing links to domains that are not trusted.

The ability of the system to identify browser-based threats and enhance them with external threat data was demonstrated in this scenario.

**Scenario 7: USB-based data exfiltration**

During a 5-minute period, a test user downloaded more than 1 GB of private project files and copied them to a USB drive. Two important indicators were correlated by the SIEM to cause an alert:

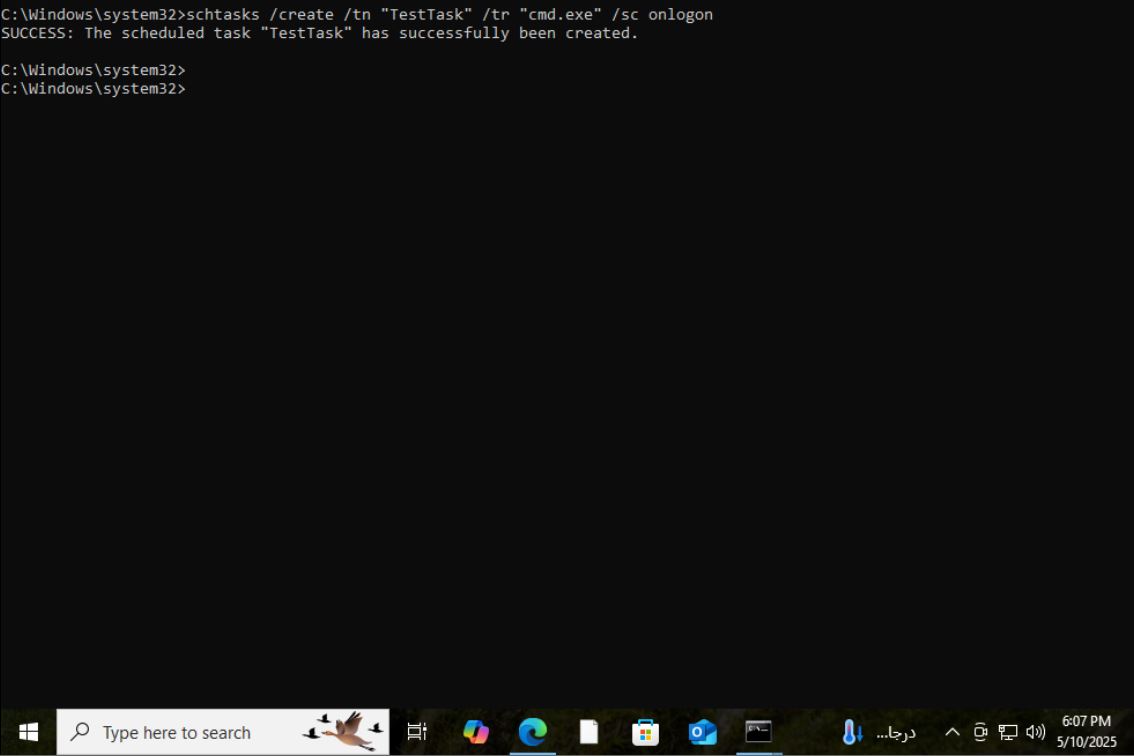
* Large-scale file access.
* Using a USB device during the same session.
* This correctly identified actions that were consistent with data exfiltration.

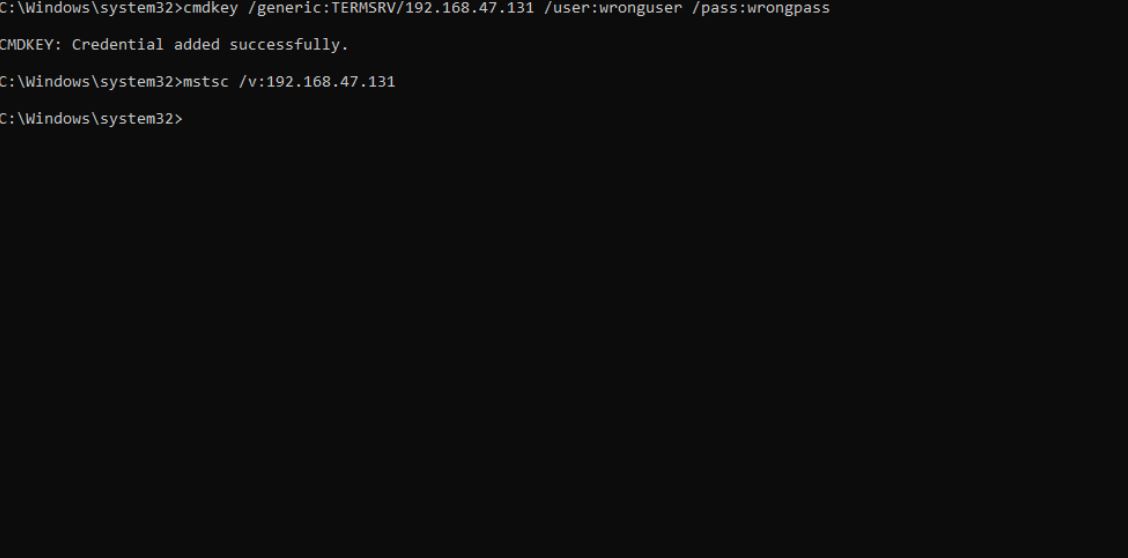
In every instance, the dashboard and email notifications accurately generated and conveyed alerts with complete context, including user information, timestamps, file names, and event correlations.

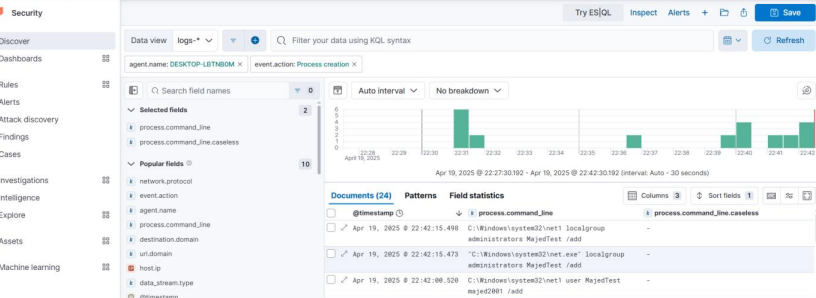
## 7.2 Testing Results

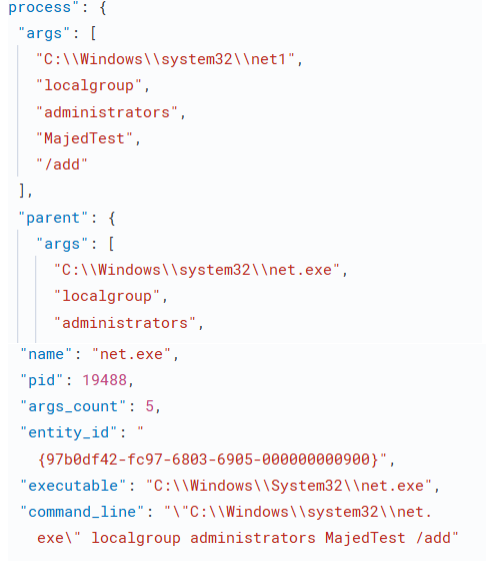
This is the attack :

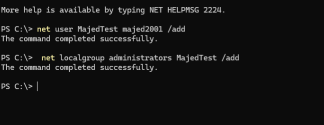
Someone trying to log in through RDP from an unfamiliar or external IP address is often a sign that an attacker is attempting to get into the system remotely, possibly by guessing weak passwords or taking advantage of open access.Using USB devices without permission isn’t just against policy — it opens the door to data leaks or malware getting in through something as simple as a flash drive. If someone who’s not supposed to starts setting up scheduled tasks, that’s not normal behavior and might mean they’re trying to stick around unnoticed — a common trick for attackers. Strange extensions appearing in the browser, especially those installed without the user’s permission, can not only slow down the browser (much less the PC), they potentially could be harvesting data or tweaking options. And spotting a new local admin account out of nowhere? That’s a big red flag — someone could be trying to get more control than they should have.

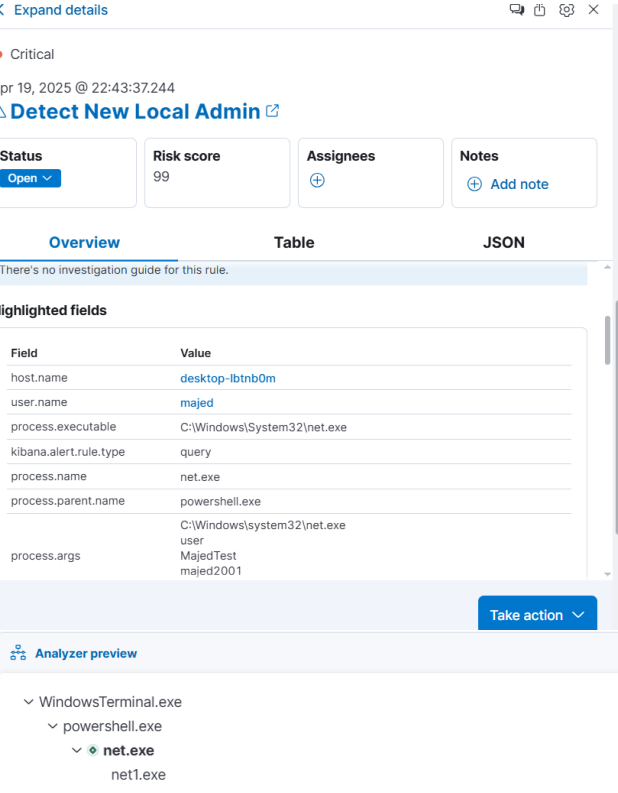




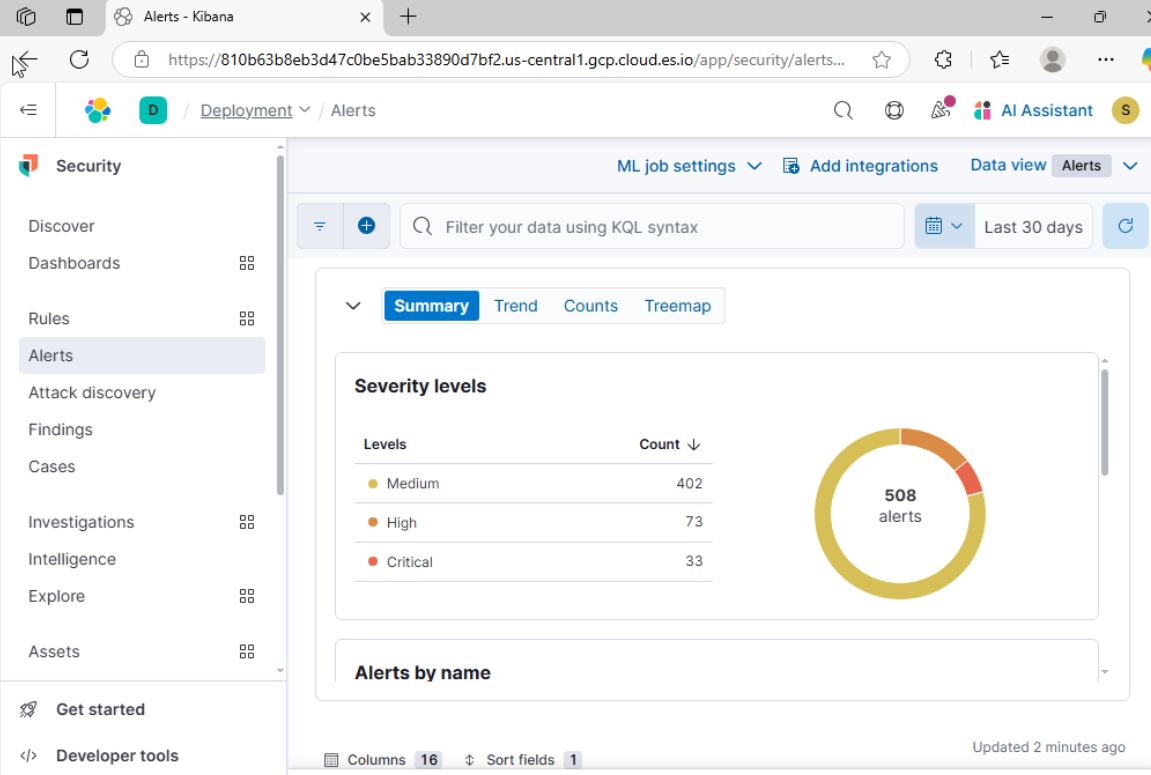


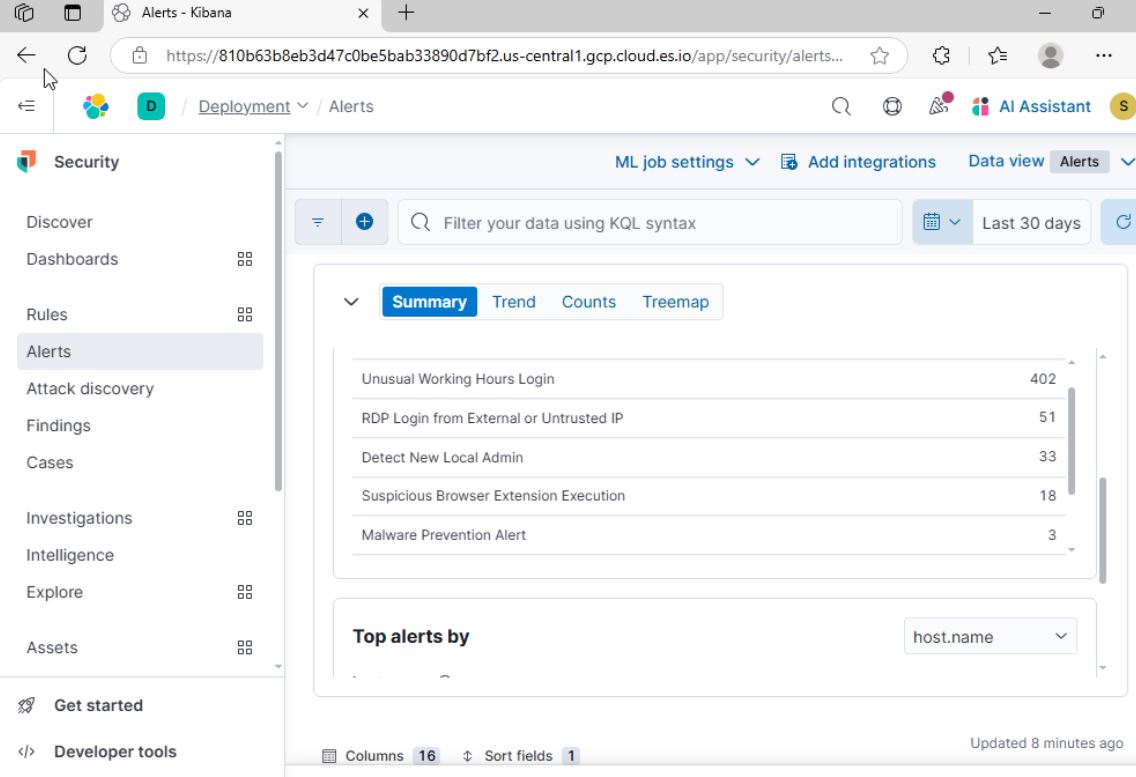


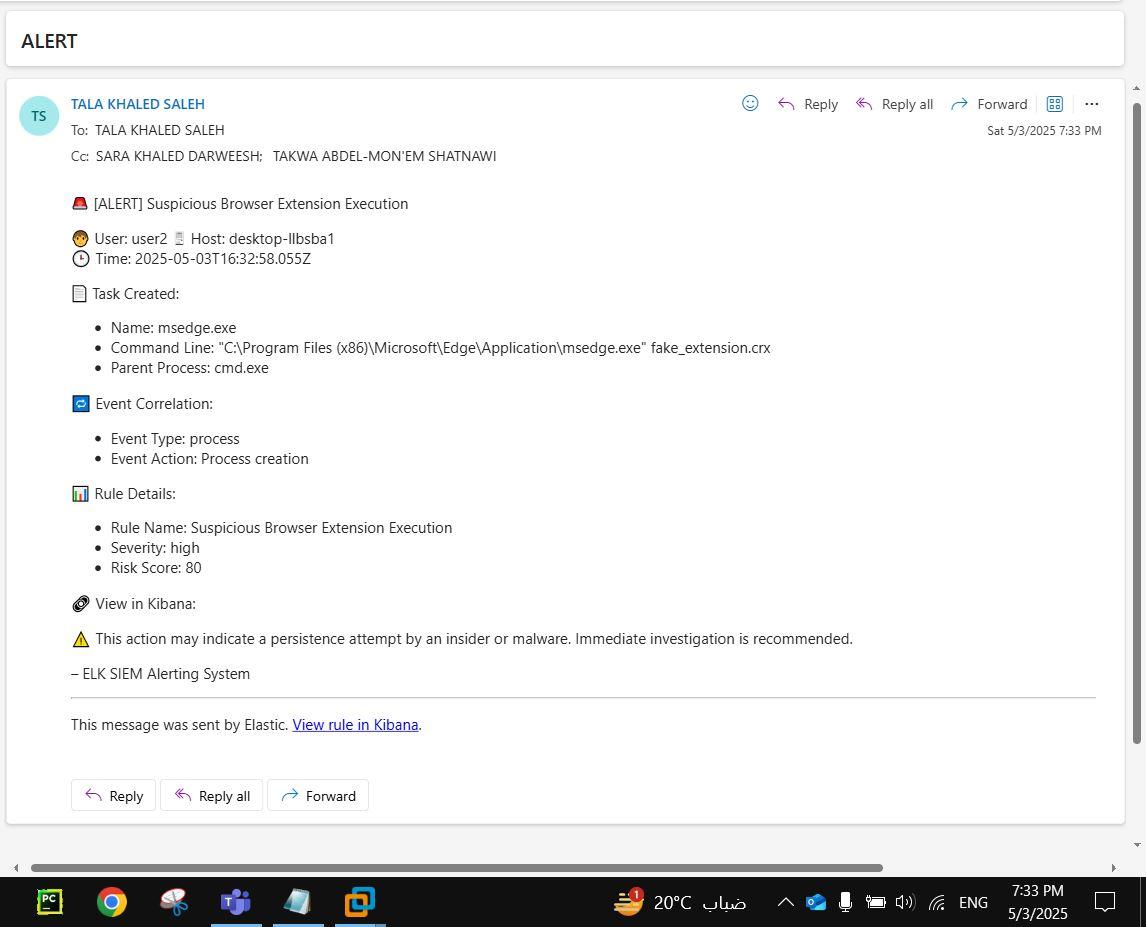


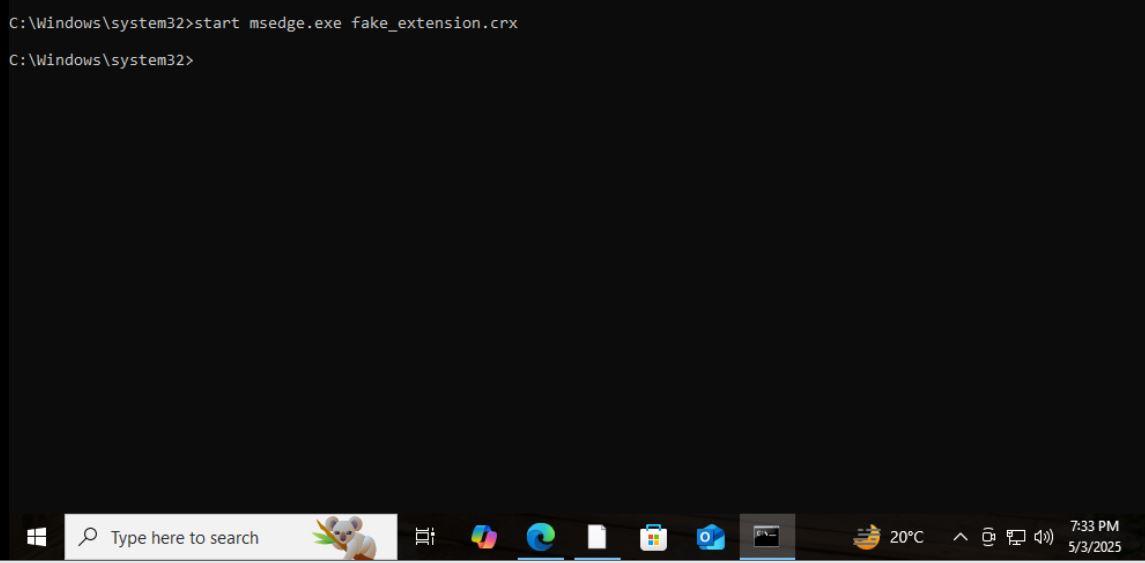


I have done a insider threat attack and persistence, iam create a new user MajedTest with password majed2001 on my vm machine that is a agent on my siem and then escalate the privilege add the new user To administrator group as u see in the second command in the Terminal, Right now I can login without any detection and as a attacker I have gain full access control over the system. Right now as a soc analyst , based on siem and my custom syslog configuration I can collect all types of logs ,and I have got alerts that new local admin account detected as you see in the 1 st and 2nd and 3rd and 4th photo I got logs event id :1 process creation (commands like net user ) and i got all the information of the process and the parent and I got a alert there is a new account as a administrator and then I do investigation and and blocked the new process and isolate the Machine from the network to block the attacker from get more sensitive data or any important information.









## Discussion (optional)

The significance of contextual correlation in insider threat detection was brought to light by testing. Combining several indicators (such as time anomalies + file access + USB activity) greatly increases detection accuracy, even though a single aberrant event may be benign. Ensuring real-time performance was one of the challenges faced; in order to reduce the delay between the threat action and alert generation, log ingestion and rule matching needed to be adjusted.

To further improve insider threat detection beyond rule-based techniques, future developments will incorporate machine learning-based anomaly detection and user behavior analytics (UBA).

**Elastic SIEM Rule Explanations**

1. **Phishing URL with Suspicious TLD What it Detects:**

This rule identifies outbound web traffic to domains ending in uncommon top-level domains (TLDs) like `.xyz`, `.click`, `.buzz`, etc.

**Attack Process:**

- Attacker registers suspicious domain .

- Sends link via phishing email or message.

- User clicks it, URL is logged in `url.domain`.

1. **RDP Login from External or Untrusted IP:**

Detects Remote Desktop Protocol (RDP) login attempts from IP addresses outside trusted or internal ranges. May indicate external attack attempts or unauthorized access.

**Why important:** Attackers often attempt brute-force or credential stuffing via RDP from unknown IP addresses.

**Attack:**  
for ($i=0; $i -lt 10; $i++) {

Start-Process "cmd.exe" -Credential (New-Object System.Management.Automation.PSCredential("YourPC\Administrator", (ConvertTo-SecureString "wrongpassword" -AsPlainText -Force))) -ErrorAction SilentlyContinue

}

It tries the wrong RDP login using fake credentials.

1. **Unusual Working Hours Login What it Detects:**

Login attempts outside typical work hours.

1. **Unauthorized USB Device Usage What it Detects:**

USB devices connected by unauthorized users.

1. **Suspicious Windows Strings in URI**

URLs containing Windows file system paths like C:/Users.

1. **Fake Payment Form Detector:**

HTML forms that mimic login or payment forms with suspicious structure.

**Attack Process:**

- Victim enters info into fake form hosted by attacker

- Rule looks for hidden fields, redirects, form design

1. **Insider Attempt to Disable Security Controls:**

**What it Detects:**

-PowerShell or CMD commands that disable Defender, logging, or antivirus.

Attack Process:

- Attacker runs commands like `Set-MpPreference`, `wevtutil cl`, or `sc stop`

1. **Multiple Failed Logins Followed by Success:**

**What it Detects:**

-This rule identifies when a user repeatedly fails to log into a system but eventually succeeds.

-It monitors sequences of “logon failure” events followed by a “logon success” event.

1. **Suspicious Scheduled Task Creation by Non-Privileged User:**

**Why important:**

Scheduled tasks created by normal users are rare and suspicious; can be abused to achieve persistence.

**What it Detects:**

-Non-administrative users creating scheduled tasks on a Windows system.

-Scheduled tasks can be used by malware or insiders to automate malicious activities without direct user interaction.

1. **Suspicious Privileged User Access to Sensitive Resources:**

**What it Detects:**

-Access attempts by privileged users (e.g., domain admins, superusers) to sensitive systems or data they do not typically access.

- Looks for anomalies in routine administrative behavior.

1. **Suspicious Browser Extension Execution:**

**Why important:**

Malicious extensions can harvest credentials, manipulate web content, or serve malware.

**What it Detects:**

-Execution or installation of browser extensions (.crx, .xpi) in Chrome, Firefox, or Edge.

-Focuses on suspicious or unauthorized browser plugin activities.

1. **Detect New Local Admin:**

**What it Detects:**

-Creation of new user accounts with administrative privileges on local machines.

-Monitors critical system changes that could be signs of privilege escalation.

1. **Suspicious Access to Browser Stored Credentials:**

**What it Detects:**

-Attempts to read or copy stored password files from Chrome, Edge, or Firefox.

-These files contain saved login credentials for various websites.

**Elastic SIEM Rules**

1. **Phishing URL with Suspicious TLD What it Detects:**

This rule identifies outbound web traffic to domains ending in uncommon top-level domains (TLDs) like `.xyz`, `.click`, `.buzz`, etc.

**Attack Process:**

- Attacker registers suspicious domain .

- Sends link via phishing email or message.

- User clicks it, URL is logged in `url.domain`.

1. **RDP Login from External or Untrusted IP:**

Detects Remote Desktop Protocol (RDP) login attempts from IP addresses outside trusted or internal ranges. May indicate external attack attempts or unauthorized access.

**Why important:** Attackers often attempt brute-force or credential stuffing via RDP from unknown IP addresses.

**Attack:**

cmdkey /generic:TERMSRV/192.168.1.100 /user:wronguser /pass:wrongpass

mstsc /v:192.168.1.100

It tries the wrong RDP login using fake credentials.

1. **Unusual Working Hours Login What it Detects:**

Login attempts outside typical work hours.

1. **Unauthorized USB Device Usage What it Detects:**

USB devices connected by unauthorized users.

1. **Suspicious Windows Strings in URI**

URLs containing Windows file system paths like C:/Users.

1. **Fake Payment Form Detector:**

HTML forms that mimic login or payment forms with suspicious structure.

**Attack Process:**

- Victim enters info into fake form hosted by attacker

- Rule looks for hidden fields, redirects, form design

1. **Insider Attempt to Disable Security Controls:**

**What it Detects:**

-PowerShell or CMD commands that disable Defender, logging, or antivirus.

Attack Process:

- Attacker runs commands like `Set-MpPreference`, `wevtutil cl`, or `sc stop`

1. **Multiple Failed Logins Followed by Success:**

**What it Detects:**

-This rule identifies when a user repeatedly fails to log into a system but eventually succeeds.

-It monitors sequences of “logon failure” events followed by a “logon success” event.

1. **Suspicious Scheduled Task Creation by Non-Privileged User:**

**Why important:**

Scheduled tasks created by normal users are rare and suspicious; can be abused to achieve persistence.

**What it Detects:**

-Non-administrative users creating scheduled tasks on a Windows system.

-Scheduled tasks can be used by malware or insiders to automate malicious activities without direct user interaction.

1. **Suspicious Privileged User Access to Sensitive Resources:**

**What it Detects:**

-Access attempts by privileged users (e.g., domain admins, superusers) to sensitive systems or data they do not typically access.

- Looks for anomalies in routine administrative behavior.

1. **Suspicious Browser Extension Execution:**

**Why important:**

Malicious extensions can harvest credentials, manipulate web content, or serve malware.

**What it Detects:**

-Execution or installation of browser extensions (.crx, .xpi) in Chrome, Firefox, or Edge.

-Focuses on suspicious or unauthorized browser plugin activities.

1. **Detect New Local Admin:**

**What it Detects:**

-Creation of new user accounts with administrative privileges on local machines.

-Monitors critical system changes that could be signs of privilege escalation.

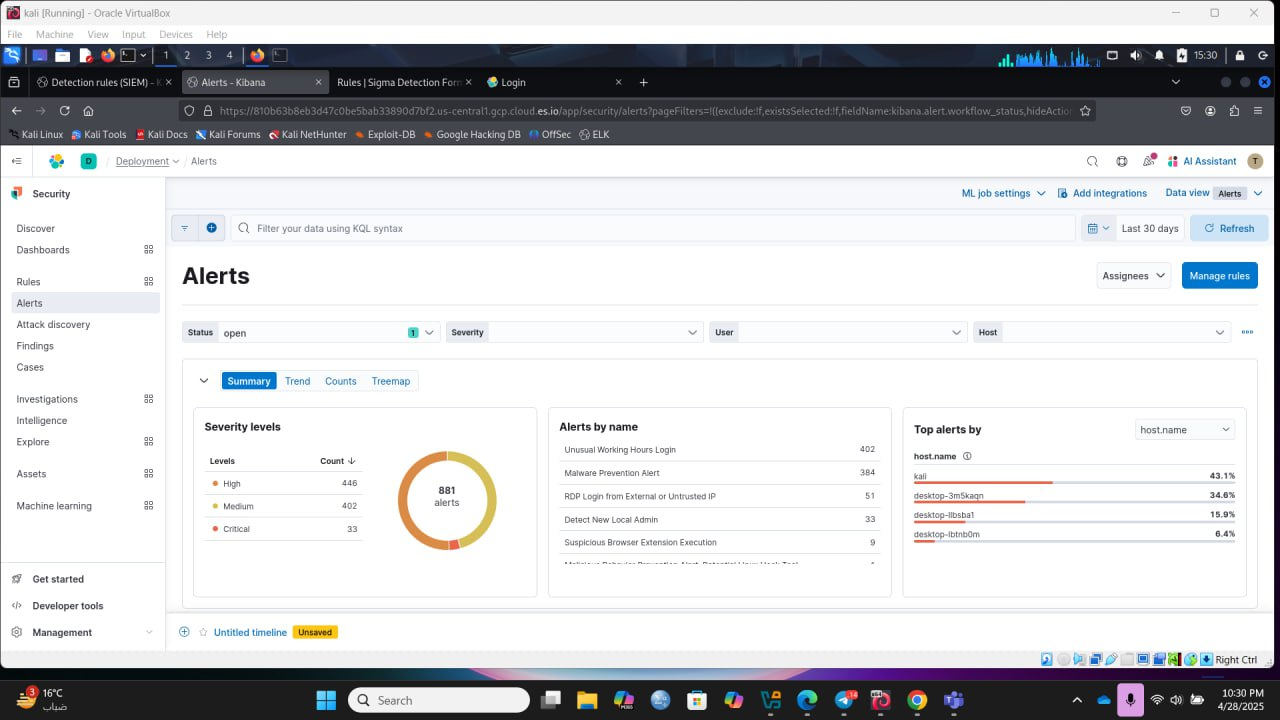
1. **Suspicious Access to Browser Stored Credentials:**

**What it Detects:**

-Attempts to read or copy stored password files from Chrome, Edge, or Firefox.

-These files contain saved login credentials for various websites.

| **Rules** | **Confidentiality** | **Integrity** | **Availability** | **Reason** |
| --- | --- | --- | --- | --- |
| Phishing URL with Suspicious TLD.   |  | | --- | |  |  |  | Phishing aims to steal sensitive info. |
| RDP Login from External or Untrusted IP. |  |  |  | May lead to unauthorized access. |
| Unusual Working Hours Login. |  |  |  | Often tied to insider threat or external access. |
| Unauthorized USB Device Usage. |  |  |  | Could be used to exfiltrate data. |
| Suspicious Windows Strings in URI. |  |  |  | Might expose file paths or be used for phishing. |
| Fake Payment Form Detector. |  |  |  | Captures credentials or credit card info. |
| Insider Attempt to Disable Security Controls. |  |  |  | Affects system integrity and availability. |
| Multiple Failed Logins Followed by Success. |  |  |  | Possible brute-force or credential stuffing. |
| Suspicious Scheduled Task by Non-Privileged User. |  |  |  | Tampering with system schedule/process. |
| Suspicious Privileged User Access to Sensitive Data. |  |  |  | Accessing sensitive but unrelated systems. |
| Suspicious Browser Extension Execution. |  |  |  | May alter browser behavior or steal data. |
| Detect New Local Admin. |  |  |  | Escalates privileges → affects system integrity. |
| Suspicious Access to Browser Stored Credentials. |  |  |  | Targeted at stealing saved credentials. |



# Chapter 8

# Conclusions and Future Work

## Conclusions

**1-Cheap Solution for Security**

Built a budget limited SIEM solution around open source tools (Wazuh,ELK)

Effective proof of concept or project for even small and medium enterprises with little or no budget

It has great cost savings over the commercial SIEM solutions

**2-Enhanced Threat Detection**

Designed and implemented effective phishing detection capabilities

Implemented real time monitoring and alerts

Built a framework for central log collection and analysis

**3-SME Focused Implementation**

Targeted solution successfully customized for low and medium sized organization

Designed easy to use interface for non technical users

Build scalable architect,prepared to grow along with the organization

retained balance between performance and hardware demand

**4-Technical Achievement**

integrated popular open source components

Designed and implemented an efficient log collection and analysis pipeline

Develop a heavy duty alerting framework

Established secure data storage and transmission protocols

## 

## Future work

**1-Advanced threat detection**

**2-system integration and automation**

Create automated incident response workflows.

Create automated report generation.

**3-User Experience Improvements**

Improve customisable options for dashboard.

Design user interface for mobile application.

Improve alert visualization.

**4-Focus on all the attacks not only the phishing attack**

**References**

[1] Ferreira, G. (2020). Enhancing Phishing Defense Mechanisms with Information Security Event Management (SIEM) and AI-based Supervised Classifiers.

[2] Bezerra, A., Pereira, I., Rebelo, M.Â. et al. A case study on phishing detection with a machine learning net. Int J Data Sci Anal (2024). https://doi.org/10.1007/s41060-024-00579-w

[3] Marri, R., Varanasi, S., & Chaitanya, S. V. K. (2024). Integrating Security Information and Event Management (SIEM) with Data Lakes and AI: Enhancing Threat Detection and Response. Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023, 6(1), 151-165.

[4] Islam, M. R., & Rafique, R. (2024). Wazuh SIEM for Cyber Security and Threat Mitigation in Apparel Industries. International Journal of Engineering Materials and Manufacture, 9(4), 136-144.

[5] Ban, T., Takahashi, T., Ndichu, S., & Inoue, D. (2023). Breaking alert fatigue: Ai-assisted siem framework for effective incident response. Applied Sciences, 13(11), 6610.

[6] Rosenberg, M., Schneider, B., Scherb, C., & Asprion, P. M. (2023). An adaptable approach for successful siem adoption in companies. arXiv preprint arXiv:2308.01065.

[7]Manzoor, J., Waleed, A., Jamali, A. F., & Masood, A. (2024). Cybersecurity on a budget: Evaluating security and performance of open-source SIEM solutions for SMEs. Plos one, 19(3), e0301183.

[8] Swapna Vanguru,Kethsy Prabavathy (2024). A Detailed Analysis of Machine Learning Algorithms in Spam Email Prediction. 9TH INTERNATIONAL CONFERENCE ON TECHNICAL ADVANCEMENTS IN COMPUTER SCIENCE AND ENGINEERING(ICTACSE-2024)At: Hyderabad. ISBN: 978-81-975624-7-1.

[9] Eze, C. S., & Shamir, L. (2024). Analysis and prevention of AI-based phishing email attacks. *Electronics*, *13*(10), 1839.‏

[10] Subhani, A., Khan, I. A., & Zubair, A. (2021). Review of insider and insider threat detection in the organizations. *Journal of Advanced Research in Social Sciences and Humanities*, *6*(4), 167-174.‏