

# **Security Operations and Monitoring**

Study and Analysis on SOC and Implementation with  
Wazuh

Aayush Dhakal

Bachelors in Computer Networking and IT Security  
Islington College, Nepal

February 2026

# Table of Contents

|       |                                       |    |
|-------|---------------------------------------|----|
| 1     | Introduction to Logs.....             | 1  |
| 1.1   | Overview .....                        | 1  |
| 1.2   | Types of logs.....                    | 1  |
| 1.3   | Sources of Logs .....                 | 2  |
| 1.4   | Log Analysis.....                     | 2  |
| 1.4.1 | Log Collection .....                  | 2  |
| 1.4.2 | Log Management.....                   | 2  |
| 1.4.3 | Log Centralization.....               | 3  |
| 1.4.4 | Log Storage .....                     | 3  |
| 1.4.5 | Log Retention .....                   | 3  |
| 1.4.6 | Log Deletion.....                     | 3  |
| 1.4.7 | Log Analysis Process.....             | 3  |
| 2     | Introduction to SIEM.....             | 5  |
| 2.1   | Overview .....                        | 5  |
| 2.2   | Features of SIEM .....                | 5  |
| 2.3   | SIEM Architecture .....               | 6  |
| 3     | Introduction to SOC.....              | 8  |
| 3.1   | Overview .....                        | 8  |
| 3.2   | Functions of SOC.....                 | 8  |
| 3.3   | Pillars of SOC .....                  | 9  |
| 4     | Endpoints Monitoring with Wazuh ..... | 10 |
| 4.1   | Overview .....                        | 10 |
| 4.2   | Features of Wazuh.....                | 10 |
| 4.3   | Agents in Wazuh.....                  | 10 |
| 4.4   | Lab Setup.....                        | 10 |
| 4.5   | File Integrity Monitoring.....        | 12 |

|      |                                      |    |
|------|--------------------------------------|----|
| 4.6  | Configuration Assessment .....       | 14 |
| 4.7  | Vulnerabilities Detection.....       | 16 |
| 4.8  | Threat Hunting .....                 | 17 |
| 4.9  | Active Response - Firewall Drop..... | 20 |
| 4.10 | Malware Detection .....              | 22 |
| 4.11 | Alerting.....                        | 25 |
| 5    | Incident Handling .....              | 27 |
| 5.1  | Overview .....                       | 27 |
| 5.2  | Incident Response .....              | 27 |
| 5.3  | Incident Response Lifecycle .....    | 28 |
| 5.4  | Phases of Incident Response .....    | 28 |
| 6    | Conclusion .....                     | 31 |
| 7    | References.....                      | 32 |
| 8    | Appendix .....                       | 33 |
| 8.1  | Sysmon Integration in Windows.....   | 33 |
| 8.2  | File Integrity Monitoring.....       | 34 |
| 8.3  | Virustotal Integration .....         | 34 |
| 8.4  | Firewall Drop.....                   | 35 |
| 8.5  | Email Alerts .....                   | 35 |
| 8.6  | EICAR execution.....                 | 37 |
| 8.7  | SSH Brute-force using Hydra.....     | 38 |

## Table of Tables

|  |    |
|--|----|
| Figure 1 SIEM Architecture .....   | 6  |
| Figure 2 Endpoints Monitoring with Wazuh .....   | 11 |
| Figure 3 Registry key logs in Wazuh .....  | 12 |
| Figure 4 FIM logs .....  | 13 |
| Figure 5 Configuration Assessment in Kali machine .....  | 14 |
| Figure 6 Configuration Assessment report .....   | 15 |
| Figure 7 Vulnerabilities in Ubuntu machine .....   | 16 |
| Figure 8 Information about vulnerability in NIST website (National Institute of Standards and Technology (NIST), 2022) ..... | 17 |
| Figure 9 Threat Hunting dashboard.....   | 17 |
| Figure 10 Threat Hunting events.....   | 18 |
| Figure 11 SSH Brute-force from Kali towards Ubuntu machine.....  | 18 |
| Figure 12 SSH Brute-force result logs.....   | 19 |
| Figure 13 Threat Hunting - Event detail 1.....   | 19 |
| Figure 14 Threat Hunting - Event detail 2.....   | 20 |
| Figure 15 Rules from SSH Brute-force.....  | 20 |
| Figure 16 SSH Brute-forcing from Kali on Ubuntu machine .....  | 21 |
| Figure 17 SSH Brute-forcing result logs .....  | 21 |
| Figure 18 Malware Detection dashboard for Ubuntu machine .....   | 22 |
| Figure 19 Malware Detection dashboard in Ubuntu machine .....  | 23 |
| Figure 20 Malware Detection log in Ubuntu machine .....  | 23 |
| Figure 21 Malware detail I .....   | 24 |
| Figure 22 Malware detail II .....  | 24 |
| Figure 23 SSH Brute-force from Kali against Ubuntu .....   | 25 |
| Figure 24 Email alert .....  | 26 |
| Figure 25 Incident Response Lifecycle (National Institute of Standards and Technology (NIST), 2025) .....                    | 28 |

## Table of Tables

|                           |    |
|---------------------------|----|
| Table 1 Lab Setup .....   | 11 |
| Table 2 Rule levels ..... | 13 |

## 1 Introduction to Logs

### 1.1 Overview

Logs are the records of events in computer or network. Any activity on the computer, such as opening a file or even inserting charging cable is stored as log. Logs are basically digital footprints.

Logs can answer these questions about an event in a system:

- What happened?
- When did it happen?
- Where did it happen?
- Who is responsible?
- Were they successful?
- What is the result of their action?

### 1.2 Types of logs

1. Application logs: They are generated by applications. Example, an error indicating app crashing.
2. Security logs: They are the logs related to security events. Example, failed login, permission changes, attempting to execute out of privilege action, firewall actions, etc.
3. System logs: They are the logs that are generated by Operating System. Example, not having access to storage or network by OS, Kernel activities, boots sequences
4. Setup logs: They are the logs that are generated by installation or updates of programs or applications.
5. Network logs: They are the logs that are generated from network traffic.
6. Server logs: They are the logs that are generated from servers.
7. Database logs: They are the logs that are generated from the activities within database. Example, database queries and updates.
8. Linux logs: They are the logs in Linux system. In Linux, logs are in /var/log/ directory.

### 1.3 Sources of Logs

1. Network devices: Router, Switch, Load Balancer, Proxy, etc.
2. Authentication system: AD domain control, AAA server, etc.
3. Vulnerability scan results
4. Web servers
5. DNS servers
6. Memory dumps
7. Mobile devices
8. Metadata (files, web requests and replies, email messages, etc.)

### 1.4 Log Analysis

#### 1.4.1 Log Collection

Log collection refers to collecting or aggregating logs from various sources. It is important to maintain an accurate sequence of logs based on timestamp. So, Network Time Protocol (NTP) is used to synchronize and organize logs in order of time.

Steps for log collection:

1. Identify log sources
2. Choose a log collector tool or software
3. Configure collection parameters (NTP, different settings)
4. Test collection (to ensure logs are appropriately collected)

#### 1.4.2 Log Management

Log management refers to gathering and organizing logs securely.

Steps for log management:

1. Storage
2. Organization: Classify logs based on source or other criteria.
3. Backup
4. Review

### 1.4.3 Log Centralization

Log centralization refers to collecting and managing logs from different sources in a centralized environment, system or tool. It helps with efficient log management.

Steps for log centralization:

1. Choose a centralized system (such as Splunk or Elastic Stack)
2. Integrate sources (collect all log sources to centralized system)
3. Setup monitoring
4. Integrate incident management

### 1.4.4 Log Storage

Logs can be stored in local system, centralized repository or cloud. The choices for log storage depend on security requirements, accessibility needs, storage capacity, cost, compliance regulations, retention policies and disaster recovery plans.

### 1.4.5 Log Retention

Log storage is not infinite so retaining important logs for future is necessary. Log storage can be classified into different categories.

- Hot storage – It stores logs from past 3 to 6 months.
- Warm storage – It stores logs from past 6 months to 2 years.
- Cold storage – It stores logs from past 2 to 5 years.

### 1.4.6 Log Deletion

Since log storage is finite, logs need to be deleted after certain time based on log retention policy. Backup of logs is necessary before deletion.

### 1.4.7 Log Analysis Process

1. Parsing: It refers to breaking down log data into manageable and understandable components.
2. Normalization: Normalization brings various log data into standard format. Different sources can generate different log formats, so it is necessary to standardize logs.

## Security Operations and Monitoring

3. Sorting: Logs can be sorted by time, pattern, event type, severity or other parameters.
4. Classification: It refers to categorizing logs based on characteristics.
5. Enrichment: It refers to adding information like geographical data, user details, threat intelligence, etc. to make logs more meaningful.
6. Correlation: It refers to identifying connections between log records. It helps to detect patterns, trends and anomalies.
7. Visualization: It means representing logs in charts, graphs or heat maps.
8. Reporting: Reporting summarizes log data into structured format providing insights.

## 2 Introduction to SIEM

### 2.1 Overview

SIEM (Security Information and Event Management) is the combination of SIM and SEM. SIM (Security Information Management) is the collection and storage of different events generated from different areas such as devices, networks and applications. It is the historical collection of events. SEM (Security Event Management) identifies such events in real time using certain tools. A SIEM is the combination of SIM and SEM that provides real-time monitoring of events and a centralized place to view such events from different areas. In addition to that, there are different features of SIEM.

### 2.2 Features of SIEM

1. Real-time monitoring: SIEM offers real-time monitoring of events on devices, networks and applications. This is useful for threat detection on these attack surfaces.
2. Incident reporting and analysis: SIEM provides reports on any security incidents occurred which are helpful in forensics. SIEM itself can also be used to analyze those incidents.
3. Threat Intelligence: Feeding difference Indicators of Compromise into SIEM tools is helpful for threat intelligence. It can detect suspicious behavior and help the cybersecurity team to understand the adversary tactics and set up proactive defense.
4. Threat hunting: In addition to threat intelligence, the proactive defense setup is also helpful in threat hunting. This means the defense will be set up against the threats outside the scope of the SIEM alerts.
5. Attack simulations and assessment: The red teamers or penetration testers can simulate attacks using SIEM to assess the organization defense.
6. Insiders' detection: SIEM can be used for detection and investigation of different suspicious activities, which can be used to detect insider activities.
7. Vulnerability Management: SIEM is useful for vulnerability scanning on attack surfaces and identify the vulnerabilities in them. The security team can utilize it to find solutions and patches for security weaknesses in the system.

8. Compliance and Regulations: SIEM can be useful to meet the compliance. SIEM tools can produce reports regarding the compliance (GDPR, PCI DSS, etc.) which is helpful for auditors. (Knerler, Parker, & Zimmerman, 2022) (Fortinet, n.d.)

### 2.3 SIEM Architecture

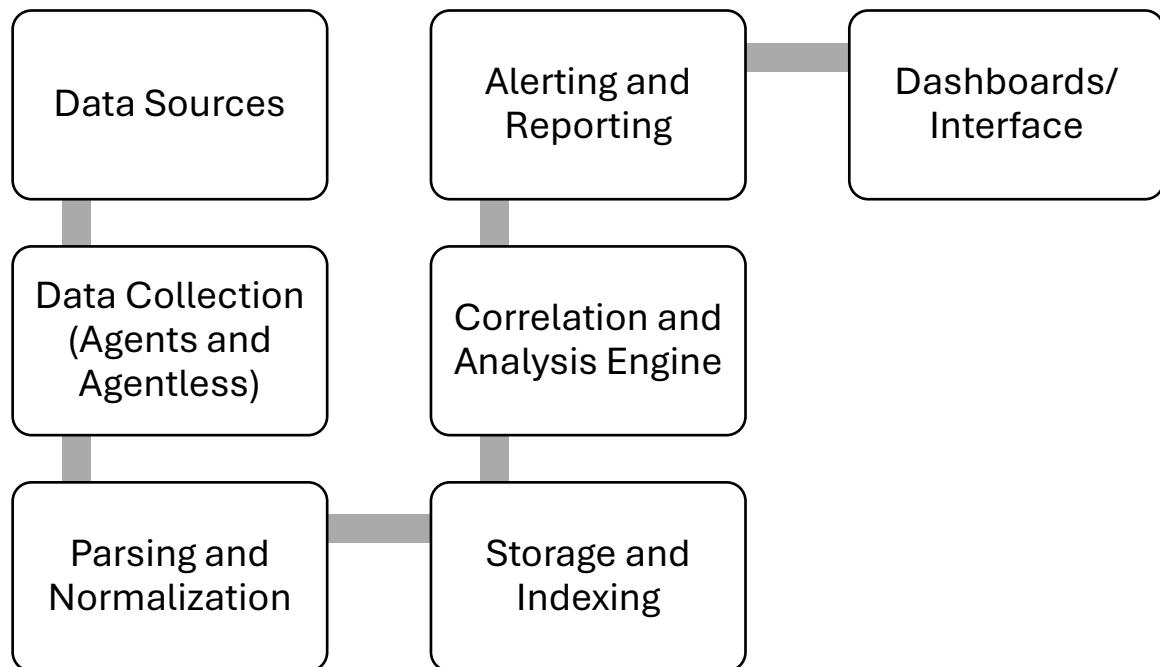


Figure 1 SIEM Architecture

1. Data Sources: The data refers to the events or logs that are fed into the SIEM. These data come from network devices, servers, security tools such as IDS/IPS and EDR, cloud services, databases and applications.
2. Data Collection: For collecting the data, agents may be installed on endpoints. If not, then agentless collection such as Syslog can be used as data collection.
3. Parsing and Normalization: The logs or data is passed into parsing engine for normalization. Logs come from different data sources, so they are in different formats. Normalization helps to convert each of those logs into a common format appropriate to the SIEM tool.
4. Storage and Indexing: The logs are stored. Raw logs are stored for compliance and forensic purposes. The indexed searchable storage is used for queries.

## Security Operations and Monitoring

There is log retention policy which determines how long logs can be stored. Generally, the log retention policy in most organizations is 1 year.

5. Correlation and Analysis engine: Correlation engine correlates different logs using rules and correlation logic to identify different patterns and it is useful to detect threats.
6. Alerting and Reporting: When the correlation engine detects any suspicious pattern, alert is generated via dashboards, emails, etc. and provides compliance reports.
7. Dashboard or Interface: It is the User Interface of SIEM. The data in SIEM can be visually represented through charts and graphs. It provides real-time monitoring of security events.

### 3 Introduction to SOC

#### 3.1 Overview

Security Operations Center (SOC) is a team of security professionals that uses various tools and techniques to monitor IT systems and detect any security incidents. They also analyze, respond and report on those incidents. Besides this, an SOC team can also assist in Vulnerability Assessment, Penetration Testing and Risk Management. While a red team in cybersecurity deals with offensive security, blue team involves defensive and SOC falls under the blue team category.

A Network Operations Center (NOC) is concerned with monitoring network. Even though it sounds like SOC, its main objective is to ensure reliable and smooth network while SOC is concerned with protecting systems from cyber threats. NOC deals with network monitoring, troubleshooting network issues, infrastructure management, etc. while SOC involves threat detection, log analysis, forensics, etc. (Knerler, Parker, & Zimmerman, 2022).

#### 3.2 Functions of SOC

Following are the key functions of Security Operations Center:

1. Continuously monitoring IT system and detecting potential intrusions.
2. Identification of vulnerabilities in a system.
3. Detection and response to incidents.
4. Investigation of incidents (forensic analysis).
5. Contain and recover the system from intrusions and eradicate the threats.
6. Threat hunting (identifying threats beyond the scope of SOC).
7. Ensure security practices align with regulatory requirements and compliances.
8. Proactive risk management

### 3.3 Pillars of SOC

The pillars of SOC are the key foundations that make up an SOC. This pillar consists of people, processes, technology and data.

1. People: People play key roles in SOC. A single person cannot handle all the work in an organization and similarly a single SOC analyst cannot do all the work by himself. So, the role of people has been divided into managers, security analysts and incident responders. Managers are supervisors in a SOC team ensuring other personnel are getting proper resources and are doing their work properly. Security analysts analyze different events and are responsible for detecting security incidents. They are the ones directly working with SIEM. Incident responders are responsible for carrying out responsive actions when incidents occur.
2. Process: Process refers to procedures or workflow in an SOC. There should be a system in which the personnel should work in an SOC. There should be plans regarding Incident Response, Risk Management, Change Management, Vulnerability Management and so on. Standard Operations Procedures (SOPs) should be developed for incident analysis and response.
3. Technology: Technology makes the work of people and processes easier. It provides tools to detect incidents, investigate them and respond to them. Some notable technologies include SIEM, EDR, IDS, etc. and there are so many tools that are used in SOC daily for different purposes.
4. Data: Data is the foundation of SOC. Every person, process and technology works on data. Data can be collected from different sources such as network devices, endpoints, applications, firewalls, etc. These data are processed and analysts use them for incident detection, investigation, etc. SIEM are fed with data, and they can visualize these data in their dashboards. They can correlate different data patterns and identify potential threats.

(Basta, Basta, Anwar, & Essar, 2024)

## 4 Endpoints Monitoring with Wazuh

### 4.1 Overview

Wazuh is an open-source SIEM tool for security monitoring of endpoints such as desktops, laptops and servers. It provides features such as configuration assessment, malware detection, file integrity monitoring, log data analysis, vulnerability detection, etc. Wazuh provides cloud and virtual machine deployments.

### 4.2 Features of Wazuh

Wazuh is an open-source SIEM tool that provides comprehensive features.

1. Threat detection: Wazuh can detect threats in real time and make alerts.
2. File Integrity Monitoring (FIM): Wazuh can monitor files or directories and detect any changes, alerting possible suspicious activities.
3. Vulnerabilities Detection: Wazuh can detect vulnerabilities such as misconfigurations, missing patches, etc. and provide remedy solutions.
4. Configuration Assessment: Wazuh audits configurations against security standards and reports the results. This is helpful to find any vulnerabilities in configurations.
5. Dashboard: Wazuh offers visualization and dashboards for events detected.

### 4.3 Agents in Wazuh

Agents are the software installed in endpoints that the host wants to monitor. Installing Wazuh agents in the system enables it to be monitored in Wazuh. Agents can be installed in virtual machines as well.

### 4.4 Lab Setup

The virtual lab was set up on the laptop, running Windows 11 as the host operating system, with VirtualBox hosting Wazuh, an Ubuntu server, and Kali Linux as virtual machines.

## Security Operations and Monitoring

| Agent hostname | Operating System              | IP Address   |
|----------------|-------------------------------|--------------|
| Aayush         | Windows 11                    | 192.168.1.73 |
| kali           | Kali Linux virtual machine    | 192.168.1.96 |
| aayush         | Ubuntu server virtual machine | 192.168.1.74 |

Table 1 Lab Setup

Wazuh is fed with Windows Event Logs from a Windows machine. But Windows Event Logs doesn't track a lot of things and therefore it is better to use Sysmon. Sysmon (System Monitor) is a Windows system service that provides detailed event logs for security monitoring and incident response.

Wazuh agents collect raw logs from endpoints and send them to Wazuh manager. Wazuh parses and normalizes logs automatically using built-in decoders and rules to normalize logs. A decoder parses a raw log and extracts relevant fields, and the rules classify the normalized logs in different categories.

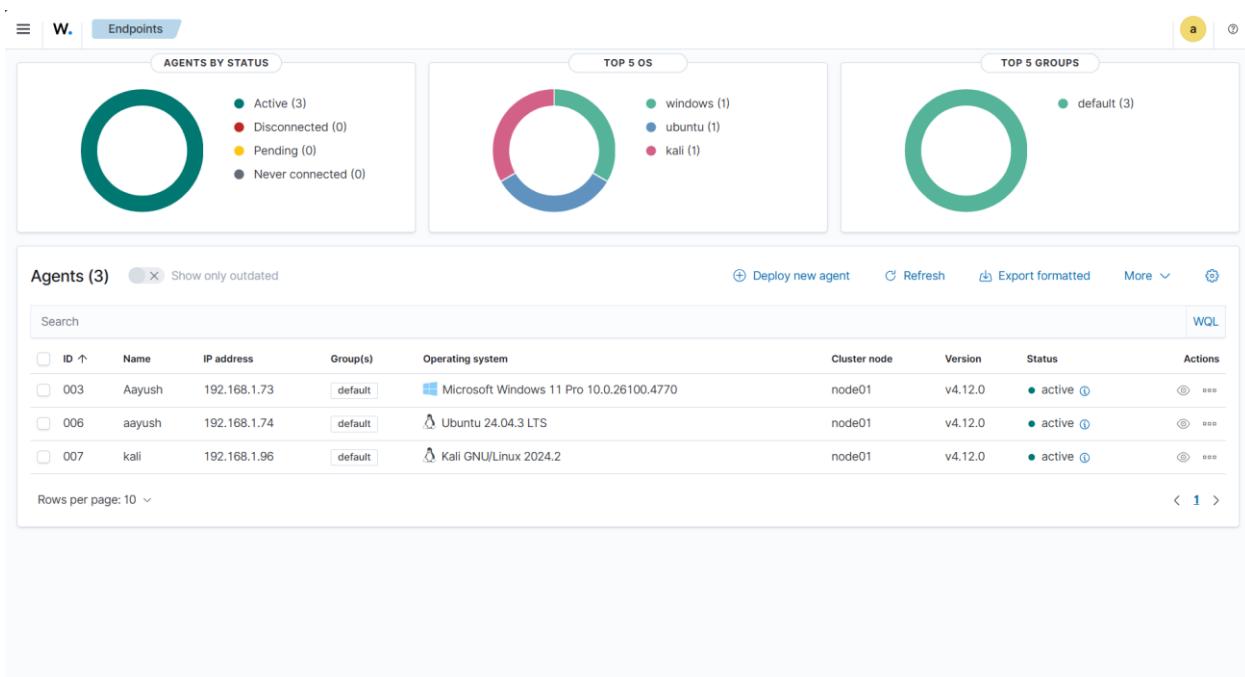
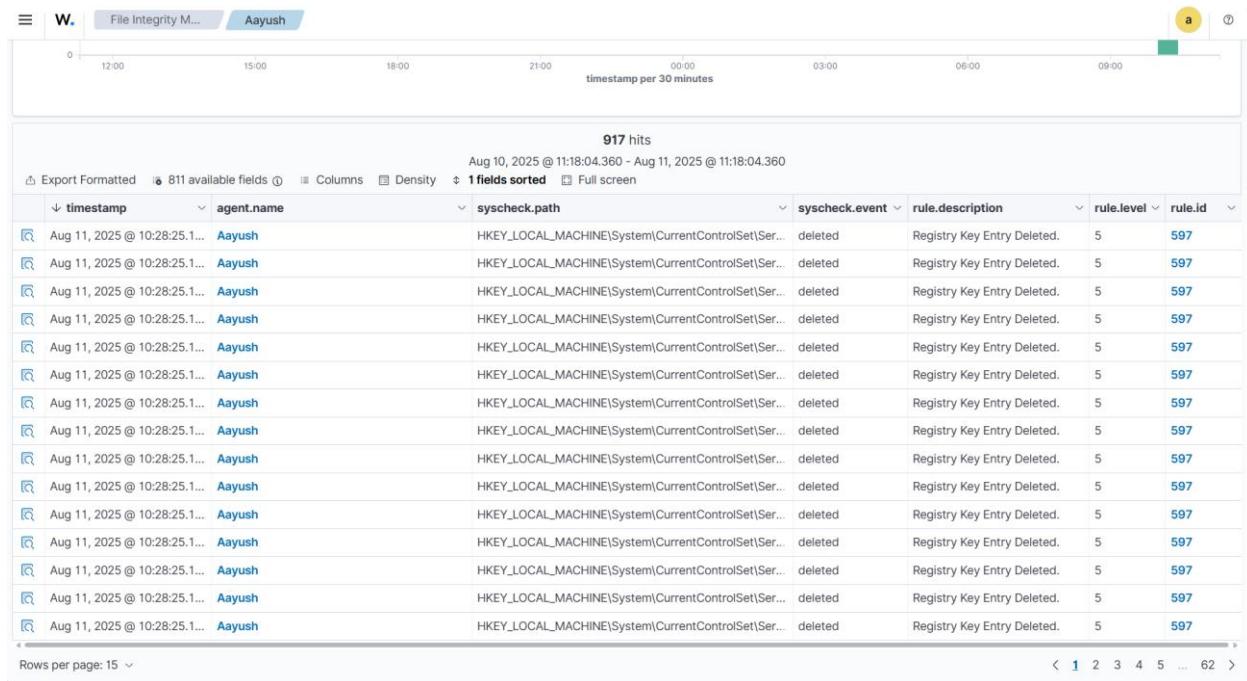


Figure 2 Endpoints Monitoring with Wazuh

## 4.5 File Integrity Monitoring

File Integrity Monitoring (FIM) is a feature of Wazuh which enables active monitoring of files and registry keys in agents which is helpful to detect any unauthorized activities in a file or directory. The agents should be configured to enable File Integrity Monitoring.



*Figure 3 Registry key logs in Wazuh*

In addition to registry key logs, a specific directory or files can be set up to monitor activities in them.

A directory 'Workstation Files' was created in Windows (Aayush) and setup for FIM.

In the File Integrity Monitoring section, we can view dashboards and logs related to the folders that have been set up for FIM.

Following activities were performed inside the 'Workstation Files' folder:

1. A bitmap image “New Bitmap image” was added.
  2. A new text file “FIM.txt” was added.
  3. A new folder “Word Documents” was created.
  4. The content of text file “FIM.txt” was changed.
  5. An MS Word file “MsWord” was added in “Word Documents” folder.
  6. “New Bitmap image” was deleted.

# Security Operations and Monitoring

## 7. “FIM.txt” was deleted.

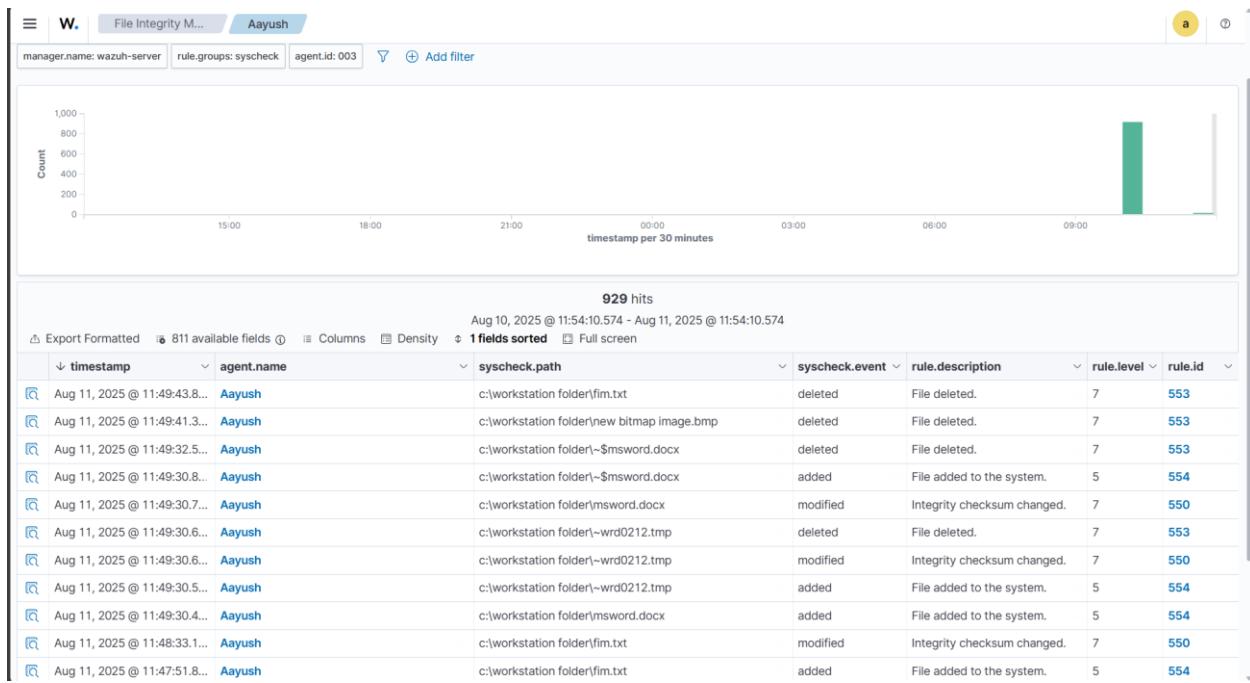


Figure 4 FIM logs

The FIM logs can be viewed of these activities. There are a lot of information we can collect related to the file changes such as:

1. timestamp: The date and time when the changes occurred.
2. agent\_name: The name of the agent where the changes occurred.
3. syscheck.path: The location and name of the file where the changes occurred.
4. syscheck.event: The event that happened in the file, such as add, delete or modify.
5. rule.description: The description of the event, such as ‘Integrity checksum changed’ means the file content had been modified.
6. rule\_level: The severity of the event.

| Rule Level | Description     |
|------------|-----------------|
| 0-3        | Low importance  |
| 4-6        | Warning         |
| 7-10       | Medium severity |
| 11-14      | High severity   |
| 15         | Critical        |

Table 2 Rule levels

## Security Operations and Monitoring

- rule\_id: The unique identifier of each Wazuh detection. Example, 550 is file modification, 1002 is new user creation, etc.

## 4.6 Configuration Assessment

Configuration Assessment in Wazuh allows verifying system configurations against security policies and compliances. Wazuh overviews the files and system configurations and audits it against the security standards, reporting any misconfigurations.

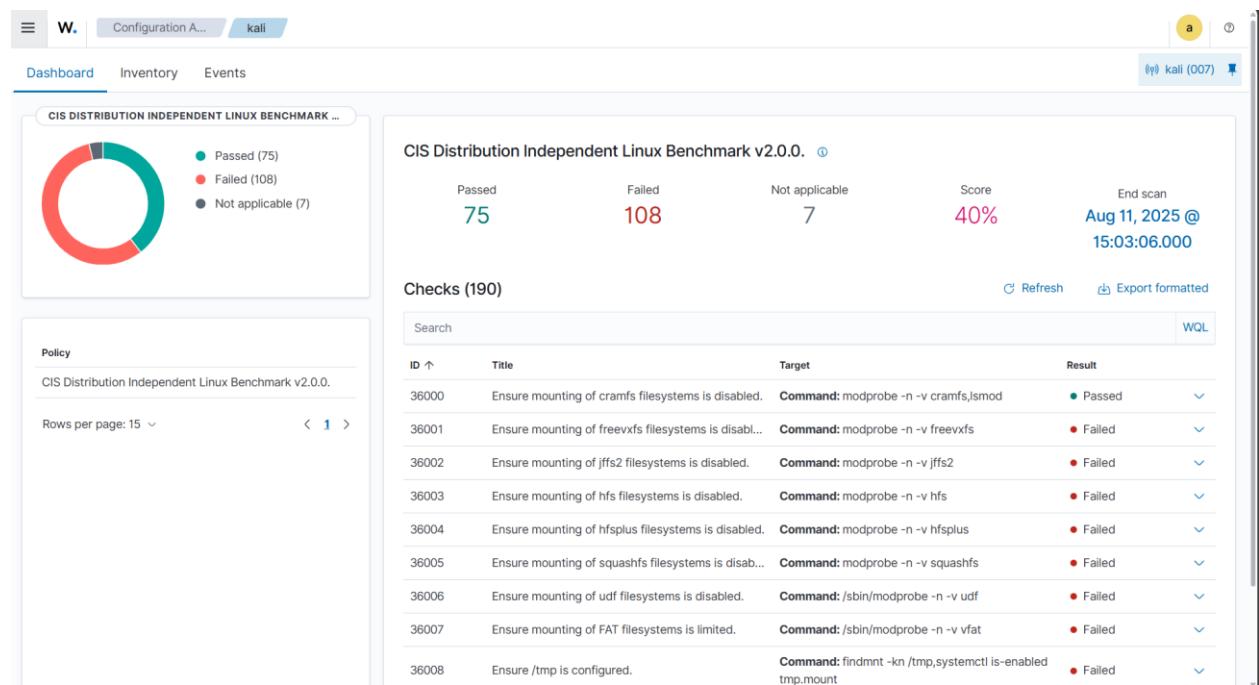


Figure 5 Configuration Assessment in Kali machine

CIS Distribution-Independent Linux Benchmark v2.0.0 is a security configuration benchmark for Linux against which Wazuh has compared and reported. Wazuh has performed 190 audits out of which 75 were passed and 108 were failed. There is a table of audit checks and detailed reports can be viewed for each of those checks.

## Security Operations and Monitoring

The screenshot shows a web-based configuration assessment tool. At the top, there are tabs for 'Configuration A...', 'kali', and a yellow circle with the letter 'a'. Below the tabs, there's a search bar and a table with the following columns: ID, Title, Target, and Result.

| ID    | Title  | Target | Result |
|-------|--|--------|--------|
| 36000 | Ensure mounting of cramfs filesystems is disabled. <b>Command:</b> modprobe -n -v cramfs;lsmod |        | Passed |

Below the table, there are sections for 'Rationale', 'Remediation', 'Description', 'Checks (Condition: all)', 'Compliance', and a large list of IDs from T1003 to T1539.

**Rationale:**  
Removing support for unneeded filesystem types reduces the local attack surface of the server. If this filesystem type is not needed, disable it.

**Remediation:**  
Edit or create a file in the /etc/modprobe.d/ directory ending in .conf Example: vim /etc/modprobe.d/cramfs.conf and add the following line: install cramfs /bin/true Run the following command to unload the cramfs module: # rmmod cramfs.

**Description:**  
The cramfs filesystem type is a compressed read-only Linux filesystem embedded in small footprint systems. A cramfs image can be used without having to first decompress the image.

**Checks (Condition: all):**

- c:modprobe -n -v cramfs → r:install /bin/false|install /bin/true|Module cramfs not found
- not c:lsmod → r:cramfs

**Compliance:**

cis: 1.1.1.1  
cis\_csc\_v7: 5.1  
cmmc\_v2.0: AC.1.002,CM.2.061,SC.3.180  
iso\_27001-2013: A.14.2.5,A.8.1.3  
mitre\_techniques:  
T1003,T1011,T1015,T1017,T1019,T1028,T1034,T1035,T1036,T1037,T1044,T1047,T1051,T1053,T1054,T1055,T1058,T1067,T1070,T1072,T1073,T1075,T1076,T1077,T1078,T1080,T1081,T1084,T1086,T1087,T1088,T1089,T1092,T1096,T1097,T1098,T1100,T1110,T1112,T1130,T1133,T1134,T1136,T1137,T1138,T1139,T1142,T1145,T1146,T1147,T1148,T1150,T1156,T1157,T1165,T1166,T1169,T1173,T1174,T1175,T1176,T1177,T1178,T1184,T1187,T1189,T1196,T1197,T1198,T1199,T1200,T1201,T1206,T1208,T1209,T1210,T1214,T1215,T1218,T1485,T1486,T1487,T1488,T1489,T1490,T1491,T1492,T1494,T1495,T1501,T1503,T1504,T1505,T1506,T1525,T1530,T1535,T1537,T1539

nist\_sp\_800-53: AU-2,CM-1,CM-2,CM-6,CM-7,IA-5,IA-6,SC-20,SC-21  
pci\_dss\_v3.2.1: 2.2

Figure 6 Configuration Assessment report

For each assessment, the report describes what the configuration is, how it can be configured, whether the system has been properly configured and other information. It is important for security teams in an organization to figure out misconfigurations to patch vulnerabilities they can pose.

## Security Operations and Monitoring

### 4.7 Vulnerabilities Detection

Vulnerabilities (security weaknesses) of the system can be monitored in Wazuh. The 'Vulnerability Detection' tab displays the tab of vulnerabilities in the system.

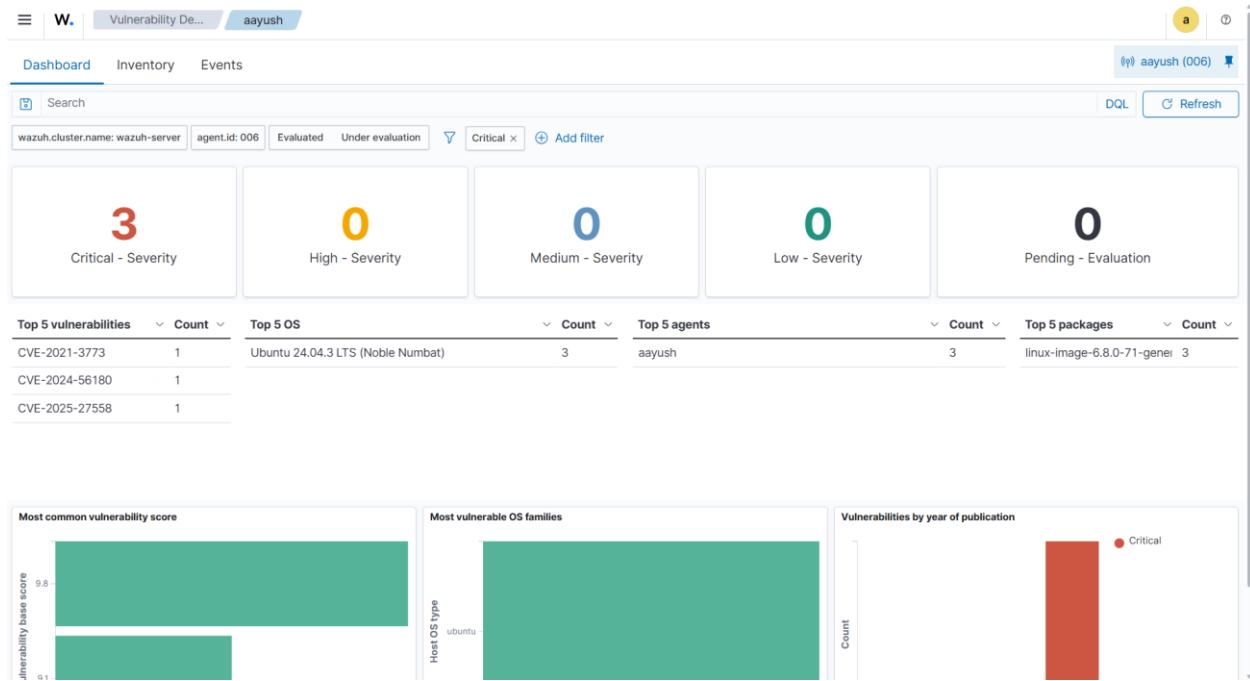


Figure 7 Vulnerabilities in Ubuntu machine

The dashboard displays the vulnerabilities along with severity. The Ubuntu machine has 3 critical vulnerabilities can there is a table displaying what those vulnerabilities are. Detailed information about those vulnerabilities can be looked up on the internet. For example, CVE-2021-3773 is a vulnerability in netfilter.

## Security Operations and Monitoring

The screenshot shows the NIST National Vulnerability Database (NVD) interface. At the top, there's a navigation bar with the NIST logo and a "NVD MENU" button. Below it, the "Information Technology Laboratory" and "NATIONAL VULNERABILITY DATABASE" are displayed. A green "VULNERABILITIES" button is visible. The main content area shows a detailed view of CVE-2021-3773. It includes sections for "MODIFIED" (with a note about enrichment), "Description" (describing a flaw in netfilter), and "QUICK INFO" (listing details like CVE Dictionary Entry, NVD Published Date, and Source). A "Metrics" section provides CVSS scores for Version 4.0, 3.x, and 2.0. Below the metrics, there's a note about NVD enrichment efforts and a link to CVSS 3.x Severity and Vector Strings.

Figure 8 Information about vulnerability in NIST website (National Institute of Standards and Technology (NIST), 2022)

## 4.8 Threat Hunting

Threat Hunting in Wazuh involves analyzing different data sources to identify cyber threats. It is a proactive measure. It helps security teams by alerting potential threats.

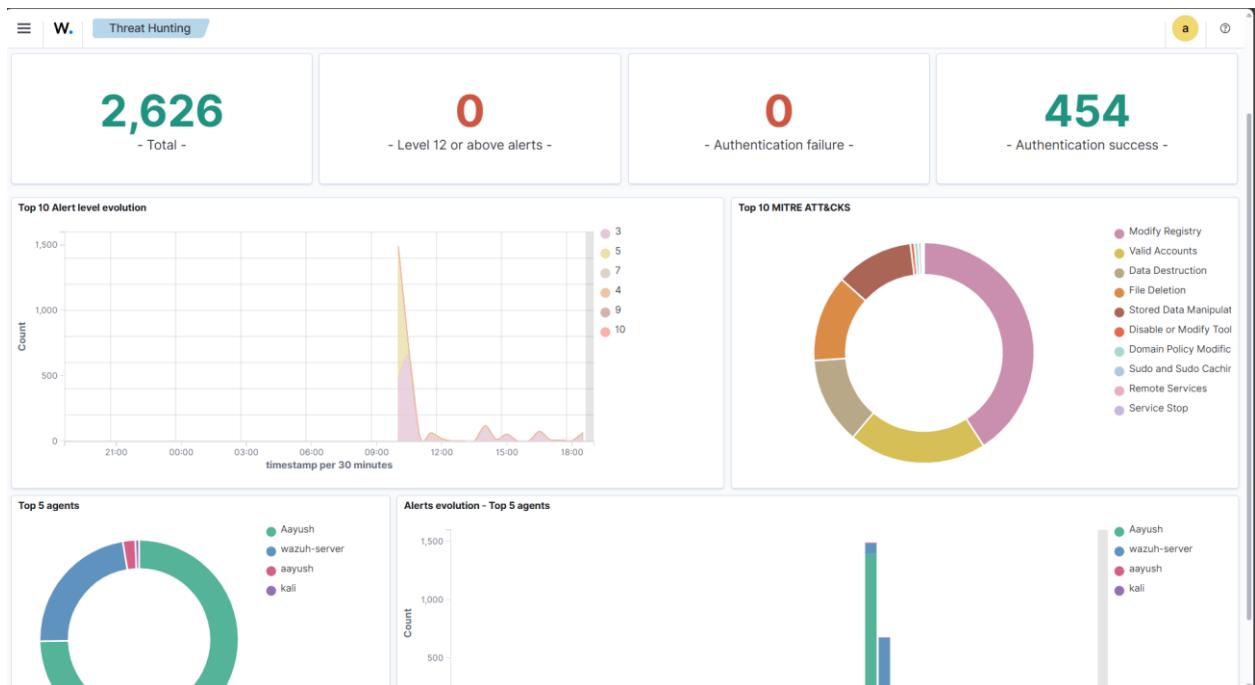


Figure 9 Threat Hunting dashboard

## Security Operations and Monitoring

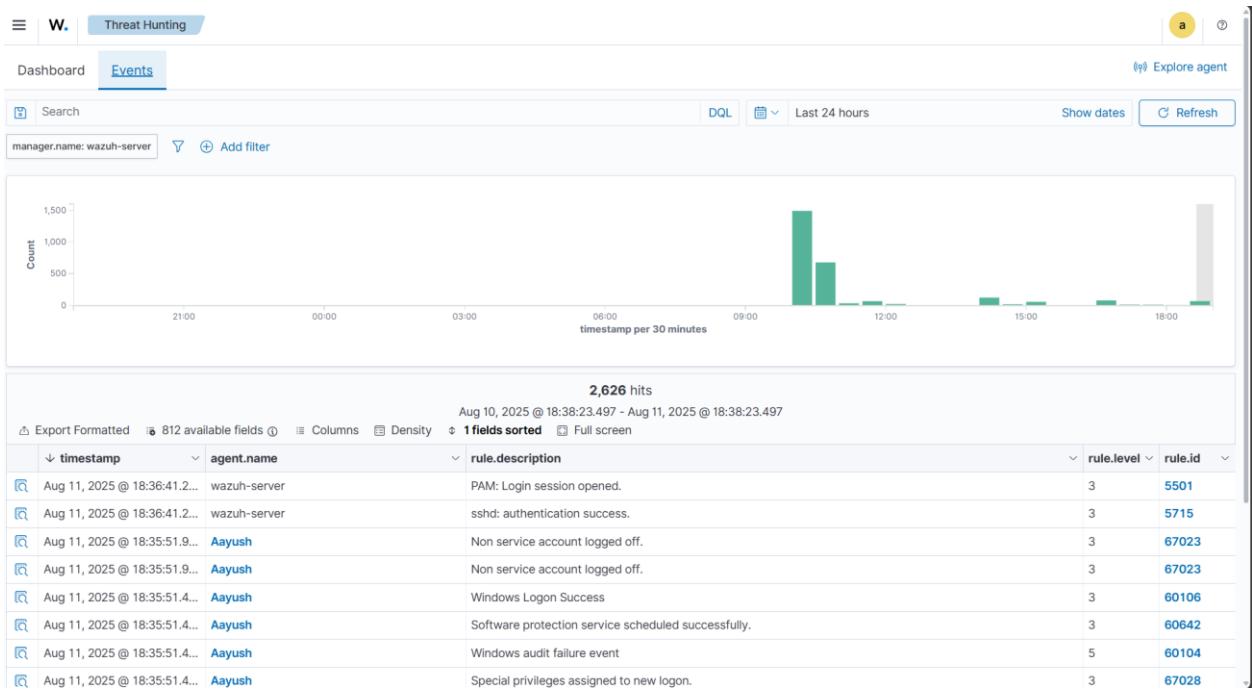


Figure 10 Threat Hunting events

There are logs in different events. The first two indicate the Wazuh server was opened using SSH.

Wazuh can detect potential threats and make an alert. Here, Kali was used to port scan towards Ubuntu machine which found SSH port open and tried to connect SSH by brute-forcing using hydra.

```
(ayush㉿kali)-[~]
└─$ nmap -sV 192.168.1.74
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-08-11 09:16 EDT
Nmap scan report for 192.168.1.74
Host is up (0.002s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh    OpenSSH 9.0p1 Ubuntu 3ubuntu13.13 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http   Apache httpd 2.4.58 ((Ubuntu))
MAC Address: 00:0C:27:D6:B3:46 (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 22.04 seconds

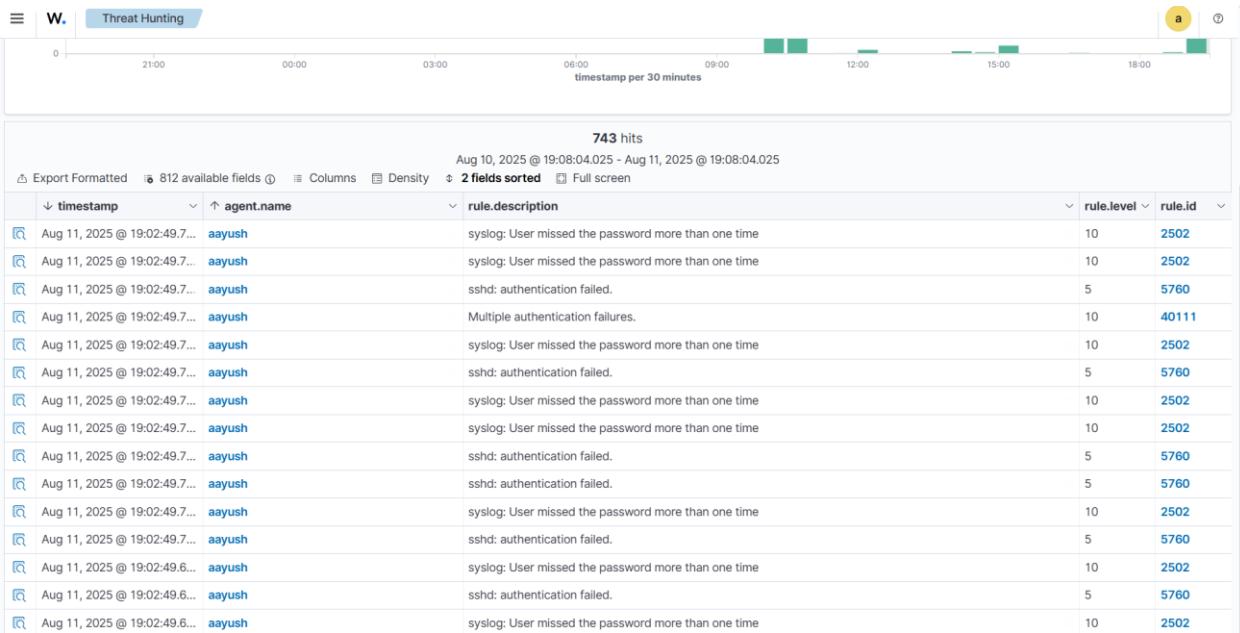
(ayush㉿kali)-[~]
└─$ hydra -1 ayush -r /usr/share/wordlists/wfuzz/others/common_pass.txt ssh://192.168.1.74
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-08-11 09:17:16
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 52 login tries (l:1:p:52), -4 tries per task
[DATA] attacking ssh://192.168.1.74:22/
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-08-11 09:17:27
```

Figure 11 SSH Brute-force from Kali towards Ubuntu machine

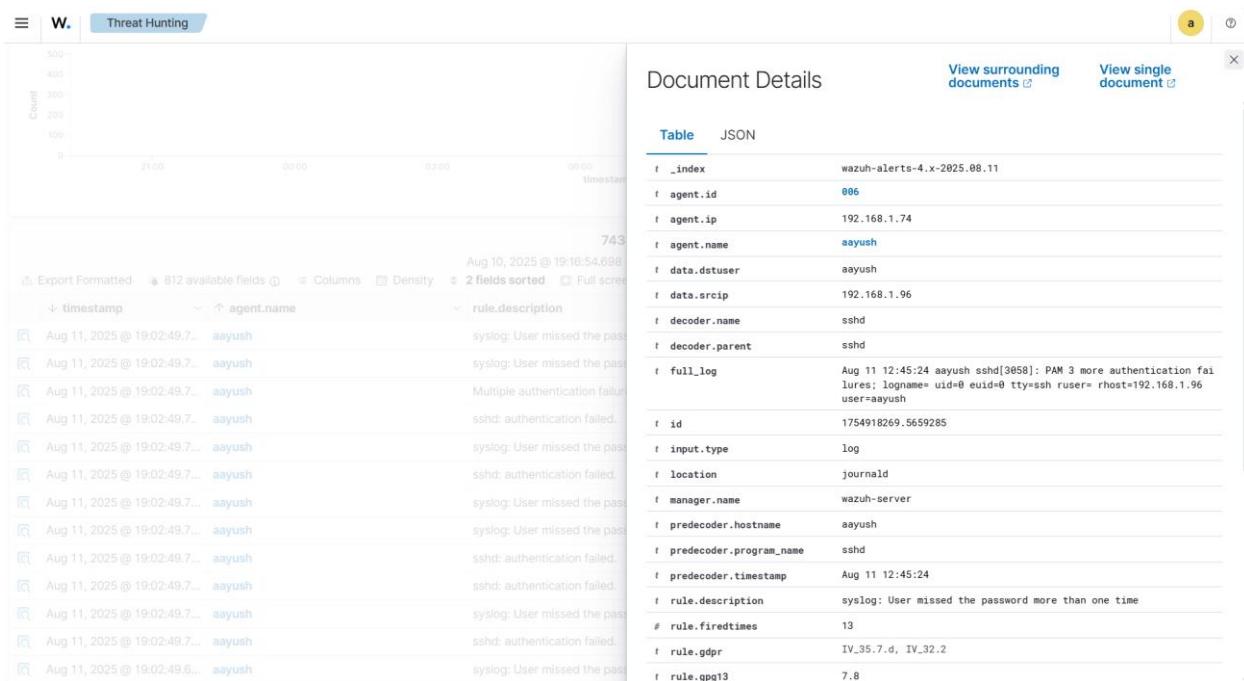
The Brute-force didn't work against the password set for the Ubuntu machine, but the logs of the brute-force attack can be seen in Wazuh and it has provided rule level 10 which is considerably high severity.

# Security Operations and Monitoring



*Figure 12 SSH Brute-force result logs*

Reading the description which says that there were multiple failed login attempts, it can be concluded that it was a brute-force attempt. Clicking on the magnifier icon on the left gives the details of the event.



*Figure 13 Threat Hunting - Event detail 1*

## Security Operations and Monitoring

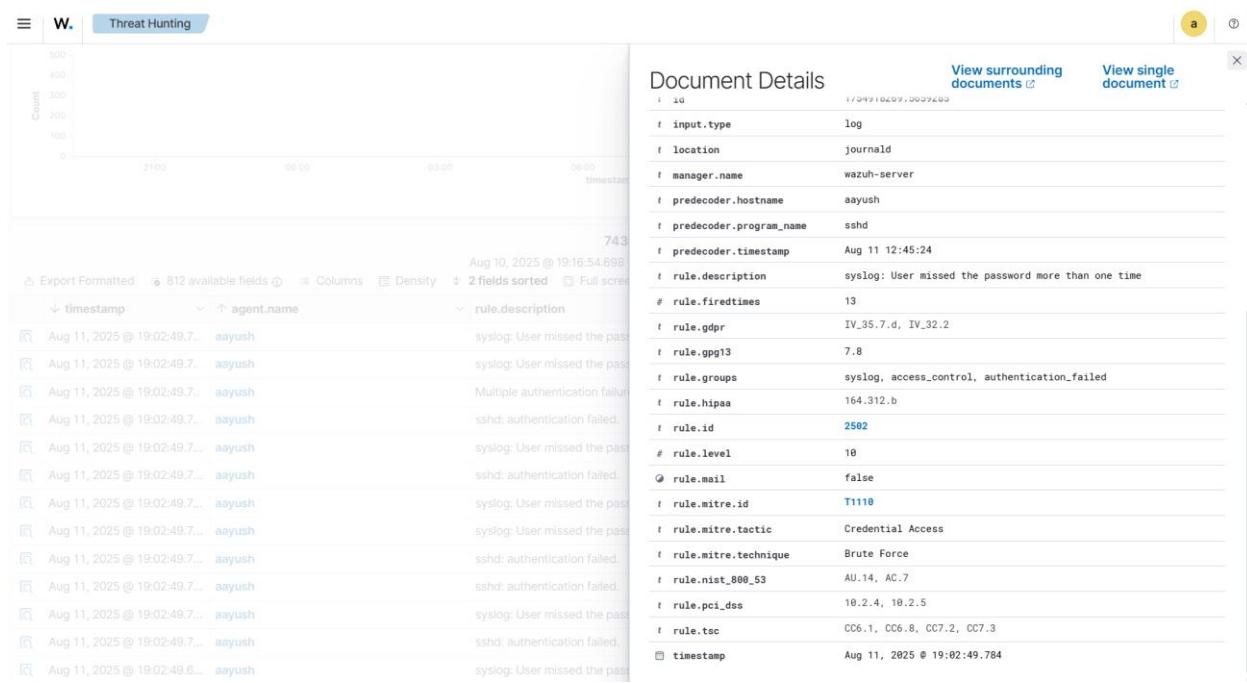


Figure 14 Threat Hunting - Event detail 2

There is so much information about the event. There is data\_srcip which informs the IP of the user from whom the brute-force attack is coming. In this case, it is 192.168.1.96 which is the IP of Kali machine. There is also information about rules which also mentions that the MITRE tactic used was Credential Access and technique was Brute-force. There is rule id which can help us investigate more.

|       |   |
|-------|---|
| 2502  | User missed password more than one time                               |
| 5760  | Authentication failed   |
| 40111 | Multiple Authentication failed  |
| 5763  | Brute force trying to get access to the system. Authentication failed |
| 5551  | Multiple failed logins attempts in short time                         |
| 5503  | User login failed   |

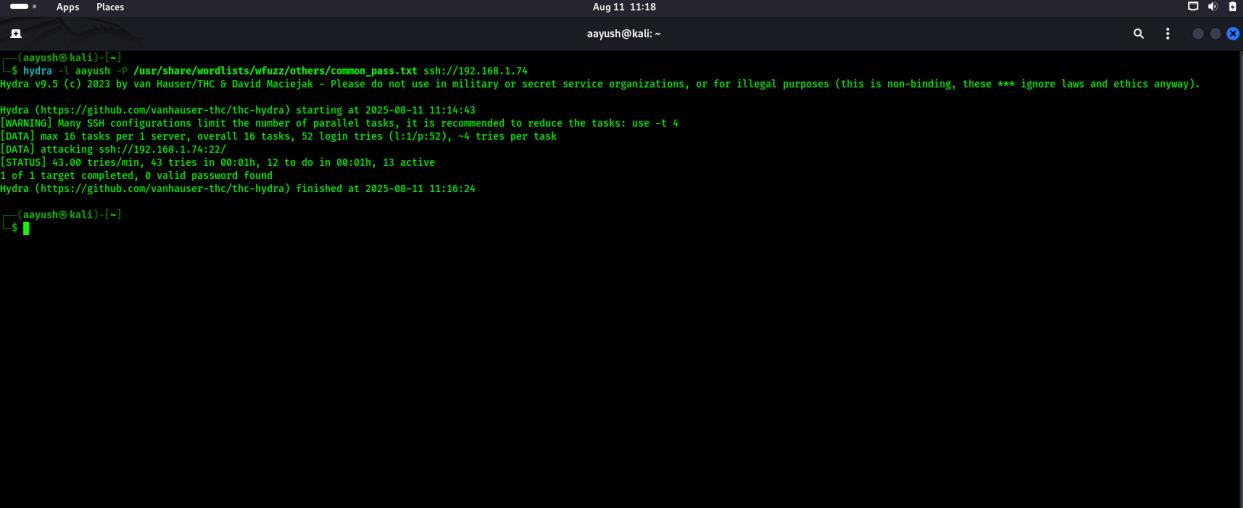
Figure 15 Rules from SSH Brute-force

### 4.9 Active Response - Firewall Drop

Wazuh firewall drop means silently ignoring the network packet instead of rejecting it. When a computer sends network packet to a system which has firewall drop enabled against that packet, the system just ignores it, refusing to forward the packet and doesn't even send any message back to the sender.

## Security Operations and Monitoring

Firewall drop for rule ID 5551 was triggered in Wazuh against rule 5551 and from Kali Linux machine, the hydra Brute-force was attempted again.



```
(ayush㉿kali)-[~]
$ hydra -L ayush -P /usr/share/wordlists/wfuzz/others/common_pass.txt ssh://192.168.1.74
hydra v0.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-08-11 11:14:43
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 10 tasks per 1 server, overall 10 tasks, 52 login tries (1:l/p:S2), -4 tries per task
[DATA] attacking ssh://192.168.1.74:22/
[STATUS] 43.00 tries/min, 12 to do in 00:01h, 13 active
1 of 1 target completed, 0 valid password found
hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-08-11 11:16:24

(ayush㉿kali)-[~]
$
```

Figure 16 SSH Brute-forcing from Kali on Ubuntu machine

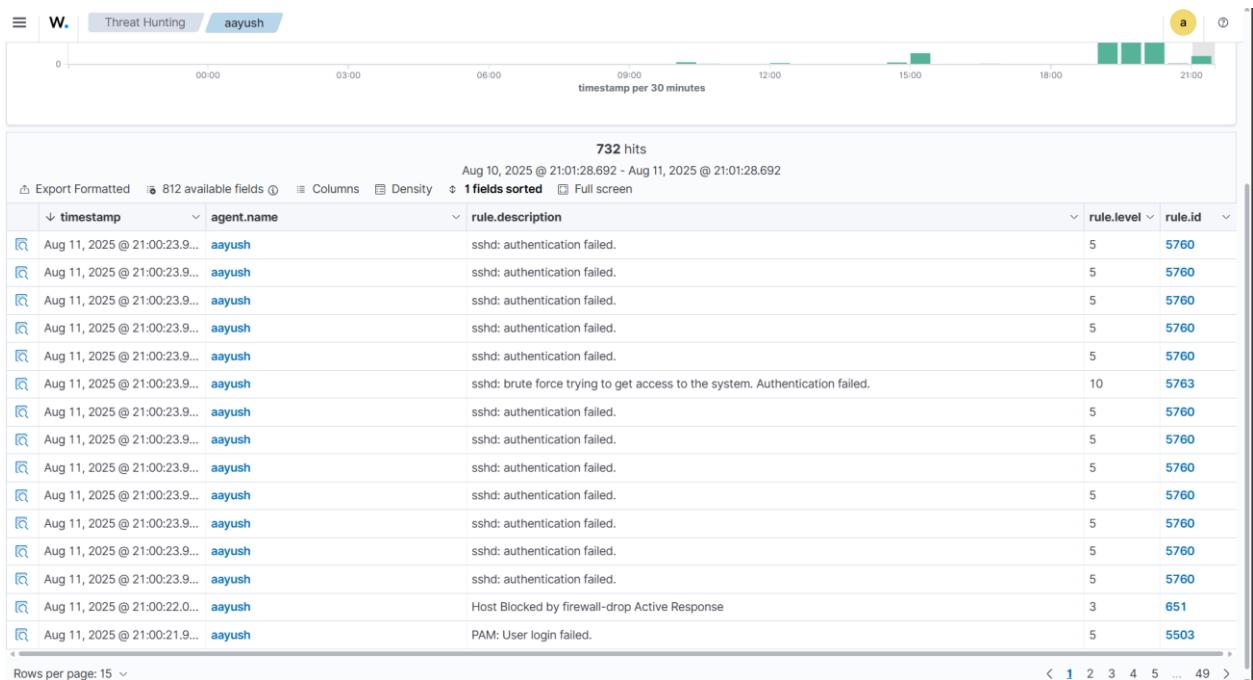


Figure 17 SSH Brute-forcing result logs

One of the logs shows that the host was blocked by firewall-drop. This happened as rule ID 5551 was triggered. The configuration was made to block for 180 seconds. So, the host IP (Kali) is blocked for 180 seconds.

### 4.10 Malware Detection

Wazuh has YARA integration which can scan files and alerts when found malicious signature. In addition to that, Virustotal can also be integrated to enhance malware detection.

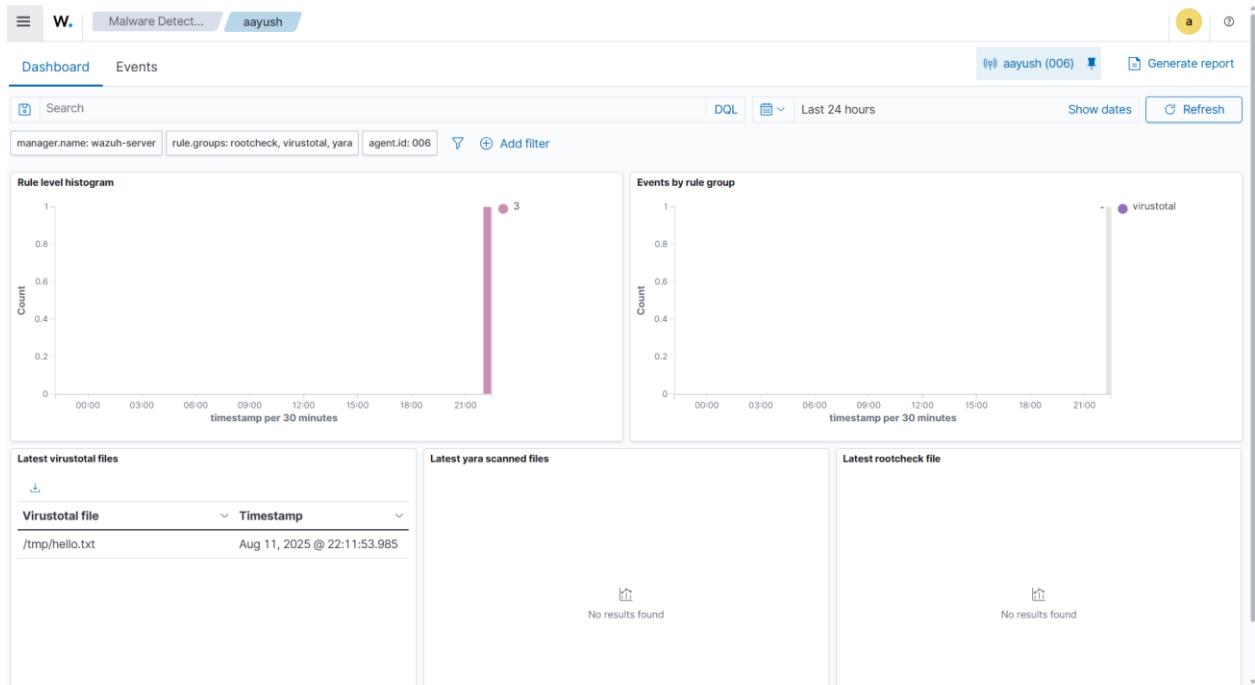


Figure 18 Malware Detection dashboard for Ubuntu machine

The Malware Detection dashboard doesn't have anything except a text file creation as there is no malware present in the system. Malware detection can be tested using features using EICAR which is a standard malware test file. The \tmp folder was configured for FIM and an EICAR file was created in that folder in ubuntu machine.

## Security Operations and Monitoring

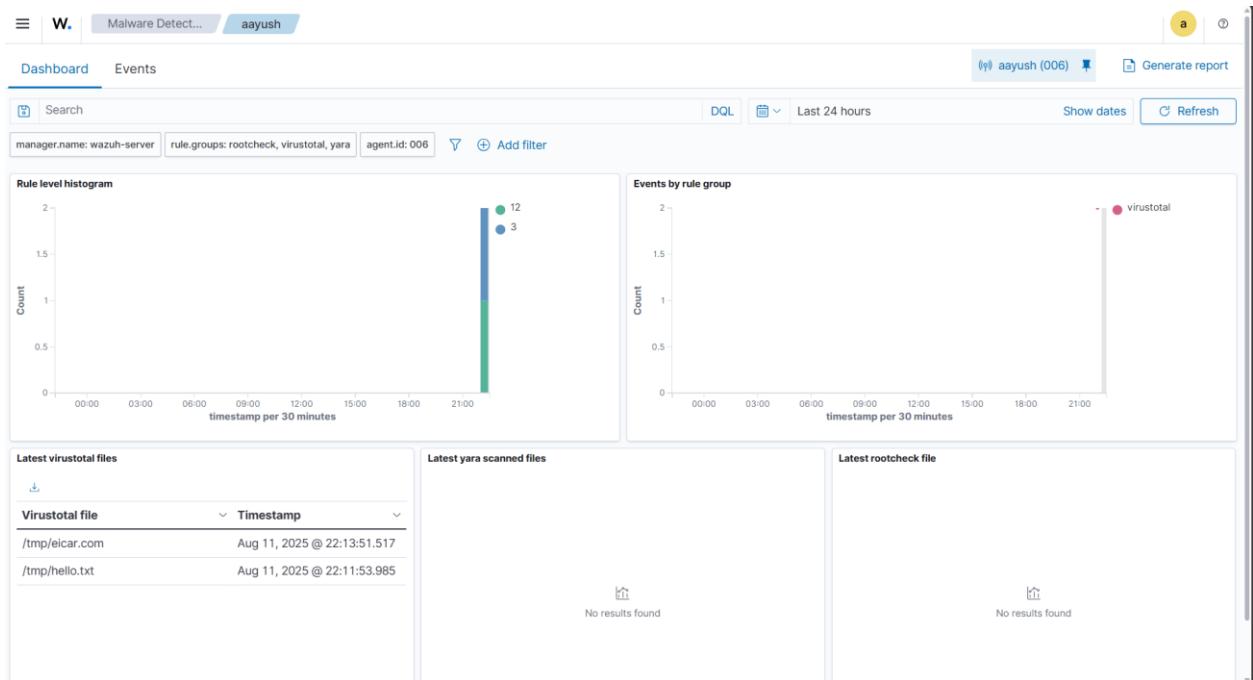


Figure 19 Malware Detection dashboard in Ubuntu machine

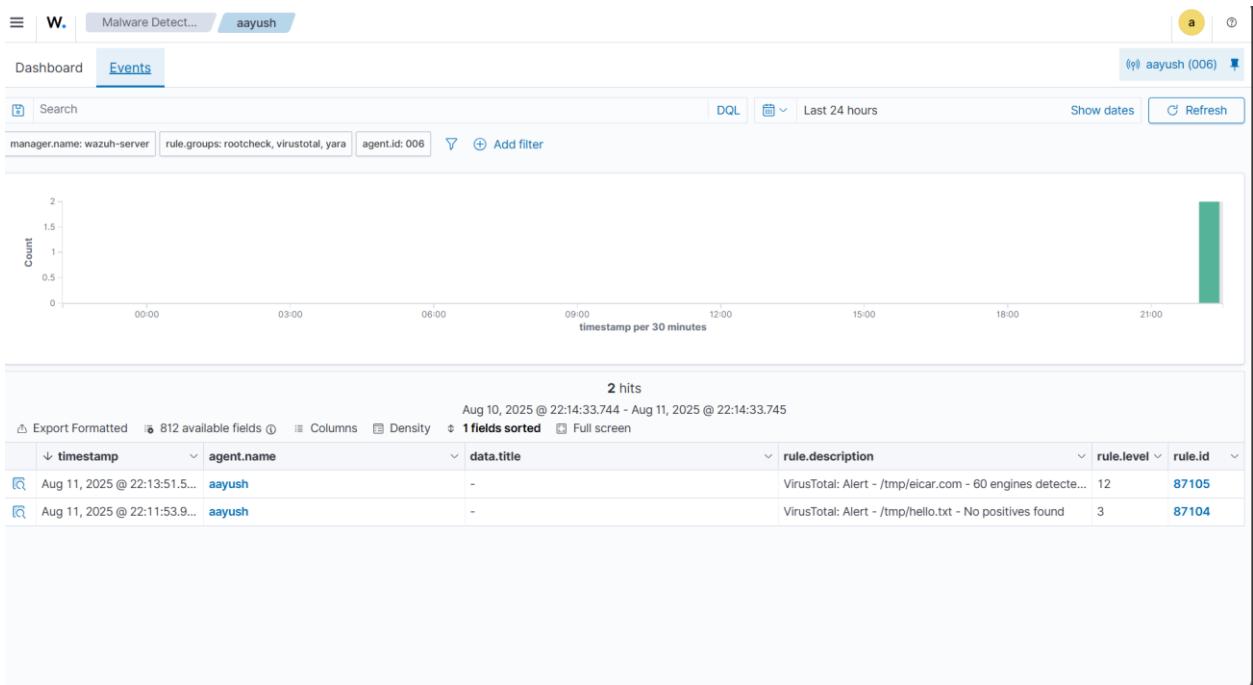
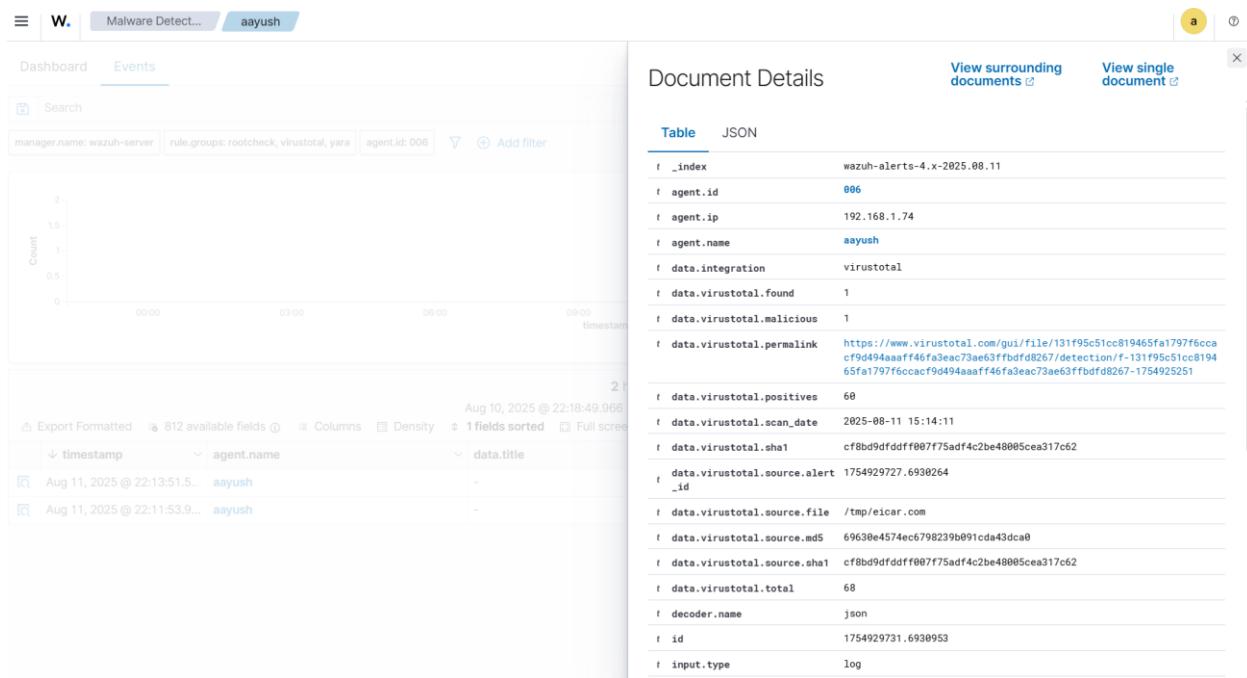


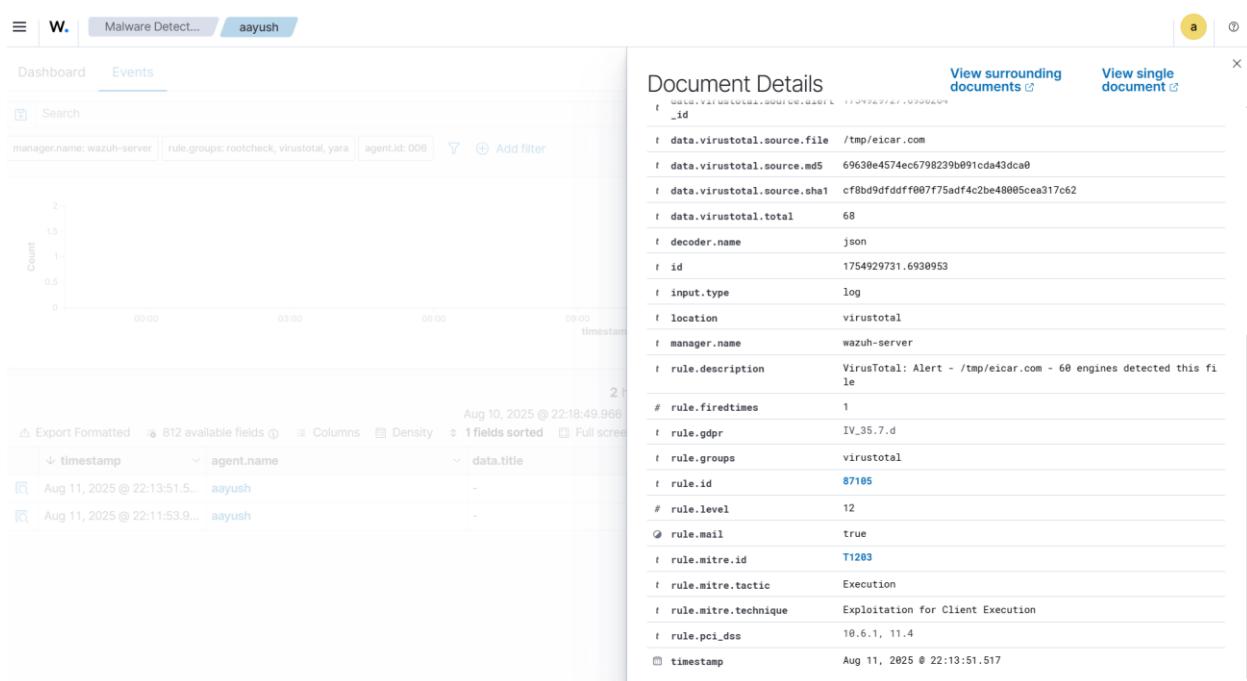
Figure 20 Malware Detection log in Ubuntu machine

The EICAR file creation has been assigned rule level 12 which is high.

## Security Operations and Monitoring



*Figure 21 Malware detail I*



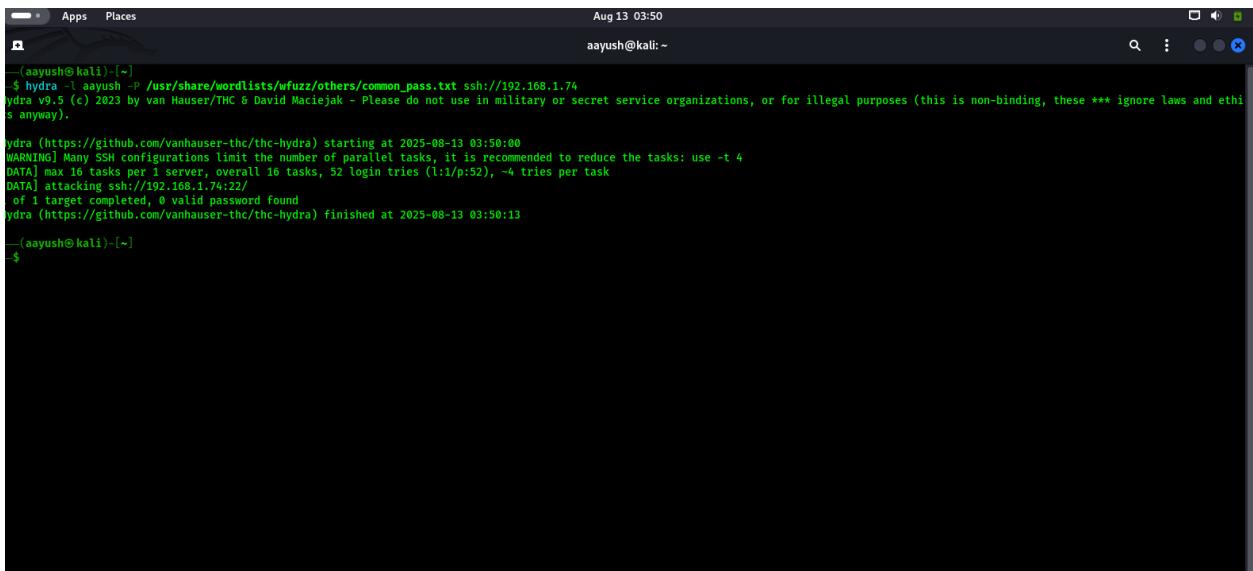
*Figure 22 Malware detail II*

From inspecting the details of the EICAR file creation, the detection was caused by Virustotal integration. Different information about the malware can be found.

### 4.11 Alerting

When Wazuh generates alerts, it can be forwarded to email. As the SIEM alerts in the Security Operations Center (SOC) won't always be under the security team's eyes, this is necessary to catch any significant alerts.

An email account was integrated into Wazuh. Wazuh should forward alerts from level 9 and above. From Kali machine, SSH Brute-force was launched against Ubuntu machine to raise alert level 10.



The screenshot shows a terminal window on a Kali Linux desktop environment. The title bar indicates the session is for user 'ayush' at 'kali'. The date and time 'Aug 13 03:50' are shown in the top right. The terminal window displays the command:

```
hydra -l ayush -P /usr/share/wordlists/wfuzz/others/common_pass.txt ssh://192.168.1.74
```

Output from the command:

```
hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-08-13 03:50:00
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 52 login tries (l:1:p:52), -4 tries per task
[DATA] attacking ssh://192.168.1.74:22/
  of 1 target completed, 0 valid password found
hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-08-13 03:50:13
```

The terminal prompt shows the user has completed the attack.

Figure 23 SSH Brute-force from Kali against Ubuntu

## Security Operations and Monitoring

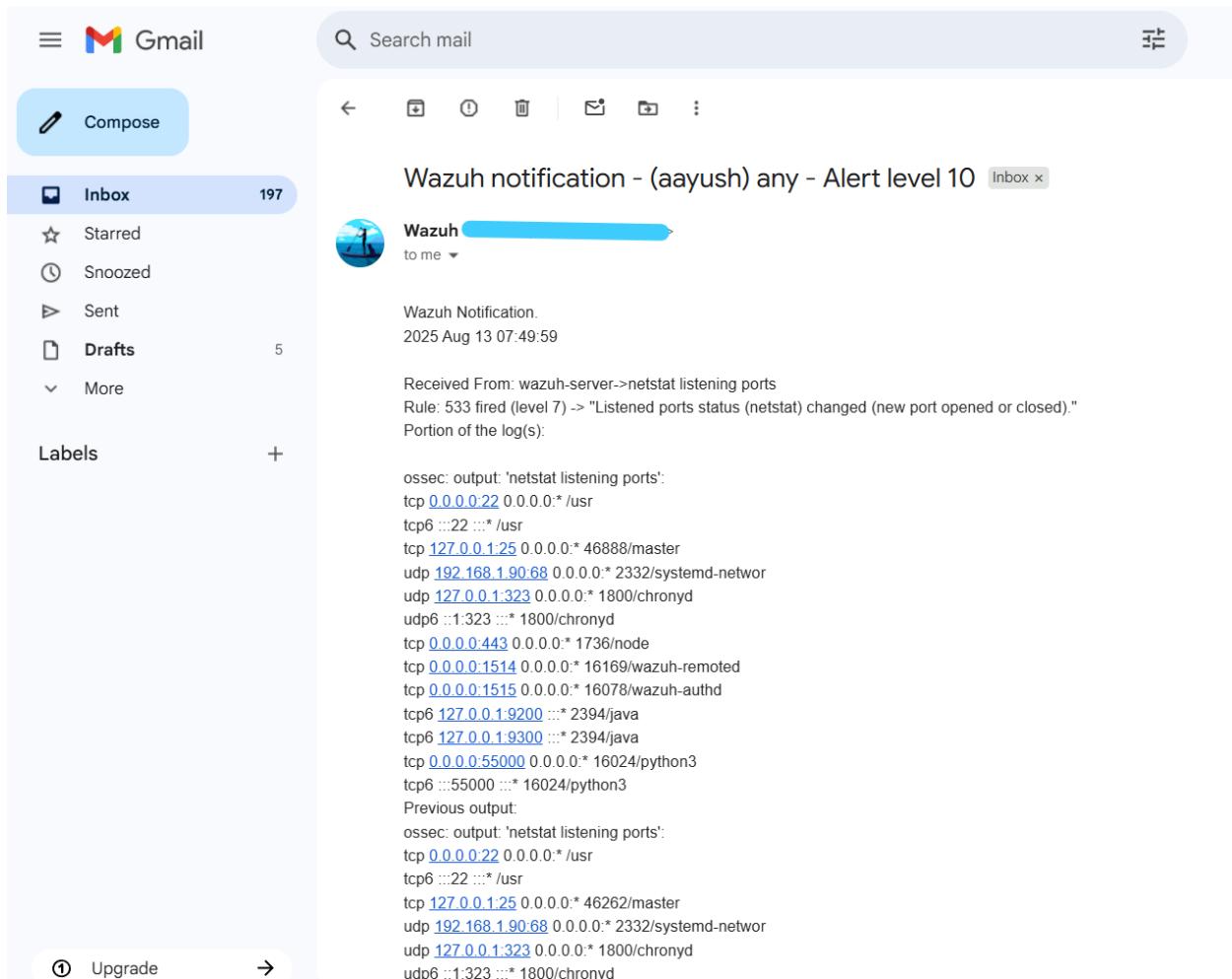


Figure 24 Email alert

## 5 Incident Handling

### 5.1 Overview

The role of a SOC analyst becomes important when an incident occurs. According to NIST, “Incidents are actions taken through the use of an information system or network that result in an actual or potentially adverse effect on an information system, network, and/or the information residing therein.” Examples of incidents can include the following:

- Phishing attack
- Malware infections
- Ransomware attack
- Web-based attacks (SQL Injection, Cross-site Scripting, etc.)
- Denial of Service (DoS)
- Unauthorized access to the system
- Device theft

(Knerler, Parker, & Zimmerman, 2022)

### 5.2 Incident Response

Incident Response and Incident Handling are interchangeable terms although Incident Handling can be classified as a broader term. Incident Response is a structured approach to respond to incidents. The goal of incident response planning is to minimize the impact, reduce recovery time and ultimately reduce losses. An incident can cost an organization its finances and operations. So, a proper incident response can help reduce such risks (EC Council, 2024).

### 5.3 Incident Response Lifecycle

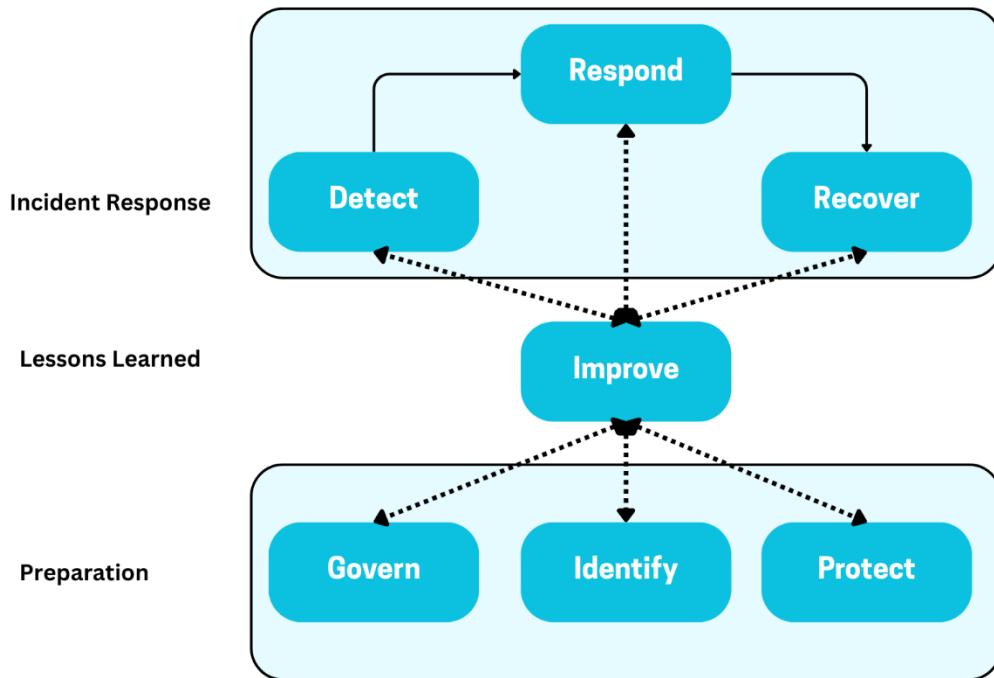


Figure 25 Incident Response Lifecycle (National Institute of Standards and Technology (NIST), 2025)

The three primary pillars of Incident Response are Detect, Respond and Recover. A SIEM tool can detect incidents, and the security team should effectively response to the incident, limiting the loss it can cause. The affected assets need to be recovered. The process of Govern, Identify and Protect aren't directly involved in Incident Response. They are related to risk management and indirectly support incident response. Every process is learned for the team and helps to improve SOC and incident handling process (National Institute of Standards and Technology (NIST), 2025).

### 5.4 Phases of Incident Response

Incident Response can be broken down in five phases:

1. Identification: The first phase involves identifying the security risks in an organization. To identify the risks, it is important to understand organization system, IT assets, information the organization holds and operations. Risk Assessment can be done, which is the process which includes identifying risks, analyzing them and categorizing them based on their severity.

## Security Operations and Monitoring

2. Protection: After assessing the risks, it is important to protect the organization from the threats they can cause. So, proper plans and controls should be implemented for protection of organization from cyber threats. Some security measures include:
  - a. Access Controls
  - b. Technology such as EDR, IDS/IPS, firewall, etc.
  - c. Risk Management Strategy
  - d. Vulnerability Assessment and Penetration Testing
  - e. Employee Training
3. Detection: A system can never be 100% secure. So, it is important to set up an incident detection system. This is where SIEM comes. It helps security teams to continuously monitor the systems and detect any security incidents.
4. Response: When a security incident is detected, there should be a proper response. Incident Response strategies should be developed to provide a framework for security teams and other employees to respond when an incident occurs. Some response strategies include:
  - a. Network isolation: Isolate the compromised part of the network to prevent compromising other parts of the network.
  - b. Account suspension: Disable compromised user accounts to prevent unauthorized access.
  - c. Quarantine: Shut down the compromised IT system and disconnect devices from the network. Remove malicious content from the compromised device or replace it with a new one.
  - d. Communication: Communicate the incident with security team, management, etc. and notify the customers and stakeholders as soon as possible.
5. Recover: Incidents would cause organization operations to stop. Proper recovery measures should be implemented to recover the organization's IT environment and overall operations. Following are some recovery measures:
  - a. Remove malicious contents from compromised system and recover data from backups.
  - b. Replace damaged system with new ones.
  - c. Update and patch system.
  - d. Reset credentials.

## Security Operations and Monitoring

- e. Implement additional security controls.

(EC Council, 2024)

The NIST Incident Response framework involves four steps:

1. Preparation: A set of policies and strategies should be developed for incident response. The staff should be trained to respond to a security incident and monitoring tools (SIEM) should be set up.
2. Detection and Analysis: When an incident occurs, they should be detected by the monitoring tool the organization has used. Different detection systems such as SIEM, Intrusion Detection System (IDS) and Endpoint Detection and Response (EDR) can collect and correlate logs to detect security incidents in a system. The security team should analyze the detected threat and collect different information regarding the incident such as:
  - a. What incident occurred?
  - b. How did it occur? What did it exploit?
  - c. When did it occur?
  - d. Which system or data was/were compromised?
  - e. Who was involved?
  - f. Who was targeted?
  - g. What is the impact?
  - h. Was the goal of the attack achieved?
3. Containment, Eradication and Recovery: The impact of the incident should be limited. Good immediate response plans can be helpful to limit the impact of the incident. This can include isolating compromised network, quarantine infected files, etc. That system that has been infected or compromised should be cleaned up or replaced. System should be updated and the vulnerabilities should be patched. Security controls should be implemented. Credentials should be changed.
4. Post-incident Activity: At the end of Incident Response is the post-incident activity. Every step till now should be documented. Meetings should be conducted to review the incident and what the organization can learn from it. It is helpful to strengthen proactive security measures.

### 6 Conclusion

An SOC is a team assembled to handle incidents in an organization. It consists of a manager who works as a supervisor in the SOC, security analysts who are responsible for monitoring events and incident responders who work to respond to any security incidents. An SOC is built by its four pillars which are people, process, technology and data. SIEM is one of the components of SOC. A SIEM is a tool that provides real-time monitoring of events and a centralized place to view events from different sources such as devices, networks, firewall, applications, etc. Data or logs from these sources are fed into SIEM which parses and normalizes them and represent them in dashboards for visual interpretation. Data should be stored based on retention policy for forensics or any other future purposes. Wazuh is an open-source SIEM tool. It provides comprehensive features such as File Integrity Monitoring, Assessing system configurations, Vulnerabilities Detection, Threat Intelligence, Threat Hunting, Malware Detection and more. An SOC team is also responsible for handling incidents. An incident response team should be set up in an organization. The steps for incident response include preparing plans for incident-response, detecting and analyzing the incident, limiting the impact of the incident, removing the threats from the incident, recovering the compromised assets and documenting the entire situation to learn lessons and implement necessary security controls.

## 7 References

Basta, A., Basta, N., Anwar, W., & Essar, M. I. (2024). *Open-Source Security Operations Center (SOC)*. Wiley.

EC Council. (2024, March). *Incident Handling*. Retrieved from EC Council: <https://www.eccouncil.org/cybersecurity-exchange/incident-handling/what-is-incident-response/>

Fortinet. (n.d.). *What is SIEM*. Retrieved from Fortinet: <https://www.fortinet.com/resources/cyberglossary/what-is-siem>

Knerler, K., Parker, I., & Zimmerman, C. (2022). *11 Strategies of a World-Class Cybersecurity Operations Center*. McLean, VA: MITRE.

National Institute of Standards and Technology (NIST). (2022, February 16). *Vulnerability detail CVE-2021-3773*. Retrieved from NIST: <https://nvd.nist.gov/vuln/detail/CVE-2021-3773>

National Institute of Standards and Technology (NIST). (2025, April). *Incident Response*. Retrieved from NIST: <https://csrc.nist.gov/projects/incident-response>

## 8 Appendix

### 8.1 Sysmon Integration in Windows

1. Configure the ossec.conf file in Windows agent:

Open notepad with administrator privilege and open ossec.conf file that can be found in ossec-agent directory. Add this block to the file:

```
<localfile>

<log_format>eventchannel</log_format>

<location>Microsoft-Windows-Sysmon/Operational</location>

</localfile>
```

2. Configure Sysmon rules in Wazuh manager:

Open rules.xml file in the Wazuh server with following command:

```
/var/ossec/etc/rules/local_rules.xml
```

Add this block in the xml file:

```
<group name="windows,sysmon">

<rule id="100001" level="10">

    <if_sid>61610</if_sid> <!-- Sysmon process creation -->

    <field name="win.system.providerName">Microsoft-Windows-
Sysmon</field>

    <match>powershell.exe</match>

    <description>PowerShell execution detected
(Sysmon)</description>

</rule>

</group>
```

3. Restart the agent with the following command in Command-Prompt:

```
net stop wazuh-agent
```

```
net start wazuh-agent
```

4. Restart Wazuh manager from the Wazuh server:

```
sudo systemctl restart wazuh-manager
```

## 8.2 File Integrity Monitoring

1. Open ossec-conf file in Windows and there is <!--File Integrity Monitoring> section. In linux open the ossec-conf file with this command:

```
sudo nano /var/ossec/etc/ossec.conf
```

2. Add this block in the configuration file:

```
<directories realtime="yes" report_changes="yes"  
check_all="yes">Directory_path</directories>
```

3. Restart the agent with the following command in Command-Prompt (Windows):

```
net stop wazuh-agent
```

```
net start wazuh-agent
```

In Linux:

```
systemctl restart wazuh-agent
```

## 8.3 Virustotal Integration

1. Login to Virustotal and get API key.
2. Open configuration file in Wazuh manager with this command:

```
sudo nano /var/ossec/etc/ossec.conf
```

Add this block:

```
<integration>  
<name>virustotal</name>  
<api_key>API_KEY</api_key> <!-- Replace with your VirusTotal  
API key -->  
<group>syscheck</group>
```

## Security Operations and Monitoring

```
<alert_format>json</alert_format>  
</integration>
```

3. Restart Wazuh manager with this command:

```
sudo systemctl restart wazuh-manager
```

### 8.4 Firewall Drop

1. Go to ossec.conf on manager with the command:

```
sudo nano /var/ossec/etc/ossec.conf
```

Add this block:

```
<active-response>  
  <command>firewall-drop</command>  
  <location>local</location>  
  <rules_id>Rule_id</rules_id>  
  <timeout>180</timeout>  
</active-response>
```

2. Restart Wazuh manager with this command:

```
sudo systemctl restart wazuh-manager
```

### 8.5 Email Alerts

1. Install Postfix in Wazuh server with these commands:

```
sudo yum install postfix cyrus-sasl-plain mailx -y  
sudo systemctl enable postfix  
sudo systemctl start postfix
```

2. Configure Postfix to forward emails by opening main.cf file with this command:

```
sudo nano /etc/postfix/main.cf
```

Add this block:

## Security Operations and Monitoring

```
relayhost = [smtp.gmail.com]:587
smtp_use_tls = yes
smtp_sasl_auth_enable = yes
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd
smtp_sasl_security_options = noanonymous
smtp_tls_CAfile = /etc/ssl/certs/ca-bundle.crt
```

3. Get Gmail account's app password and set Gmail credentials by making a `sasl_passwd` directory with this command:

```
sudo nano /etc/postfix/sasl_passwd
```

Add this block:

```
[smtp.gmail.com]:587 Gmail address@gmail.com:App password
```

4. Execute these commands:

```
sudo chmod 600 /etc/postfix/sasl_passwd
```

```
sudo postmap /etc/postfix/sasl_passwd
```

3. Restart Postfix with this command:

```
sudo systemctl restart postfix
```

4. Go to ossec.conf on manager with the command:

```
sudo nano /var/ossec/etc/ossec.conf
```

Modify this block:

```
<global>

<jsonout_output>yes</jsonout_output>

<alerts_log>yes</alerts_log>

<logall>no</logall>

<logall_json>no</logall_json>

<email_notification>yes</email_notification>
```

## Security Operations and Monitoring

```
<smtp_server>localhost</smtp_server>

<email_from>wazuh@example.wazuh.com</email_from>

<email_to>Gmail_address@gmail.com</email_to>

<email_maxperhour>12</email_maxperhour>

<email_log_source>alerts.log</email_log_source>

<agents_disconnection_time>10m</agents_disconnection_time>

<agents_disconnection_alert_time>0</agents_disconnection_alert_time>

<update_check>yes</update_check>

</global>

<alerts>

<log_alert_level>3</log_alert_level>

<email_alert_level>9</email_alert_level>

</alerts>
```

### 5. Restart Wazuh manager with this command:

```
systemctl restart wazuh-manager
```

## 8.6 EICAR execution

### 1. Run this command in powershell:

```
Set-Content -Path "Path_to_directory\eicar.com" -Value
"X5O!P%@AP[4\PZX54 (P^) 7CC]7}EICAR-STANDARD-ANTIVIRUS-TEST-
FILE!$H+H*"
```

Wazuh will generate an event of file addition and check hashes in Virus Total. It will generate a Virus Total alert.

## 8.7 SSH Brute-force using Hydra

1. Run this command in terminal:

```
Hydra -l Target_username -P Path_to_text_file_of_wordlist  
ssh://Target_ip
```