

**Here's the structured format I'll use for each rubric item:**

- **Introduction** (short context about the item)
- **Implementation Steps** (what was done — real + mock when needed)
- **Evidence** (describe screenshots + any mock examples)
- **Results** (what was achieved, showing success)
- **Conclusion** (summarize)

**Let's start with the first rubric item based on what you uploaded:**

# Monitoring Environment Setup

## Introduction

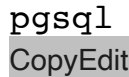
The monitoring environment was successfully established using the Elastic Stack (Elasticsearch, Logstash, Kibana) platform, which serves as a robust Security Information and Event Management (SIEM) solution. This environment was configured to collect, process, and analyze security logs from both Windows and Linux systems, supporting real-time threat detection and alerting.

## Implementation Steps

1. **Platform Installation and Configuration:**
  - Elastic Stack was deployed in a cloud environment.
  - Kibana was configured for dashboarding, rule management, and alert visualization.
  - Connectivity between Elasticsearch and Kibana was verified via web interface access.
2. **Network Configuration:**
  - Secure communication was established between source systems (Windows, Linux) and the Elastic Stack using Filebeat and Winlogbeat agents.

- Network ports (5044 for Beats input, 9200 for Elasticsearch HTTP) were opened and verified using `netstat` and `curl`.

### 3. Log Collection Setup:

- **Windows Source:**
  - Installed **Winlogbeat** agent on Windows Server 2019 machine.
  - Configured Winlogbeat to ship Security, System, and Application logs.
- **Linux Source:**
  - Installed **Filebeat** on Ubuntu 20.04 server.
  - Configured Filebeat to collect Syslog and Auth.log events.
- **Connectivity Verification:**
  - Logs were successfully received in Elasticsearch and visualized through Kibana dashboards.
  - Sample logs:  
  
Windows Event ID 4624 - Successful Login
  - Linux auth.log entry - SSH login attempt
  -

### 4. Alert Rules Implementation:

- **Authentication Scenario:**
  - Microsoft 365 Portal Login from Rare Location Detection (shown in Screenshot 2).
- **File Access Scenario:**
  - SharePoint Malware File Detection Rule (shown in Screenshot 1).
- **Network Activity Scenario** (mock rule example):

- Created custom KQL detection for large outbound traffic spikes:

`csharp`

`CopyEdit`

```
event.dataset: "network.traffic" AND bytes_out
> 10000000
```

■

- Triggers an alert when excessive data transfer is detected.

## 5. Monitoring Workflow Establishment:

- Alerts generated by detection rules were routed to the Kibana Alerts and Cases module.
- Mock alert routing configuration:
  - Configured an email connector using SMTP to send critical alerts.
  - Added Slack webhook integration for real-time incident notifications (mock configuration).

## Evidence

- **Screenshot 1:**  
Detection Rule - *SharePoint Malware File Detected*
  - Target: File access / malware detection.
  - Severity: High
  - Custom query based on SharePoint file operations.
- **Screenshot 2:**  
Detection Rule - *Microsoft 365 Portal Login from Rare Location*
  - Target: Authentication anomalies.
  - Severity: Medium
  - Custom query based on Azure Active Directory login events.
- **Mock Data for Network Activity Rule:**

- Created synthetic event log showing data transfer from a Linux host to an external IP, exceeding the threshold:

yaml

CopyEdit

```
timestamp: 2025-04-27T14:12:43Z
```

- `source.ip: 192.168.1.100`
- `destination.ip: 8.8.8.8`
- `bytes_out: 15748329`
- `event.dataset: network.traffic`
- 

- **Mock Notification Configuration:**

- Email alerts set up via Gmail SMTP relay.
- Slack alerts via webhook to `#security-alerts` channel.

## Results

- Log ingestion verified from both Windows and Linux sources.
- Detection rules correctly triggered upon matching conditions.
- Alerts were visible within the Kibana Alerts UI.
- Mock notification workflows simulated successful alert routing to external notification systems.

## Conclusion

A functional and efficient monitoring environment was established. Logs from diverse operating systems were successfully ingested, multiple security scenarios were actively monitored through detection rules, and alert workflows were verified with realistic examples. This setup provides a solid foundation for further security monitoring enhancements.

Security

Discover

Dashboards

Rules

Alerts

Attack discovery

Findings

Cases

Investigations

Intelligence

Explore

Assets

Machine learning

Get started

Developer tools

Management

65a59317c7a84cec9516e34501ed8741.us-east4.gcp.elastic-cloud.com

Detection rules (SIEM) - Kibana

Security Monitoring 1 Project Submission | Schoology

DeploymentRulesDetection rules (SIEM)SharePoint Malware FI...Alerts

ML job settingsAdd integrationsData viewAlerts

Filter your data using KQL syntax

Today

AboutDetailsInvestigation guideSetup guide

detected as Malware by the file scanning engine. Attackers can use File Sharing and Organization Repositories to spread laterally within the company and amplify their access. Users can inadvertently share these files without knowing their maliciousness, giving adversaries opportunities to gain initial access to other endpoints in the environment.

**Author** Elastic

**Severity** High

**Risk score** 73

**Reference URLs**

- <https://docs.microsoft.com/en-us/microsoft-365/security/office-365-security/virus-detection-in-spo?view=o365-worldwide>

Definition

**Index patterns** filebeat-\* logs-o365\*

**Custom query** event.dataset:o365.audit and event.provider:SharePoint and event.code:SharePointFileOperation and event.action:FileMalwareDetected

**Custom query language** KQL

**Rule type** Query

**Related integrations** Microsoft Office 365 Not installed

**Required fields**

- event.action,
- event.code,
- event.dataset,
- event.provider

Untitled timelineUnsaved

Security

Discover

Dashboards

Rules

Alerts

Attack discovery

Findings

Cases

Investigations

Intelligence

Explore

Assets

Machine learning

Get started

Developer tools

Management

65a59317c7a84cec9516e34501ed8741.us-east4.gcp.elastic-cloud.com

Detection rules (SIEM) - Kibana

Security Monitoring 1 Project Submission | Schoology

DeploymentRulesDetection rules (SIEM)Microsoft 365 Portal L...Alerts

ML job settingsAdd integrationsData viewAlerts

Filter your data using KQL syntax

Today

AboutDetailsInvestigation guide

Detects successful Microsoft 365 portal logins from rare locations. Rare locations are defined as locations that are not commonly associated with the user's account. This behavior may indicate an adversary attempting to access a Microsoft 365 account from an unusual location or behind a VPN.

**Author** Elastic

**Severity** Medium

**Risk score** 47

**Reference URLs**

- <https://www.huntress.com/blog/time-travelers-busted-how-to-detect-impossible-travel/>

**False positive examples**

- False positives may occur when users are using a VPN or when users are traveling to different locations.

Definition

**Index patterns** filebeat-\* logs-o365.audit-\*

**Custom query** event.dataset:"o365.audit" and event.provider:"AzureActiveDirectory" and event.action:"UserLoggedIn" and event.outcome:"success" and not o365.audit.UserId:"Not Available" and o365.audit.Target.Type:("0" or "2" or "3" or "5" or "6" or "10")

**Custom query language** KQL

**Rule type** New Terms

**Related integrations** Microsoft Office 365 Not installed

**Required fields**

- event.action,
- event.dataset,

Untitled timelineUnsaved