SECURITY INFORMATION AND EVENT MANAGEMENT

A PROJECT REPORT

Submitted by

VICKY KUMAR	**210101120004**
VASUDEV JHA	**210101120010**
PRAGYA KUMARI	**210101120027**
ANIL RAJ	**210101120180**
ATUL VAIBHAV	**210101120138**
MANISH KUMAR	**210101120244**

in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Centurion UNIVERSITY
Shaping Lives...
Empowering Communities...

SCHOOL OF ENGINEERING AND TECHNOLOGY

PARALAKHEMINDI CAMPUS

CENTURION UNIVERSITY OF TECHNOLOGY AND MANAGEMENT

ODISHA

APRIL 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF ENGINEERING AND TECHNOLOGY
PARALAKHEMINDI CAMPUS

BONAFIDE CERTIFICATE

Certified that this project report **Security Information and Event Management** is the bonafide work of "**VICKY KUMAR"** who carried out the project work under my supervision. This is to further certify to the best of my knowledge, that this project has not been carried out earlier in this institute and the university.

Dr. Suvendu Kumar Nayak
Professor of Computer Science and Engineering

Certified that the above-mentioned project has been duly carried out as per the norms of the college and statutes of the university.

Dr. Devendra Maharana
HEAD OF THE DEPARTMENT
Professor of Computer Science and Engineering

DEPARTMENT SEAL

DECLARATION

I hereby declare that the project entitled "**Security Information and Event Management**" submitted for the "Domain Project" of 5th semester B. Tech in Computer Science and Engineering is my original work and the project has not formed the basis for the award of any Degree / Diploma or any other similar titles in any other University / Institute.

Name of the Student: VICKY KUMAR

Signature of the Student:

Registration No: 210101120004

Place: PARALAKHEMINDI

Date: 12/04/2024

ACKNOWLEDGEMENTS

I wish to express my profound and sincere gratitude to Dr. Suvendu Kumar Nayak Department of Computer Science and Engineering, SoET, Paralakhemundi Campus, who guided me into the intricacies of this project nonchalantly with matchless magnanimity.

I thank Dr. Debendra Maharana, Head of the Dept. of Department of Computer Science and Engineering, SoET, Paralakhemundi Campus and Dr. Praffula Panda, Dean, School of Engineering and Technology, Paralakhemundi Campus for extending their support during Course of this investigation.

I would be failing in my duty if I don't acknowledge the cooperation rendered during various stages of image interpretation by CyberDojo Team

I am highly grateful to Mr. Yash Raval who evinced keen interest and invaluable support in the progress and successful completion of my project work.

I am indebted to teammates for their constant encouragement, co-operation and help. Words of gratitude are not enough to describe the accommodation and fortitude which they have shown throughout my endeavor.

Name of the Student: VICKY KUMAR

Signature of the Student:

Registration No: 210101120004

Place: PARALAKHEMINDI

Date: 12/04/2024

TABLE OF CONTENTS

ABSTRACT

In an era dominated by digital technologies, cybersecurity has emerged as a critical concern for organizations worldwide. The increasing frequency and sophistication of cyber threats necessitate robust security measures to protect sensitive data and critical infrastructure. Security Information and Event Management (SIEM) systems have become indispensable tools for organizations seeking to monitor, detect, and respond to security incidents in real-time. This project focuses on the design and implementation of a SIEM system using the ELK stack (Elasticsearch, Logstash, and Kibana), Beats, and Wazuh.

The primary objective of this project is to develop a comprehensive SIEM system that enhances an organization's overall cybersecurity posture. This is achieved through the integration of the ELK stack, Beats, and Wazuh to collect, store, and analyze security-related data from various sources, such as logs, events, and network traffic. The system is designed to provide security analysts with a user-friendly interface for monitoring security events and incidents, enabling them to quickly identify and respond to potential threats.

Key components of the project include designing a scalable and resilient SIEM architecture, implementing data collection mechanisms using Beats, enhancing security monitoring capabilities with Wazuh, and developing custom dashboards and visualizations in Kibana. Thorough testing and validation are conducted to ensure the effectiveness and reliability of the SIEM system. Comprehensive documentation and training sessions are provided to facilitate knowledge sharing and capacity building within the organization.

The insights gained from this project can provide valuable guidance for organizations looking to enhance their cybersecurity defenses and protect against emerging threats. By empowering organizations with the tools and knowledge needed to effectively safeguard their digital assets, this project aims to contribute to a more secure and resilient cyber landscape.

CHAPTER – 1

INTRODUCTION

1.1 Introduction

In today's digital age, cybersecurity has become a critical concern for organizations worldwide. The increasing frequency and sophistication of cyber threats highlight the need for robust security measures to protect sensitive data and critical infrastructure. Security Information and Event Management (SIEM) systems have emerged as a key component of modern cybersecurity strategies, providing organizations with the ability to monitor, detect, and respond to security incidents in real-time.

This project focuses on the design and implementation of a SIEM system using the ELK stack, Beats, and Wazuh. The ELK stack, comprised of Elasticsearch, Logstash, and Kibana, offers a powerful platform for collecting, storing, and analyzing security-related data. Beats, lightweight data shippers, facilitate the collection of data from various sources, while Wazuh enhances security monitoring and threat detection capabilities.

In addition to the technological aspects, this project also considers the operational and organizational implications of implementing a SIEM system. It explores best practices for configuring and fine-tuning the SIEM system to maximize its effectiveness in real-world scenarios. Furthermore, considerations such as scalability, integration with existing security infrastructure, and user training are addressed to ensure the successful deployment and operation of the SIEM system.

By leveraging the capabilities of the ELK stack, Beats, and Wazuh, this SIEM system aims to enhance the overall security posture of organizations, helping them mitigate the risks posed by cyber threats. The insights gained from this project can provide valuable guidance for organizations looking to enhance their cybersecurity defenses and protect against emerging threats in an increasingly interconnected world. The ultimate goal of this project is to empower organizations with the tools and knowledge needed to effectively safeguard their digital assets and maintain a strong security posture in the face of evolving cyber threats.

1.2 Objective

The objective of this project is to design and implement a Security Information and Event Management (SIEM) system using the ELK stack, Beats, and Wazuh. The system aims to provide organizations with an effective platform for monitoring, detecting, and responding to security incidents in real-time. Specific objectives include:

Designing the SIEM architecture: Define the architecture of the SIEM system, including the components of the ELK stack, Beats, and Wazuh, and how they will be integrated to collect, store, and analyze security-related data.

Implementing data collection: Configure Beats to collect data from various sources, such as logs, events, and network traffic, and ingest them into the SIEM system for analysis.

Enhancing security monitoring: Utilize Wazuh to enhance security monitoring capabilities, including log analysis, anomaly detection, and threat intelligence integration.

Developing custom dashboards and visualizations: Create custom dashboards and visualizations in Kibana to provide a user-friendly interface for monitoring security events and incidents.

Testing and validation: Conduct thorough testing and validation of the SIEM system to ensure its effectiveness in detecting and responding to security threats.

Documentation and knowledge sharing: Provide comprehensive documentation of the SIEM system design, implementation, and operational procedures to facilitate knowledge sharing and future maintenance.

By achieving these objectives, this project aims to empower organizations with a powerful SIEM solution that can enhance their cybersecurity posture and enable them to respond effectively to the ever-evolving threat landscape.

CHAPTER – 2

BACKGROUND

Cybersecurity is a growing concern for organizations worldwide, given the increasing frequency and sophistication of cyber threats. The management and analysis of security-related data pose significant challenges for organizations, including the volume and diversity of data, the complexity of modern IT environments, and the need for timely detection and response to security incidents. Traditional Security Information and Event Management (SIEM) systems have limitations in terms of cost, complexity, and scalability, leading to the need for more efficient and effective solutions.

2.1 Problem Statement

The increasing frequency and sophistication of cyber threats pose significant challenges for organizations in effectively managing and analyzing security-related data. Traditional Security Information and Event Management (SIEM) systems are often costly, complex, and difficult to scale, limiting their effectiveness in addressing modern cybersecurity challenges. There is a need for more efficient and cost-effective SIEM solutions that can help organizations improve their security monitoring and threat detection capabilities.

2.1.1 Challenges Faced by Organizations

Organizations face several challenges in managing and analysing security-related data. The volume and diversity of data generated by modern IT environments make it difficult to effectively monitor for security incidents. Additionally, the complexity of modern IT infrastructures, with a mix of on-premises and cloud-based systems, further complicates security monitoring efforts. Organizations also struggle with the timely detection and response to security incidents, often due to the lack of visibility into their IT environments and the inability to correlate security events across different systems.

2.1.2 Role of SIEM Systems

Security Information and Event Management (SIEM) systems play a crucial role in helping organizations monitor, detect, and respond to security incidents. SIEM systems collect and analyse security-related data from various sources, such as logs, events, and network traffic, to provide organizations with a comprehensive view of their security posture. By correlating and analysing this data, SIEM systems can identify potential security threats and alert security teams to take action. Additionally, SIEM systems help organizations meet regulatory compliance requirements by providing detailed security event logs and reports.

2.1.3 Limitations of Traditional SIEM Systems

Traditional Security Information and Event Management (SIEM) systems have several limitations that hinder their effectiveness in addressing modern cybersecurity challenges. These limitations include high costs associated with licensing, implementation, and maintenance, making them inaccessible to many organizations. Additionally, traditional SIEM systems often require specialized skills to deploy and manage, leading to complexity and resource constraints for organizations. Scalability is another challenge, as traditional SIEM systems may struggle to handle the increasing volume and diversity of security data generated by modern IT environments. Lastly, traditional SIEM systems may lack the ability to provide real-time visibility and response capabilities, leaving organizations vulnerable to advanced and persistent threats.

2.2 Aim of SIEM

A Security Information and Event Management (SIEM) system aims to provide organizations with a platform for real-time monitoring, detection, and response to security incidents. By collecting, storing, and analyzing security data from various sources, including logs and network traffic, SIEM systems help organizations improve their security posture and protect against cyber threats. SIEM systems also assist organizations in meeting regulatory compliance requirements by providing detailed security event logs and reports.

Additionally, SIEM systems help organizations streamline their security operations by providing a centralized platform for managing security alerts and incidents. By correlating and analyzing security data from multiple sources, SIEM systems help security teams identify and prioritize security incidents, enabling them to respond quickly and effectively. This proactive approach to security helps organizations minimize the impact of security breaches and ensure the security of their digital assets.

CHAPTER – 3

PROPOSED METHODOLOGY

The proposed methodology for this project involves designing and implementing a Security Information and Event Management (SIEM) system using the ELK stack, Beats, and Wazuh. The methodology will include an initial assessment of the organization's security needs and objectives, followed by the design of the SIEM architecture. Data collection mechanisms will be configured using Beats, and security monitoring capabilities will be enhanced with Wazuh. Custom dashboards and visualizations will be developed in Kibana to provide a user-friendly interface for monitoring security events and incidents. Thorough testing and validation will be conducted to ensure the effectiveness and reliability of the SIEM system. Comprehensive documentation and training will be provided to facilitate knowledge sharing and capacity building within the organization.

3.1 Approach

The approach for this project involves conducting a thorough analysis of the organization's security requirements to identify specific needs and objectives for the SIEM system. The architecture of the SIEM system, including components of the ELK stack, Beats, and Wazuh, will be designed to collect, store, and analyze security-related data. Beats will be configured to collect data from various sources, such as logs, events, and network traffic, and ingest them into Elasticsearch for storage and analysis. Wazuh will enhance security monitoring capabilities, including log analysis, anomaly detection, and integration with threat intelligence feeds. Custom dashboards and visualizations will be developed in Kibana to provide a user-friendly interface for monitoring security events and incidents. Thorough testing and validation will be conducted to ensure the effectiveness of the SIEM system in detecting and responding to security threats. Comprehensive documentation and training will be provided to facilitate knowledge sharing and capacity building within the organization. The SIEM system will be deployed in the organization's environment and monitored to ensure it meets security requirements and objectives.

3.2 Tools

The SIEM system will be designed and implemented using the ELK stack (Elasticsearch, Logstash, and Kibana), Beats, and Wazuh. These tools will provide the necessary capabilities for collecting, storing, analyzing, and visualizing security-related data, enhancing the organization's security posture.

3.2.1 Elasticsearch

Elasticsearch will serve as the core component for storing and indexing security-related data in the SIEM system. It will be configured to handle the high volume and diversity of data generated by various sources, including logs, events, and network traffic. Elasticsearch's distributed architecture and scalability features will ensure efficient storage and retrieval of data, even as the volume of data grows.

The data ingested into Elasticsearch will be indexed and analyzed to enable quick and easy search capabilities. Elasticsearch's powerful search and analytics capabilities will allow security teams to query and analyze security data in real-time, providing valuable insights into potential threats and vulnerabilities.

Additionally, Elasticsearch's integration with other components of the ELK stack, such as Logstash and Kibana, will enable seamless data flow and visualization of security-related data. This integration will provide security teams with a comprehensive view of security events and incidents, enabling them to respond quickly and effectively to security threats.

Overall, Elasticsearch will play a critical role in the SIEM system by providing a scalable and efficient storage and search engine for security-related data, enhancing the organization's ability to monitor, detect, and respond to security incidents.

3.2.2 Logstash

Logstash will serve as a critical component in the SIEM system, functioning as a data processing pipeline to ingest, transform, and enrich security-related data before it is indexed in Elasticsearch. It will be configured to collect data from various sources, including logs, events, and network traffic, using a wide range of input plugins to accommodate diverse data formats and sources.

Once the data is ingested, Logstash will apply filters and transformations to standardize the data format, enrich it with additional contextual information, and enhance its quality. These filters and transformations will include parsing log messages, extracting relevant fields, and performing data enrichment using external data sources or lookup tables.

Logstash's flexibility and scalability will be leveraged to ensure that it can handle the diverse data sources and processing requirements of the SIEM system. It will be configured to scale horizontally to accommodate increases in data volume, ensuring that data ingestion and processing remain efficient and responsive.

By effectively ingesting, transforming, and enriching security-related data, Logstash will play a crucial role in enabling the SIEM system to provide accurate and actionable insights into potential security threats and vulnerabilities. Its seamless integration with Elasticsearch and Kibana will further enhance the overall effectiveness and usability of the SIEM system.

3.2.3 Beats

Beats will play a crucial role in the SIEM system as lightweight data shippers designed to efficiently collect security-related data from various sources and send it to the central SIEM platform for further processing and analysis. Different Beats modules will be configured to collect data from specific sources, such as Filebeat for log files, Packetbeat for network traffic, and Winlogbeat for Windows event logs.

Filebeat will be configured to collect log files from servers, applications, and other systems, providing valuable insights into system and application activities. Packetbeat will capture network traffic data, allowing security teams to monitor network communications and detect potential security threats. Winlogbeat will collect Windows event logs, providing visibility into system and application events on Windows machines.

Beats' lightweight nature and low resource consumption make it ideal for collecting data from endpoints and other sources without impacting system performance. Beats will be configured to send data securely to the SIEM system using encrypted connections, ensuring the confidentiality and integrity of the data in transit.

By effectively collecting and forwarding security-related data to the SIEM system, Beats will enable security teams to monitor, detect, and respond to security incidents in real-time, enhancing the organization's overall cybersecurity posture.

3.2.4 Kibana

Kibana will serve as the primary visualization and dashboarding tool for the SIEM system, providing security teams with a user-friendly interface to explore, visualize, and analyze security-related data. Custom dashboards will be created in Kibana to display key security metrics, trends, and alerts, enabling security teams to quickly identify and respond to security incidents.

Kibana's interactive visualizations, such as line charts, bar charts, and pie charts, will be used to present security data in a meaningful and easy-to-understand format. These visualizations will allow security teams to track security events over time, identify patterns and anomalies, and gain insights into potential security threats.

In addition to visualizations, Kibana's powerful search capabilities will enable security teams to query and filter security data based on various criteria, such as time range, severity level, and source. This will allow security teams to quickly locate specific security events and investigate them further.

Kibana will also be used to create and manage alerts based on predefined conditions. These alerts will notify security teams in real-time when certain security events occur, enabling them to take immediate action to mitigate potential threats.

Overall, Kibana's capabilities as a visualization and dashboarding tool will enhance the organization's ability to gain actionable insights from the SIEM system's data, ultimately improving its cybersecurity posture.

3.2.5 Wazuh

Wazuh will be integrated into the SIEM system as a critical component to enhance security monitoring capabilities. Wazuh's advanced features, including log analysis, anomaly detection, and integration with threat intelligence feeds, will provide the SIEM system with additional layers of security and threat detection.

Wazuh's agent-based architecture will be deployed on endpoints across the organization's network to collect and analyze security-related data. The agents will monitor log files, system configurations, file integrity, and other aspects of endpoint security to provide real-time visibility into security events and incidents. This real-time monitoring capability will enable security teams to quickly detect and respond to security threats, minimizing the impact of potential breaches.

One of the key features of Wazuh is its ruleset, which is regularly updated with the latest threat intelligence. These rules define specific conditions that, when met, indicate a potential security threat. By leveraging this ruleset, Wazuh can detect known and emerging security threats, including malware infections, unauthorized access attempts, and suspicious network activity.

In addition to its detection capabilities, Wazuh can also integrate with external threat intelligence feeds to enhance its ability to detect and respond to security threats. This integration allows Wazuh to correlate security events with known indicators of compromise (IOCs), providing security teams with additional context to investigate and respond to security incidents.

Overall, Wazuh's integration into the SIEM system will provide the organization with a comprehensive security monitoring solution, helping to improve its overall cybersecurity posture and protect against a wide range of security threats.

CHAPTER – 4

PROJECT WORK PART-I

4.1 Project Research, Brainstorming

This section provides an in-depth look at the extensive research and brainstorming process undertaken to conceptualize and develop the Security Information and Event Management (SIEM) system using the ELK stack, Beats, and Wazuh. The initial research phase involved a comprehensive analysis of the organization's existing security infrastructure, policies, and practices to identify gaps and areas for improvement. This analysis helped define the specific security requirements that the SIEM system needed to address.

A series of brainstorming sessions were then conducted to explore different design options and implementation strategies for the SIEM system. These sessions involved key stakeholders from the IT, security, and executive teams, as well as external consultants and experts in the field of cybersecurity. The brainstorming sessions were structured to encourage creative thinking and collaboration, resulting in a wide range of ideas and concepts being discussed.

The brainstorming sessions covered various aspects of the SIEM system, including the architecture, data collection methods, alerting mechanisms, and user interface design. Different scenarios and use cases were considered to ensure that the SIEM system would be flexible and scalable enough to meet the organization's current and future security needs. Feedback from these sessions was used to refine the design and implementation plan for the SIEM system.

Collaboration with external experts and consultants provided additional insights and recommendations based on their experience with similar projects. Their input helped validate the design decisions and ensure that the SIEM system would adhere to industry best practices and standards. Additionally, feedback from stakeholders and experts helped build consensus and alignment on the project goals and objectives.

In conclusion, the project research and brainstorming process played a crucial role in shaping the design and development of the SIEM system, ensuring that it was well-aligned with the organization's security requirements and capable of providing effective security monitoring and threat detection capabilities.

4.1.1 Data Collection

This section provides a detailed overview of the data collection mechanisms implemented in the SIEM system using Beats and Logstash. The primary goal of data collection is to gather security-related data from various sources across the organization's IT infrastructure and feed it into the SIEM system for further analysis and monitoring.

Beats, as lightweight data shippers, are deployed on endpoints, servers, and network devices to collect data in near real-time. Different Beats modules, such as Filebeat for log files, Packetbeat for network traffic, and Winlogbeat for Windows event logs, are configured to collect data from specific sources. This ensures that a wide range of security-relevant data is captured, providing comprehensive visibility into the organization's IT environment.

Logstash, on the other hand, serves as a data processing pipeline that preprocesses and enriches the collected data before indexing it in Elasticsearch. Logstash filters are used to parse and structure the data, extract relevant information, and enrich it with additional context, such as geolocation data or user information. This preprocessing step is crucial for ensuring that the data is in a standardized format and contains the necessary information for analysis.

Scalability and performance considerations are also taken into account during the data collection process. Beats and Logstash are configured to handle the high volume of data generated by the organization's IT infrastructure efficiently. This includes configuring load balancing and failover mechanisms to ensure continuous data collection even in the event of server failures or network issues.

Overall, the data collection process is critical for ensuring that the SIEM system has access to timely and accurate security-related data from across the organization's IT environment. By effectively collecting and preprocessing this data, the SIEM system can provide security teams with the information they need to detect and respond to security threats effectively.

4.1.2 Normalization and Parsing

This section provides a detailed overview of the normalization and parsing process used to prepare the collected data for indexing into Elasticsearch. Normalization involves standardizing data formats and structures to ensure consistency and compatibility with the SIEM system. This includes standardizing timestamps to a common format, converting field

names to a consistent naming convention, and normalizing data types to ensure uniformity across the dataset.

Parsing, on the other hand, involves breaking down the collected data into its component parts to extract relevant information and enrich it with additional context. This process is crucial for making the data usable for analysis and visualization. Logstash provides a range of parsing capabilities through its filters, allowing for the extraction of specific fields, the conversion of data types, and the enrichment of data with external sources, such as threat intelligence feeds or geolocation data.

Custom parsing rules and filters are often used to handle specific data formats and sources that may not be supported out-of-the-box by Logstash. For example, custom Grok patterns may be created to parse log messages from custom applications, or custom scripts may be used to enrich data with information from external APIs. These customizations ensure that the data is structured and formatted correctly for analysis, enabling security teams to effectively monitor and detect security threats.

Overall, the normalization and parsing process is critical for ensuring that the collected data is in a usable format for analysis and visualization in the SIEM system. By standardizing and enriching the data, security teams can gain valuable insights into their organization's security posture and respond effectively to security incidents.

4.1.3 Correlation and Analysis

This section will delve into the correlation and analysis process employed to identify patterns and anomalies in the security-related data stored in Elasticsearch. Correlation involves correlating data from multiple sources to detect complex security incidents that may span across different systems or occur over an extended period. The section will discuss the use of correlation rules and machine learning algorithms to detect and alert on potential security threats.

Analysis, on the other hand, involves applying statistical and machine learning techniques to the collected data to identify trends, outliers, and potential security risks. This includes analyzing the frequency and severity of security events, detecting changes in user behavior or system activity, and identifying indicators of compromise (IOCs) that may indicate a security breach. The section will also cover the use of dashboards and visualizations in Kibana to present the analyzed data in a meaningful and actionable way.

Additionally, the section will discuss the importance of context in correlation and analysis,

including the use of threat intelligence feeds and external data sources to enrich the analysis with additional context about known threats and vulnerabilities. By correlating and analyzing data in this way, security teams can gain a comprehensive view of their organization's security posture and respond quickly and effectively to security incidents.

4.1.4 Alerting and Notification

This section will detail the alerting and notification mechanisms implemented in the SIEM system to notify security teams of potential security incidents. Alerts are generated based on predefined correlation rules and analysis of security-related data stored in Elasticsearch. The section will cover the configuration of alerting rules, the escalation process for alerts, and the integration of alerting mechanisms with external communication channels.

Alerting rules will be defined to trigger alerts based on specific conditions or patterns identified in the data. These rules will be customized to match the organization's security requirements and compliance standards. The escalation process for alerts will ensure that critical alerts are escalated to higher-level security personnel for immediate action, while less critical alerts are handled by lower-level analysts.

Integration with external communication channels, such as email, SMS, or chat platforms, will enable security teams to receive alerts in real-time and take immediate action. The section will also discuss the use of custom scripts and automation tools to streamline the alerting and notification process, ensuring that security teams can respond quickly and effectively to potential security incidents.

Additionally, the section will cover the importance of alert validation and tuning to reduce false positives and ensure that security teams are alerted only to genuine security threats. Regular review and refinement of alerting rules will be essential to maintain the effectiveness of the alerting and notification mechanisms over time.

4.1.5 Incident Response and Workflow

This section will outline the incident response process implemented in the SIEM system to effectively detect, respond to, and mitigate security incidents. It will cover the development of incident response workflows, the integration of the SIEM system with incident response tools and processes, and the coordination of incident response activities across different teams.

The incident response process will be based on established frameworks, such as the NIST Cybersecurity Framework or the SANS Incident Handling Process, tailored to fit the organization's specific security requirements. The section will discuss the different phases of the incident response process, including preparation, detection, containment, eradication, and recovery, and the role of the SIEM system in each phase.

The SIEM system will be integrated with incident response tools, such as ticketing systems, case management tools, and forensic analysis tools, to facilitate the incident response process. Automated workflows will be developed to streamline the handling of security incidents, from initial detection to final resolution, ensuring that incidents are addressed in a timely and effective manner.

The section will also cover the coordination of incident response activities across different teams, including IT, security, legal, and executive teams. Communication and collaboration between these teams will be essential for effectively managing security incidents and minimizing the impact on the organization.

Overall, the incident response process and workflow implemented in the SIEM system will ensure that security incidents are detected, responded to, and mitigated in a systematic and efficient manner, reducing the risk of security breaches and maintaining the organization's security posture.

4.1.6 Compliance and Reporting

This section will provide an in-depth overview of how the SIEM system addresses compliance requirements and facilitates reporting to demonstrate adherence to these requirements. The SIEM system will be designed to integrate with various compliance frameworks, such as the Payment Card Industry Data Security Standard (PCI DSS), the Health Insurance Portability and Accountability Act (HIPAA), and the General Data Protection Regulation (GDPR), among others.

Integration with these frameworks will allow the SIEM system to collect, monitor, and analyze security-related data in a manner that aligns with the specific requirements and guidelines outlined in each framework. This will include monitoring access controls, logging and monitoring, and incident response, among other security-related activities, to ensure that the organization remains compliant with applicable regulations.

The SIEM system will also be equipped with robust reporting capabilities to generate compliance reports that demonstrate adherence to these frameworks. These reports will provide detailed insights into the organization's security posture, including metrics on security events, incident response times, and policy violations. Automated reporting features will streamline the process of generating and distributing compliance reports, ensuring that stakeholders receive timely and accurate information.

Furthermore, the SIEM system will automate compliance checks by continuously monitoring security controls and comparing them against the requirements outlined in the compliance frameworks. Any deviations or non-compliant activities will trigger alerts and notifications,

allowing security teams to take corrective action promptly.

The use of dashboards and visualizations will enhance the organization's ability to monitor compliance status in real-time, providing a comprehensive view of its adherence to various regulatory requirements. This will enable security teams to proactively address compliance issues and mitigate potential risks.

Overall, the SIEM system's compliance and reporting capabilities will play a crucial role in helping the organization maintain a strong security posture and meet its regulatory obligations.

4.1.7 Integration and Orchestration

This section will provide a comprehensive overview of the integration and orchestration capabilities of the SIEM system, highlighting its ability to seamlessly integrate with a variety of security tools and systems to enhance its functionality and automate security operations.

The SIEM system will be integrated with threat intelligence feeds from reputable sources to enrich security data and improve threat detection capabilities. By integrating threat intelligence feeds, the SIEM system will be able to correlate security events with known indicators of compromise (IOCs) and identify potential threats in real-time.

Additionally, the SIEM system will be integrated with endpoint detection and response (EDR) systems to extend its visibility into endpoint security. This integration will allow the SIEM system to collect and analyze endpoint data, such as process execution, file modifications, and network connections, to detect and respond to endpoint threats more effectively.

Furthermore, the SIEM system will be integrated with security information sharing platforms, such as Information Sharing and Analysis Centers (ISACs), to collaborate with other organizations and share threat intelligence. This integration will enable the SIEM system to benefit from shared threat intelligence and enhance its ability to detect and respond to emerging threats.

The section will also discuss the use of orchestration tools, such as SOAR (Security Orchestration, Automation, and Response) platforms, to automate incident response processes and streamline security operations. By orchestrating incident response workflows, the SIEM system will be able to automatically investigate, contain, and remediate security incidents, reducing the response time and minimizing the impact of security breaches.

Overall, the integration and orchestration capabilities of the SIEM system will enhance its effectiveness in detecting and responding to security threats, ensuring that the organization's security posture remains strong and resilient.

4.1.8 Scalability and Performance

This section provides a detailed overview of the scalability and performance considerations that were incorporated into the design and implementation of the SIEM system to ensure that it can handle large volumes of data and deliver optimal performance.

Scalability Design: The SIEM system was designed with scalability in mind, using a distributed architecture to distribute the workload across multiple nodes. Elasticsearch, the core component for storing and indexing security-related data, was deployed in a clustered configuration to ensure that it can scale horizontally as the volume of data grows. Load balancing mechanisms were implemented to evenly distribute incoming data across the cluster, ensuring that no single node is overwhelmed with data.

Performance Optimization: To optimize performance, several strategies were employed. Index management techniques, such as index sharding and replica management, were implemented to ensure that data is distributed evenly across the cluster and that queries can be executed in parallel. Additionally, query optimization techniques, such as the use of filters and caching, were employed to reduce the time it takes to retrieve and analyze data.

Monitoring and Tuning: Continuous monitoring and tuning of the SIEM system were conducted to ensure that it is performing optimally. Key performance metrics, such as indexing rate, query latency, and resource utilization, were monitored using monitoring tools integrated with the ELK stack. Any performance bottlenecks or issues were identified and addressed through tuning, such as adjusting the cluster configuration or optimizing query performance.

Capacity Planning: Capacity planning was an essential aspect of ensuring scalability and performance. By forecasting future data growth and user demand, the SIEM system was designed with sufficient resources and capacity to handle future requirements. This included planning for additional nodes, storage capacity, and network bandwidth to accommodate future growth.

Testing and Validation: Finally, extensive testing and validation were conducted to ensure that the SIEM system meets the scalability and performance requirements. This included stress testing the system under heavy loads to identify its limits and ensuring that it can handle peak loads without degradation in performance.

Overall, the scalability and performance considerations incorporated into the design and implementation of the SIEM system ensure that it can effectively handle the organization's security data needs and deliver optimal performance under varying workloads.

4.1.9 Threat Intelligence Integration

This section provides an in-depth look at the integration of threat intelligence feeds into the SIEM system to enhance its ability to detect and respond to security threats. Threat intelligence feeds provide valuable information about known threats, including indicators of compromise (IOCs), malware signatures, and malicious IP addresses, which can help the SIEM system identify and mitigate security incidents.

Types of Threat Intelligence Feeds: The SIEM system integrates with various types of threat intelligence feeds, including commercial feeds, open-source feeds, and proprietary feeds from security vendors. These feeds provide real-time information about emerging threats and known attack vectors, helping the SIEM system stay updated with the latest threat intelligence.

Integration Process: The integration process involves configuring the SIEM system to ingest threat intelligence feeds from external sources. This may require setting up data connectors or APIs to fetch threat intelligence data and integrating it into the SIEM system's data pipeline. The integration process also includes mapping threat intelligence data to relevant security events and indicators within the SIEM system for correlation and analysis.

Impact on Threat Detection: The integration of threat intelligence feeds significantly enhances the SIEM system's threat detection capabilities. By correlating security events with threat intelligence data, the SIEM system can identify patterns and anomalies that may indicate a security threat. This allows security teams to respond quickly and effectively to potential security incidents, minimizing the impact on the organization.

Enrichment of Security Data: Threat intelligence feeds also enrich security data by providing additional context about known threats. This enrichment helps security teams prioritize and respond to security incidents based on the level of risk posed by the threat. For example, if an IP address is identified as malicious in a threat intelligence feed, the SIEM system can automatically block traffic from that IP address to prevent a potential attack.

Overall, the threat intelligence feeds into the SIEM system plays a crucial role in enhancing its threat detection capabilities and improving the organization's overall security posture.

4.1.10 Continuous Monitoring and Improvement

This section provides a detailed overview of the strategies and processes implemented to continuously monitor and improve the effectiveness of the SIEM system in detecting and responding to security threats.

Key Performance Indicators (KPIs) and Metrics: The SIEM system is monitored using a set of KPIs and metrics to measure its performance and effectiveness. Key metrics include the number of security events detected, the percentage of false positives, the time to detect and respond to security incidents, and the overall coverage of security monitoring. These metrics are regularly reviewed to identify trends and areas for improvement.

Regular Audits and Reviews: Regular audits and reviews are conducted to assess the SIEM system's performance and identify areas for improvement. These audits include reviewing the configuration of the SIEM system, analyzing the effectiveness of correlation rules and alerting mechanisms, and evaluating the SIEM system's ability to detect and respond to emerging threats. Based on the findings of these audits, corrective actions are taken to improve the SIEM system's effectiveness.

Implementation of Best Practices and Lessons Learned: Best practices and lessons learned from security incidents are incorporated into the SIEM system's operations to enhance its capabilities. This includes updating correlation rules based on new threat intelligence, refining alerting mechanisms to reduce false positives, and enhancing the SIEM system's integration with other security tools and systems. By continuously improving the SIEM system based on best practices and lessons learned, its effectiveness in detecting and responding to security threats is enhanced.

Training and Development: Security teams responsible for managing the SIEM system undergo regular training and development to stay updated with the latest security trends and technologies. This includes training on new features and capabilities of the SIEM system, as well as general security best practices. By ensuring that security teams are well-trained and knowledgeable, the effectiveness of the SIEM system is maximized.

Overall, the continuous monitoring and improvement strategies implemented ensure that the SIEM system remains effective in detecting and responding to security threats, helping to protect the organization's assets and data from cyber threats.

4.2 Use Case

The practical application of the SIEM system in detecting and responding to a security threat. The use case will outline the context of the scenario, the security events detected by the SIEM system, the analysis performed, and the actions taken to mitigate the threat. The use case will highlight the effectiveness of the SIEM system in enhancing the organization's security posture and reducing the impact of security incidents.

4.2.1 Detecting compromised user credentials

Context: In this use case scenario, the organization's security team is using the SIEM system to monitor and detect potential security threats. One of the critical threats they are concerned about is the compromise of user credentials, which could lead to unauthorized access to sensitive systems and data.

Detection: The SIEM system is configured to monitor login events across the organization's network and systems. It uses correlation rules to detect unusual login patterns, such as multiple failed login attempts or login attempts from unusual locations. These events are correlated with threat intelligence feeds to identify known malicious IP addresses or patterns associated with credential stuffing attacks.

Analysis: When the SIEM system detects suspicious login activity, it triggers an alert for the security team to investigate. The security team analyzes the alert to determine the severity of the threat and the potential impact on the organization. They review the affected user accounts and the source of the login attempts to assess the risk level.

Response: Based on the analysis, the security team takes appropriate actions to mitigate the threat. This may include resetting compromised user passwords, blocking suspicious IP addresses, or implementing multi-factor authentication (MFA) for affected accounts. The SIEM system logs these actions for auditing and compliance purposes.

Outcome: By using the SIEM system to detect and respond to compromised user credentials, the organization is able to prevent unauthorized access to its systems and data. This helps protect sensitive information and maintain the integrity of its IT infrastructure.

Lessons Learned: The organization uses this use case to review its security policies and practices related to user authentication. It implements additional security measures, such as regular password changes and user awareness training, to reduce the risk of future credential compromise incidents.

This use case demonstrates how the SIEM system can be used to effectively detect and respond

to security threats related to compromised user credentials, helping organizations enhance their security posture and protect against unauthorized access.

4.2.2 Tracking system changes

Context: In this use case scenario, the organization's IT team is using the SIEM system to monitor and track changes to critical systems and configurations. The goal is to ensure that any unauthorized or unexpected changes are detected and investigated promptly to prevent security incidents and maintain system integrity.

Detection: The SIEM system is configured to monitor system logs and audit trails for changes to files, configurations, and permissions. It uses predefined correlation rules to detect changes that deviate from normal or expected patterns. This includes changes to system files, user accounts, group memberships, and network configurations.

Analysis: When the SIEM system detects a suspicious change, it triggers an alert for the IT team to investigate. The IT team analyzes the alert to determine the nature of the change, the potential impact on system security, and the responsible user or application. They review the change logs and audit trails to understand the sequence of events leading to the change.

Response: Based on the analysis, the IT team takes appropriate actions to mitigate the impact of the change. This may include reverting the change, updating security configurations, or implementing additional security measures to prevent similar changes in the future. The SIEM system logs these actions for auditing and compliance purposes.

Outcome: By using the SIEM system to track system changes, the organization is able to detect and respond to unauthorized or unexpected changes promptly. This helps prevent security incidents and ensures the integrity and availability of its IT systems.

Lessons Learned: The organization uses this use case to review its change management processes and policies. It implements stricter controls and monitoring mechanisms to reduce the risk of unauthorized changes and improve overall system security.

This use case demonstrates how the SIEM system can be used to effectively track system changes and maintain the integrity and security of critical IT systems.

4.2.3 Detecting unusual behavior on privileged accounts

Context: In this use case scenario, the organization's security team is using the SIEM system to monitor and detect unusual behavior on privileged accounts. Privileged accounts have elevated access rights and pose a higher security risk if compromised. The goal is to detect and respond to any suspicious activity on these accounts to prevent unauthorized access and data breaches.

Detection: The SIEM system is configured to monitor privileged account activity, including login attempts, file access, and system configuration changes. It uses behavioral analytics and machine learning algorithms to establish a baseline of normal behavior for each privileged account. Any deviations from this baseline, such as unusual login times or access to sensitive files, trigger an alert for further investigation.

Analysis: When the SIEM system detects unusual behavior on a privileged account, it triggers an alert for the security team to investigate. The security team analyzes the alert to determine the nature of the activity, the potential impact on system security, and the responsible user or application. They review the privileged account's activity logs and audit trails to understand the sequence of events leading to the unusual behavior.

Response: Based on the analysis, the security team takes appropriate actions to mitigate the impact of the unusual behavior. This may include disabling the privileged account, revoking access rights, or implementing additional security measures to prevent unauthorized access. The SIEM system logs these actions for auditing and compliance purposes.

Outcome: By using the SIEM system to detect and respond to unusual behavior on privileged accounts, the organization is able to prevent unauthorized access and data breaches. This helps protect sensitive information and maintain the integrity of its IT systems.

Lessons Learned: The organization uses this use case to review its privileged account management practices and policies. It implements stricter controls and monitoring mechanisms to reduce the risk of unauthorized access and improve overall system security.

This use case demonstrates how the SIEM system can be used to effectively detect and respond to unusual behavior on privileged accounts, helping organizations enhance their security posture and protect against unauthorized access.

4.2.4 Secure cloud-based applications

Context: In this use case scenario, the organization is using cloud-based applications to store and process sensitive data. The security team is using the SIEM system to monitor and secure these applications against security threats, ensuring the confidentiality, integrity, and availability of the data.

Detection: The SIEM system is configured to monitor user access to cloud-based applications, including login attempts, file uploads and downloads, and data access patterns. It uses correlation rules and anomaly detection algorithms to detect suspicious activity, such as unauthorized access or data exfiltration, and trigger alerts for further investigation.

Analysis: When the SIEM system detects suspicious activity on a cloud-based application, it triggers an alert for the security team to investigate. The security team analyzes the alert to determine the nature of the activity, the potential impact on data security, and the responsible user or application. They review the application's logs and audit trails to understand the sequence of events leading to the suspicious activity.

Response: Based on the analysis, the security team takes appropriate actions to mitigate the impact of the suspicious activity. This may include revoking user access, implementing additional security controls, or notifying relevant stakeholders about the incident. The SIEM system logs these actions for auditing and compliance purposes.

Outcome: By using the SIEM system to secure cloud-based applications, the organization is able to protect sensitive data from security threats and ensure compliance with data protection regulations. This helps maintain the trust of customers and stakeholders and avoid potential financial and reputational damage.

Lessons Learned: The organization uses this use case to review its cloud security policies and practices. It implements additional security measures, such as encryption, multi-factor authentication, and regular security audits, to enhance the security of its cloud-based applications.

This use case demonstrates how the SIEM system can be used to effectively secure cloud-based applications, helping organizations protect sensitive data and maintain the integrity of their IT infrastructure.

4.2.5 Phishing detection

Context: In this use case scenario, the organization's security team is using the SIEM system to detect and respond to phishing attacks. Phishing attacks are a common method used by cybercriminals to steal sensitive information, such as login credentials and financial data, by tricking users into clicking on malicious links or attachments in emails.

Detection: The SIEM system is configured to monitor email traffic and detect phishing attempts. It uses email filtering rules and machine learning algorithms to analyze email content, sender reputation, and user behavior to identify potential phishing emails. Suspicious emails are flagged and quarantined for further investigation.

Analysis: When the SIEM system detects a phishing email, it triggers an alert for the security team to investigate. The security team analyzes the email to determine its legitimacy, the potential impact of the phishing attack, and the actions required to mitigate the threat. They also review the email headers and attachments to identify any malicious links or files.

Response: Based on the analysis, the security team takes appropriate actions to mitigate the impact of the phishing attack. This may include blocking the sender's email address, quarantining the email, or notifying users about the phishing attempt. The SIEM system logs these actions for auditing and compliance purposes.

Outcome: By using the SIEM system to detect and respond to phishing attacks, the organization is able to protect sensitive information and prevent unauthorized access to its systems. This helps maintain the integrity of its IT infrastructure and avoid potential financial and reputational damage.

Lessons Learned: The organization uses this use case to review its email security policies and practices. It implements additional security measures, such as user awareness training and email authentication protocols, to enhance its ability to detect and respond to phishing attacks.

This use case demonstrates how the SIEM system can be used to effectively detect and respond to phishing attacks, helping organizations protect sensitive information and maintain the integrity of their IT infrastructure.

4.2.6 Monitoring loads and uptimes

Context: In this use case scenario, the organization's IT team is using the SIEM system to monitor the loads and uptimes of critical systems and applications. Monitoring loads helps ensure that systems are operating within acceptable limits, while monitoring uptimes helps ensure that systems are available and accessible to users.

Detection: The SIEM system is configured to collect and analyze performance metrics, such as CPU usage, memory usage, disk space, and network traffic, from critical systems and applications. It uses predefined thresholds and anomaly detection algorithms to detect abnormal loads or performance issues that may indicate a potential problem.

Analysis: When the SIEM system detects abnormal loads or performance issues, it triggers an alert for the IT team to investigate. The IT team analyzes the alert to determine the cause of the issue, the potential impact on system performance, and the actions required to mitigate the problem. They also review historical performance data to identify any trends or patterns that may help diagnose the issue.

Response: Based on the analysis, the IT team takes appropriate actions to mitigate the impact of the performance issue. This may include reallocating resources, optimizing configurations, or implementing performance tuning measures. The SIEM system logs these actions for auditing and compliance purposes.

Outcome: By using the SIEM system to monitor loads and uptimes, the organization is able to proactively identify and address performance issues before they impact system availability and user experience. This helps ensure that critical systems and applications are operating efficiently and effectively.

Lessons Learned: The organization uses this use case to review its monitoring practices and performance management strategies. It implements additional monitoring tools and practices, such as automated alerting and capacity planning, to enhance its ability to monitor loads and uptimes effectively.

This use case demonstrates how the SIEM system can be used to effectively monitor loads and uptimes of critical systems and applications, helping organizations ensure optimal performance and availability.

4.2.7 Log Management

Context: In this use case scenario, the organization's security team is using the SIEM system to manage and analyze logs from various sources, including network devices, servers, and applications. Effective log management is crucial for maintaining visibility into security events and ensuring compliance with regulatory requirements.

Collection: The SIEM system is configured to collect logs from all critical systems and applications. This includes syslog messages, event logs, and application logs. The logs are normalized and parsed to ensure consistency and ease of analysis.

Analysis: Once the logs are collected, the SIEM system analyzes them to identify security events and anomalies. It uses correlation rules and threat intelligence feeds to correlate events across different sources and detect patterns indicative of a security incident.

Storage: The SIEM system stores the logs in a secure and tamper-evident manner to comply with regulatory requirements. Logs are encrypted both at rest and in transit to ensure data integrity and confidentiality.

Retrieval: Security analysts can easily retrieve and search logs using the SIEM system's search and query capabilities. This allows them to quickly investigate security incidents and respond to threats.

Compliance: The SIEM system helps the organization comply with regulatory requirements related to log management, such as PCI DSS, HIPAA, and GDPR. It generates compliance reports and alerts security teams to any non-compliance issues.

Integration: The SIEM system integrates with other security tools and systems, such as intrusion detection systems (IDS) and endpoint detection and response (EDR) systems, to provide a comprehensive view of the organization's security posture.

Outcome: By using the SIEM system for log management, the organization is able to improve its ability to detect and respond to security threats, ensure compliance with regulatory requirements, and maintain visibility into its IT environment.

Lessons Learned: The organization uses this use case to review its log management practices and identify areas for improvement. It implements additional logging and monitoring controls to enhance its security posture and reduce the risk of security incidents.

This use case demonstrates how the SIEM system can be used for effective log management, helping organizations improve their security posture and ensure compliance with regulatory requirements.

4.2.8 SIEM for GDPR, HIPAA, or PCI compliance

Context: In this use case scenario, the organization is subject to regulatory requirements such as the General Data Protection Regulation (GDPR), Health Insurance Portability and Accountability Act (HIPAA), or Payment Card Industry Data Security Standard (PCI DSS). The organization's security team is using the SIEM system to ensure compliance with these regulations by monitoring and securing sensitive data and systems.

GDPR Compliance: The SIEM system is configured to monitor and protect personal data as required by GDPR. This includes monitoring access to personal data, detecting and responding to data breaches, and ensuring the security of data processing activities.
The SIEM system helps the organization comply with GDPR's requirements for data protection impact assessments (DPIAs), data breach notification, and data subject access requests (DSARs) by providing visibility into data processing activities and security incidents.

HIPAA Compliance: The SIEM system helps the organization comply with HIPAA's requirements for protecting electronic protected health information (ePHI). This includes monitoring access to ePHI, detecting and responding to security incidents, and implementing safeguards to ensure the confidentiality, integrity, and availability of ePHI.
The SIEM system assists with HIPAA's requirements for risk analysis, risk management, and audit controls by providing real-time visibility into security events and generating audit trails for compliance reporting.

PCI DSS Compliance: The SIEM system helps the organization comply with PCI DSS's requirements for securing payment card data and systems. This includes monitoring access to cardholder data, detecting and responding to security incidents, and implementing controls to protect cardholder data.
The SIEM system assists with PCI DSS's requirements for log management, access control, and security monitoring by aggregating and analyzing log data from critical systems and applications.

Outcome: By using the SIEM system for GDPR, HIPAA, or PCI compliance, the organization is able to ensure the security and privacy of sensitive data, protect against data breaches and security incidents, and demonstrate compliance with regulatory requirements.
The SIEM system provides visibility into security events and activities, facilitates incident response and forensic investigations, and generates reports for compliance audits and regulatory reporting.

Lessons Learned: The organization uses this use case to review its compliance efforts and identify areas for improvement. It implements additional security controls, enhances monitoring and reporting capabilities, and provides ongoing training and awareness programs to maintain compliance with regulatory requirements.

4.2.9 Threat Hunting

Context: In this use case scenario, the organization's security team is using the SIEM system for proactive threat hunting. Threat hunting is a proactive approach to cybersecurity that involves actively searching for signs of malicious activity within the organization's IT environment.

Hypothesis Generation: The security team starts by generating hypotheses about potential threats based on threat intelligence, security trends, and known attack patterns. These hypotheses guide their search for suspicious activity and help prioritize their efforts.

Data Collection and Analysis: The SIEM system is configured to collect and analyze a wide range of data sources, including logs, network traffic, and endpoint telemetry. The security team uses the SIEM system's analytics and visualization tools to identify anomalies, patterns, and indicators of compromise (IOCs) that may indicate a security threat.

Threat Hunting Campaigns: The security team conducts regular threat hunting campaigns based on their hypotheses. They use the SIEM system to search for specific IOCs, such as IP addresses, domain names, or file hashes, that may indicate malicious activity. They also look for unusual patterns of behavior, such as spikes in network traffic or unauthorized access attempts.

Incident Response: If the security team discovers evidence of a security threat during their threat hunting activities, they initiate an incident response process. This may involve isolating affected systems, removing malware, and implementing additional security controls to prevent further compromise.

Outcome: By using the SIEM system for threat hunting, the organization is able to proactively identify and respond to security threats before they cause damage. Threat hunting helps the organization stay ahead of attackers and improve its overall security posture.

Lessons Learned: The organization uses this use case to refine its threat hunting techniques and improve its ability to detect and respond to emerging threats. They continuously update their hypotheses based on new threat intelligence and security research to stay ahead of evolving threats.

This use case demonstrates how the SIEM system can be used for proactive threat hunting, helping organizations identify and mitigate security threats before they cause harm.

4.2.10 SIEM for automation

Context: In this use case scenario, the organization's security team is using the SIEM system to automate security processes and responses. Automation helps improve the efficiency and effectiveness of security operations by reducing manual intervention and response times.

Automated Alerting: The SIEM system is configured to automatically generate alerts for security events based on predefined rules and thresholds. These alerts are prioritized based on severity and relevance, allowing the security team to focus on the most critical threats.

Automated Response: The SIEM system is integrated with other security tools and systems, such as firewalls, intrusion detection systems (IDS), and endpoint security solutions, to enable automated responses to security incidents. For example, the SIEM system can automatically block suspicious IP addresses, quarantine infected devices, or update firewall rules in response to detected threats.

Orchestration: The SIEM system acts as an orchestration platform, coordinating the actions of different security tools and systems to respond to security incidents. This orchestration capability helps streamline incident response workflows and ensure a coordinated and effective response to threats.

Playbooks: The security team develops playbooks that define automated response actions for different types of security incidents. These playbooks are integrated into the SIEM system and can be triggered automatically based on the nature and severity of the incident.

Outcome: By using the SIEM system for automation, the organization is able to improve its security posture by reducing response times, minimizing human error, and ensuring a consistent and effective response to security incidents. Automation also allows the security team to focus on more strategic tasks, such as threat hunting and security analysis.

Lessons Learned: The organization uses this use case to identify opportunities for further automation and optimization of its security operations. They continuously refine their playbooks and automation processes based on feedback and lessons learned from previous incidents.

This use case demonstrates how the SIEM system can be used for automation, helping organizations improve the efficiency and effectiveness of their security operations.

CHAPTER – 5

PROJECT WORK PART-II

5.1 VM Setup

Overview: We will discuss the setup of the virtual machine (VM) provided by Cyber-Dojo for our project. The VM serves as our operating environment, providing the necessary resources and tools to develop and deploy our SIEM solution.

VM Configuration: The VM is configured with the required operating system and software packages to support our SIEM implementation. This includes the installation of the ELK stack (Elasticsearch, Logstash, Kibana) for log management and analysis, as well as additional tools for data collection, normalization, and visualization.

Resource Allocation: The VM is allocated with sufficient resources, including CPU, memory, and storage, to support our SIEM operations. The resource allocation is optimized to ensure optimal performance and scalability of our SIEM solution.

Network Configuration: The VM is configured with the necessary network settings to allow communication with external systems and services. This includes configuring firewall rules, network interfaces, and DNS settings to ensure connectivity and security.

Access Control: Access to the VM is restricted to authorized personnel only. This includes implementing strong authentication mechanisms, such as SSH keys or multi-factor authentication, and regular monitoring of access logs for suspicious activity.

Backup and Recovery: The VM is configured with backup and recovery mechanisms to protect against data loss and ensure business continuity. This includes regular backups of critical data and configurations, as well as testing of recovery procedures to ensure their effectiveness.

Monitoring and Management: The VM is monitored and managed using tools and scripts to ensure its health and availability. This includes monitoring resource usage, system logs, and security events, and taking proactive measures to address any issues that may arise.

Conclusion: The VM setup provides us with a stable and secure operating environment for developing and deploying our SIEM solution. It serves as the foundation for our project, enabling us to focus on building a robust and effective security platform.

.

5.2 Implementation

The implementation phase of our project involves the deployment and configuration of our Security Information and Event Management (SIEM) solution using the virtual machine (VM) provided by Cyber-Dojo. We will be setting up the ELK stack, consisting of Elasticsearch, Logstash, and Kibana, to handle log management, analysis, and visualization. This phase also includes integrating data sources, configuring log parsing and enrichment, creating custom dashboards in Kibana, and setting up alerting and monitoring mechanisms. The goal of the implementation phase is to establish a robust and efficient SIEM platform that can effectively monitor and analyze security events in our IT environment.

5.2.1 Wazuh Indexer

The Wazuh Indexer plays a crucial role in the overall architecture of the Wazuh platform, responsible for indexing and storing security events received from agents. In this implementation, we focus on deploying and configuring the Wazuh Indexer component to ensure efficient event indexing and retrieval.

Deployment: The Wazuh Indexer can be deployed on a dedicated server or as part of a distributed setup, depending on the scale of the environment. For a basic setup, we deploy the Indexer on a server with sufficient resources to handle the indexing workload.

Configuration: Once deployed, the Wazuh Indexer is configured to listen for incoming security events from Wazuh agents and other sources. Configuration includes specifying the listening port, enabling encryption for data transfer, and setting up authentication mechanisms to ensure secure communication.

Indexing Rules: Indexing rules define how incoming events are processed and indexed by the Wazuh Indexer. These rules can be customized to filter events based on specific criteria, such as severity level, source IP, or event type, to optimize storage and retrieval.

Storage Optimization: To ensure efficient use of storage resources, the Wazuh Indexer configuration includes settings for data retention and indexing strategies. These settings help manage the size of the index and optimize search performance.

Search and Retrieval: With the Wazuh Indexer configured, users can search and retrieve security events using the Wazuh API or the Kibana interface. The Indexer provides fast and efficient search capabilities, allowing users to quickly identify and analyze security incidents.

Scalability: For environments with high event volumes, the Wazuh Indexer can be scaled horizontally by deploying multiple instances and using load balancers to distribute the workload. This ensures that the indexing infrastructure can handle the growing volume of security events.

Steps:

```
cyberdojo@cyberdojo:~$ sudo su
[sudo] password for cyberdojo:
root@cyberdojo:/home/cyberdojo# curl -sO https://packages.wazuh.com/4.7/wazuh-ce
rts-tool.sh
root@cyberdojo:/home/cyberdojo# curl -sO https://packages.wazuh.com/4.7/config.y
ml
root@cyberdojo:/home/cyberdojo# ls
config.yml  Documents  Music     Public     Videos
Desktop     Downloads  Pictures  Templates  wazuh-certs-tool.sh
```

For all wazuh nodes , be it server, indexer or dashboard, edit the config.yml file and enter ip addresses of other nodes.

```
root@cyberdojo:/home/cyberdojo# nano config.yml
```

```
nodes:
  # Wazuh indexer nodes
  indexer:
    - name: node-1
      ip: "192.168.1.30"
    #- name: node-2
    #  ip: "<indexer-node-ip>"
    #- name: node-3
    #  ip: "<indexer-node-ip>"

  # Wazuh server nodes
  # If there is more than one Wazuh server
  # node, each one must have a node_type
  server:
    - name: wazuh-1
      ip: "192.168.1.22"
    #  node_type: master
    #- name: wazuh-2
    #  ip: "<wazuh-manager-ip>"
    #  node_type: worker

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text^T To Spell  ^_ Go To Line
```

```
# Wazuh dashboard nodes
dashboard:
  - name: dashboard
    ip: "192.168.1.49"
```

```
root@cyberdojo:/home/cyberdojo# bash ./wazuh-certs-tool.sh -A
22/03/2024 14:44:36 INFO: Admin certificates created.
22/03/2024 14:44:36 INFO: Wazuh indexer certificates created.
22/03/2024 14:44:37 INFO: Wazuh server certificates created.
22/03/2024 14:44:37 INFO: Wazuh dashboard certificates created.
root@cyberdojo:/home/cyberdojo#
```

The next step is just for checking, its not part of the process. Only the next step.

```
root@cyberdojo:/home/cyberdojo# cd wazuh-certificates/
root@cyberdojo:/home/cyberdojo/wazuh-certificates# ls
admin-key.pem  dashboard-key.pem  node-1-key.pem  root-ca.key  wazuh-1-key.pem
admin.pem      dashboard.pem      node-1.pem      root-ca.pem  wazuh-1.pem
root@cyberdojo:/home/cyberdojo/wazuh-certificates# cd ..
```

```
root@cyberdojo:/home/cyberdojo# pwd
/home/cyberdojo
```

```
root@cyberdojo:/home/cyberdojo# tar -cvf ./wazuh-certificates.tar -C ./wazuh-cer
tificates/ .
./
./root-ca.pem
./admin.pem
./dashboard-key.pem
./root-ca.key
./node-1-key.pem
./admin-key.pem
./wazuh-1-key.pem
./node-1.pem
./dashboard.pem
./wazuh-1.pem
root@cyberdojo:/home/cyberdojo# ls
config.yml  Downloads  Public     wazuh-certificates
Desktop     Music      Templates  wazuh-certificates.tar
Documents   Pictures   Videos     wazuh-certs-tool.sh
```

```
root@cyberdojo:/home/cyberdojo# rm -rf ./wazuh-certificates
root@cyberdojo:/home/cyberdojo# ls
config.yml  Downloads  Public     wazuh-certificates.tar
Desktop     Music      Templates  wazuh-certs-tool.sh
Documents   Pictures   Videos
root@cyberdojo:/home/cyberdojo#
```

Sent to wazuh server

```
root@cyberdojo:/home/cyberdojo# scp wazuh-certificates.tar cyberdojo@192.168.1.2
2:/home/cyberdojo
The authenticity of host '192.168.1.22 (192.168.1.22)' can't be established.
ECDSA key fingerprint is SHA256:xU3eX9J1/6zKBvyK1pawnE2YJloMVI8AqzkcmNCaS5k.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.1.22' (ECDSA) to the list of known hosts.
cyberdojo@192.168.1.22's password:
wazuh-certificates.tar                        100%   30KB  25.5MB/s   00:00
root@cyberdojo:/home/cyberdojo#
```

Sent to wazuh dashboard

```
root@cyberdojo:/home/cyberdojo# scp wazuh-certificates.tar cyberdojo@192.168.1.4
9:/home/cyberdojo
The authenticity of host '192.168.1.49 (192.168.1.49)' can't be established.
ECDSA key fingerprint is SHA256:xU3eX9J1/6zKBvyK1pawnE2YJloMVI8AqzkcmNCaS5k.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.49' (ECDSA) to the list of known hosts.
cyberdojo@192.168.1.49's password:
wazuh-certificates.tar                        100%   30KB  15.4MB/s   00:00
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# apt-get install debconf adduser procps
Reading package lists... Done
Building dependency tree
Reading state information... Done
adduser is already the newest version (3.118ubuntu2).
adduser set to manually installed.
debconf is already the newest version (1.5.73).
debconf set to manually installed.
procps is already the newest version (2:3.3.16-1ubuntu2.3).
procps set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 57 not upgraded.
```

```
root@cyberdojo:/home/cyberdojo# apt-get update
Hit:1 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:3 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:4 http://in.archive.ubuntu.com/ubuntu focal-security InRelease [114 kB]
0% [4 InRelease 2,508 B/114 kB 2%]
```

```
root@cyberdojo:/home/cyberdojo# apt-get install gnupg apt-transport-https
Reading package lists... Done
Building dependency tree
Reading state information... Done
gnupg is already the newest version (2.2.19-3ubuntu2.2).
gnupg set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 207 not upgraded.
Need to get 1,704 B of archives.
After this operation, 162 kB of additional disk space will be used.
```

```
root@cyberdojo:/home/cyberdojo# curl -s https://packages.wazuh.com/key/GPG-KEY-W
AZUH | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/wazuh.g
pg --import && chmod 644 /usr/share/keyrings/wazuh.gpg
gpg: keyring '/usr/share/keyrings/wazuh.gpg' created
gpg: directory '/root/.gnupg' created
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key 96B3EE5F29111145: public key "Wazuh.com (Wazuh Signing Key) <support@wa
zuh.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# echo "deb [signed-by=/usr/share/keyrings/wazuh.g
pg] https://packages.wazuh.com/4.x/apt/ stable main" | tee -a /etc/apt/sources.l
ist.d/wazuh.list
deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/apt
/ stable main
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# apt-get update
0% [Connecting to in.archive.ubuntu.com]
```

```
root@cyberdojo:/home/cyberdojo# apt-get -y install wazuh-indexer
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  wazuh-indexer
0 upgraded, 1 newly installed, 0 to remove and 207 not upgraded.
Need to get 678 MB of archives.
After this operation, 969 MB of additional disk space will be used.
Get:1 https://packages.wazuh.com/4.x/apt stable/main amd64 wazuh-indexer amd64 4
.7.3-1 [678 MB]
1% [1 wazuh-indexer 10.7 MB/678 MB 2%]
```

```
root@cyberdojo:/home/cyberdojo# nano /etc/wazuh-indexer/opensearch.yml
```

```
network.host: "192.168.1.30"
node.name: "node-1"
cluster.initial_master_nodes:
- "node-1"
#- "node-2"
#- "node-3"
cluster.name: "wazuh-cluster"
#discovery.seed_hosts:
#  - "node-1-ip"
#  - "node-2-ip"
#  - "node-3-ip"
node.max_local_storage_nodes: "3"
path.data: /var/lib/wazuh-indexer
path.logs: /var/log/wazuh-indexer

plugins.security.ssl.http.pemcert_filepath: /etc/wazuh-indexer/certs/indexer.pem
plugins.security.ssl.http.pemkey_filepath: /etc/wazuh-indexer/certs/indexer-key>
plugins.security.ssl.http.pemtrustedcas_filepath: /etc/wazuh-indexer/certs/root>
plugins.security.ssl.transport.pemcert_filepath: /etc/wazuh-indexer/certs/index>
plugins.security.ssl.transport.pemkey_filepath: /etc/wazuh-indexer/certs/indexe>
                [ Read 42 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text^T To Spell  ^_ Go To Line
```

encrypt communications between the Wazuh ce

```
root@cyberdojo:/home/cyberdojo# NODE_NAME=node-1
root@cyberdojo:/home/cyberdojo# mkdir /etc/wazuh-indexer/certs
root@cyberdojo:/home/cyberdojo# tar -xf ./wazuh-certificates.tar -C /etc/wazuh-i
ndexer/certs/ ./$NODE_NAME.pem ./$NODE_NAME-key.pem ./admin.pem ./admin-key.pem
./root-ca.pem
root@cyberdojo:/home/cyberdojo# mv -n /etc/wazuh-indexer/certs/$NODE_NAME.pem /e
tc/wazuh-indexer/certs/indexer.pem
root@cyberdojo:/home/cyberdojo# mv -n /etc/wazuh-indexer/certs/$NODE_NAME-key.pe
m /etc/wazuh-indexer/certs/indexer-key.pem
root@cyberdojo:/home/cyberdojo# chmod 500 /etc/wazuh-indexer/certs
root@cyberdojo:/home/cyberdojo# chmod 400 /etc/wazuh-indexer/certs/*
root@cyberdojo:/home/cyberdojo# chown -R wazuh-indexer:wazuh-indexer /etc/wazuh-
indexer/certs
root@cyberdojo:/home/cyberdojo# rm -f ./wazuh-certificates.tar
root@cyberdojo:/home/cyberdojo# ls
config.yml  Documents  Music     Public     Videos
Desktop     Downloads  Pictures  Templates  wazuh-certs-tool.sh
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# systemctl daemon-reload
root@cyberdojo:/home/cyberdojo# systemctl enable wazuh-indexer
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-indexer.servic
e → /lib/systemd/system/wazuh-indexer.service.
root@cyberdojo:/home/cyberdojo# systemctl start wazuh-indexer
root@cyberdojo:/home/cyberdojo# systemctl status wazuh-indexer
● wazuh-indexer.service - Wazuh-indexer
     Loaded: loaded (/lib/systemd/system/wazuh-indexer.service; enabled; vendor>
     Active: active (running) since Fri 2024-03-22 15:25:22 UTC; 22s ago
       Docs: https://documentation.wazuh.com
   Main PID: 8386 (java)
      Tasks: 65 (limit: 10416)
     Memory: 1.3G
     CGroup: /system.slice/wazuh-indexer.service
             └─8386 /usr/share/wazuh-indexer/jdk/bin/java -Xshare:auto -Dopense>
```

Repeat this stage of the installation process for every Wazuh indexer node in your cluster. Then proceed with initializing your single-node or multi-node cluster in the next stage.

```
root@cyberdojo:/home/cyberdojo# /usr/share/wazuh-indexer/bin/indexer-security-in
it.sh
************************************************************************
** This tool will be deprecated in the next major release of OpenSearch **
** https://github.com/opensearch-project/security/issues/1755           **
************************************************************************
Security Admin v7
Will connect to 192.168.1.30:9200 ... done
Connected as "CN=admin,OU=Wazuh,O=Wazuh,L=California,C=US"
OpenSearch Version: 2.8.0
Contacting opensearch cluster 'opensearch' and wait for YELLOW clusterstate ...
Clustername: wazuh-cluster
Clusterstate: GREEN
Number of nodes: 1
Number of data nodes: 1
.opendistro_security index does not exists, attempt to create it ... done (0-all
 replicas)
```

Testing the cluster installation

```
root@cyberdojo:/home/cyberdojo# curl -k -u admin:admin https://192.168.1.30:9200
{
  "name" : "node-1",
  "cluster_name" : "wazuh-cluster",
  "cluster_uuid" : "sB1t8eo9RxuQ32RBG0dzvg",
  "version" : {
    "number" : "7.10.2",
    "build_type" : "rpm",
    "build_hash" : "db90a415ff2fd428b4f7b3f800a51dc229287cb4",
    "build_date" : "2023-06-03T06:24:25.112415503Z",
    "build_snapshot" : false,
    "lucene_version" : "9.6.0",
    "minimum_wire_compatibility_version" : "7.10.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "The OpenSearch Project: https://opensearch.org/"
}
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# curl -k -u admin:admin https://192.168.1.30:9200/_cat/nodes?v
ip          heap.percent ram.percent cpu load_1m load_5m load_15m node.role node.roles                                        clust
er_manager name
192.168.1.30           15          83   8    0.76    0.50     0.45 dimr      cluster_manager,data,ingest,remote_cluster_client *
          node-1
root@cyberdojo:/home/cyberdojo#
```

For filebeat, enable port 5044

```
root@cyberdojo:/home/cyberdojo# ufw allow 5044
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

Also open port 9200

```
root@cyberdojo:/home/cyberdojo# ufw allow 9200
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

Open port 55000 so that wazuh dashboard can access wazuh API on wazuh server

```
root@cyberdojo:/home/cyberdojo# ufw allow 55000
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

Wazuh agent communicates on port 1514 and 1515

```
root@cyberdojo:/home/cyberdojo# ufw allow 1514
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# ufw allow 1515
Rule added
Rule added (v6)
```

5.2.2 Wazuh Server

The Wazuh Server is a critical component in the Wazuh architecture, responsible for receiving, processing, and analyzing security event data from Wazuh agents. In this implementation, we focus on deploying and configuring the Wazuh Server to ensure effective security monitoring and threat detection.

Deployment: The Wazuh Server can be deployed on a dedicated server or as part of a distributed setup, depending on the organization's requirements. For a basic setup, we deploy the Wazuh Server on a server with sufficient resources to handle the event processing workload.

Configuration: Once deployed, the Wazuh Server is configured to listen for incoming security events from Wazuh agents and other sources. Configuration includes specifying the listening port, enabling encryption for data transfer, and setting up authentication mechanisms to ensure secure communication.

Agent Registration: Wazuh agents need to be registered with the Wazuh Server to send security event data. This involves installing the Wazuh agent on each endpoint and configuring it to communicate with the Wazuh Server using the appropriate credentials.

Ruleset Configuration: The Wazuh Server uses a ruleset to identify and categorize security events. The ruleset can be customized to include additional rules specific to the organization's security policies and compliance requirements.

Integration with SIEM: The Wazuh Server can be integrated with a Security Information and Event Management (SIEM) system, such as Elasticsearch, Logstash, and Kibana (ELK stack), for centralized log management and analysis. Integration involves configuring the Wazuh Server to forward security events to the SIEM system for further analysis.

Monitoring and Alerting: The Wazuh Server provides real-time monitoring of security events and can generate alerts based on predefined criteria. Alerts can be configured to notify security personnel via email, SMS, or other channels, allowing for prompt response to security incidents.

Scalability: For environments with high event volumes, the Wazuh Server can be scaled horizontally by deploying multiple instances and using load balancers to distribute the workload. This ensures that the server infrastructure can handle the growing volume of security events.

Steps:

```
 73  curl -sO https://packages.wazuh.com/4.7/wazuh-install.sh
 74  curl -sO https://packages.wazuh.com/4.7/config.yml
 75  apt-get install gnupg apt-transport-https
 76  apt-get update
 77  apt-get install gnupg apt-transport-https
 78  curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/wazu
gpg --import && chmod 644 /usr/share/keyrings/wazuh.gpg
 79  echo "deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/apt/ stable main" | tee -a /etc/apt/source
list.d/wazuh.list
 80  apt-get update
 81  apt-get -y install wazuh-manager
 82  systemctl daemon-reload
 83  systemctl enable wazuh-manager
 84  systemctl start wazuh-manager
 85  systemctl status wazuh-manager
 86  apt-get -y install filebeat
 87  kill -9 50676
 88  apt-get -y install filebeat
 89  systemctl status wazuh-manager
 90  curl -so /etc/filebeat/filebeat.yml https://packages.wazuh.com/4.7/tpl/wazuh/filebeat/filebeat.yml
 91  nano /etc/filebeat/filebeat.yml
 92  history
ot@cyberdojo:/home/cyberdojo#
```

For all wazuh nodes , be it server, indexer or dashboard, edit the config.yml file and enter ip addresses of other nodes.

```
root@cyberdojo:/home/cyberdojo# ls
config.yml  Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  wazuh-install.sh
root@cyberdojo:/home/cyberdojo# nano config.yml
root@cyberdojo:/home/cyberdojo#
```

```
nodes:
  # Wazuh indexer nodes
  indexer:
    - name: node-1
      ip: "192.168.1.30"
    #- name: node-2
    #  ip: "<indexer-node-ip>"
    #- name: node-3
    #  ip: "<indexer-node-ip>"

  # Wazuh server nodes
  # If there is more than one Wazuh server
  # node, each one must have a node_type
  server:
    - name: wazuh-1
      ip: "192.168.1.22"
    #  node_type: master
    #- name: wazuh-2
    #  ip: "<wazuh-manager-ip>"
    #  node_type: worker
    #- name: wazuh-3
    #  ip: "<wazuh-manager-ip>"
    #  node_type: worker

  # Wazuh dashboard nodes
  dashboard:
    - name: dashboard
      ip: "192.168.1.49"
```

```
root@cyberdojo: /home/cyberdojo
  GNU nano 4.8                    /etc/filebeat/filebeat.yml                    Modifi
# Wazuh - Filebeat configuration file
output.elasticsearch:
  hosts: ["192.168.1.30:9200"]
  protocol: https
  username: ${username}
  password: ${password}
  ssl.certificate_authorities:
    - /etc/filebeat/certs/root-ca.pem
  ssl.certificate: "/etc/filebeat/certs/filebeat.pem"
  ssl.key: "/etc/filebeat/certs/filebeat-key.pem"
setup.template.json.enabled: true
setup.template.json.path: '/etc/filebeat/wazuh-template.json'
setup.template.json.name: 'wazuh'
setup.ilm.overwrite: true
setup.ilm.enabled: false

filebeat.modules:
  - module: wazuh
    alerts:
      enabled: true
    archives:
      enabled: false

logging.level: info
logging.to_files: true
logging.files:
  path: /var/log/filebeat
  name: filebeat
  keepfiles: 7
  permissions: 0644
```

First make sure the tar file we sent from indexer is there.

```
root@cyberdojo:/home/cyberdojo# ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  wazuh-certificates.tar
```

Then,

```
root@cyberdojo:/home/cyberdojo# filebeat keystore create
Created filebeat keystore
root@cyberdojo:/home/cyberdojo#
```

Add the default username and password `admin` : `admin` to the secrets keystore.

```
root@cyberdojo:/home/cyberdojo# echo admin | filebeat keystore add username --stdin --force
Successfully updated the keystore
root@cyberdojo:/home/cyberdojo# echo admin | filebeat keystore add password --stdin --force
Successfully updated the keystore
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# curl -so /etc/filebeat/wazuh-template.json https://raw.githubusercontent.com/wazuh/wazuh/v4.7.3/exte
nsions/elasticsearch/7.x/wazuh-template.json
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# chmod go+r /etc/filebeat/wazuh-template.json
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# curl -s https://packages.wazuh.com/4.x/filebeat/wazuh-filebeat-0.3.tar.gz | tar -xvz -C /usr/share/f
ilebeat/module
wazuh/
wazuh/archives/
wazuh/archives/ingest/
wazuh/archives/ingest/pipeline.json
wazuh/archives/config/
wazuh/archives/config/archives.yml
wazuh/archives/manifest.yml
wazuh/_meta/
wazuh/_meta/config.yml
wazuh/_meta/docs.asciidoc
wazuh/_meta/fields.yml
wazuh/alerts/
wazuh/alerts/ingest/
wazuh/alerts/ingest/pipeline.json
wazuh/alerts/config/
wazuh/alerts/config/alerts.yml
wazuh/alerts/manifest.yml
wazuh/module.yml
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# NODE_NAME=wazuh-1
root@cyberdojo:/home/cyberdojo# mkdir /etc/filebeat/certs
root@cyberdojo:/home/cyberdojo# tar -xf ./wazuh-certificates.tar -C /etc/filebeat/certs/ ./$NODE_NAME.pem ./$NODE_NAME-key.pem ./roo
t-ca.pem
root@cyberdojo:/home/cyberdojo# mv -n /etc/filebeat/certs/$NODE_NAME.pem /etc/filebeat/certs/filebeat.pem
root@cyberdojo:/home/cyberdojo# mv -n /etc/filebeat/certs/$NODE_NAME-key.pem /etc/filebeat/certs/filebeat-key.pem
root@cyberdojo:/home/cyberdojo# chmod 500 /etc/filebeat/certs
root@cyberdojo:/home/cyberdojo# chmod 400 /etc/filebeat/certs/*
root@cyberdojo:/home/cyberdojo# chown -R root:root /etc/filebeat/certs
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# systemctl daemon-reload
root@cyberdojo:/home/cyberdojo# systemctl enable filebeat
Synchronizing state of filebeat.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable filebeat
Created symlink /etc/systemd/system/multi-user.target.wants/filebeat.service → /lib/systemd/system/filebeat.service.
root@cyberdojo:/home/cyberdojo# systemctl start filebeat
root@cyberdojo:/home/cyberdojo# systemctl status filebeat
● filebeat.service - Filebeat sends log files to Logstash or directly to Elasticsearch.
     Loaded: loaded (/lib/systemd/system/filebeat.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-03-22 15:45:54 UTC; 8s ago
       Docs: https://www.elastic.co/products/beats/filebeat
   Main PID: 54843 (filebeat)
      Tasks: 10 (limit: 10416)
     Memory: 11.4M
     CGroup: /system.slice/filebeat.service
             └─54843 /usr/share/filebeat/bin/filebeat --environment systemd -c /etc/filebeat/filebeat.yml --path.home /usr/shar

Mar 22 15:45:54 cyberdojo systemd[1]: Started Filebeat sends log files to Logstash or directly to Elasticsearch..
lines 1-11/11 (END)
```

Verify filebeat is properly installed

```
root@cyberdojo:/home/cyberdojo# filebeat test output
elasticsearch: https://192.168.1.30:9200...
  parse url... OK
  connection...
    parse host... OK
    dns lookup... OK
    addresses: 192.168.1.30
```

For filebeat, enable port 5044 and 9200

```
root@cyberdojo:/home/cyberdojo# ufw allow 5044
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# ufw allow 9200
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# filebeat test output
elasticsearch: https://192.168.1.30:9200...
  parse url... OK
  connection...
    parse host... OK
    dns lookup... OK
    addresses: 192.168.1.30
    dial up... OK
  TLS...
    security: server's certificate chain verification is enabled
    handshake... OK
    TLS version: TLSv1.3
    dial up... OK
  talk to server... OK
  version: 7.10.2
root@cyberdojo:/home/cyberdojo#
```

Open port 55000 so that wazuh dashboard can access wazuh API on wazuh server

```
root@cyberdojo:/home/cyberdojo# ufw allow 55000
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

Wazuh agent communicates on port 1514 and 1515

```
root@cyberdojo:/home/cyberdojo# ufw allow 1514
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# ufw allow 1515
Rule added
Rule added (v6)
```

Miscallaneous tasks

Find running process related to wazuh

```
root@cyberdojo:/home/cyberdojo# ps -ef | grep wazuh
wazuh      61315       1  0 Mar22 ?        00:00:19 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
wazuh      61316   61315  0 Mar22 ?        00:00:01 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
wazuh      61319   61315  0 Mar22 ?        00:00:44 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
wazuh      61322   61315  0 Mar22 ?        00:00:00 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
root       61363       1  0 Mar22 ?        00:02:14 /var/ossec/bin/wazuh-authd
wazuh      61379       1  0 Mar22 ?        00:00:32 /var/ossec/bin/wazuh-db
root       61403       1  0 Mar22 ?        00:00:00 /var/ossec/bin/wazuh-execd
wazuh      61417       1  0 Mar22 ?        00:00:25 /var/ossec/bin/wazuh-analysisd
root       61430       1  0 Mar22 ?        00:00:30 /var/ossec/bin/wazuh-syscheckd
wazuh      61496       1  0 Mar22 ?        00:01:26 /var/ossec/bin/wazuh-remoted
root       61532       1  0 Mar22 ?        00:00:04 /var/ossec/bin/wazuh-logcollector
wazuh      61551       1  0 Mar22 ?        00:00:01 /var/ossec/bin/wazuh-monitord
root       61573       1  0 Mar22 ?        00:00:05 /var/ossec/bin/wazuh-modulesd
root       66050    4641  0 03:20 pts/0    00:00:00 grep --color=auto wazuh
root@cyberdojo:/home/cyberdojo#
```

Find wazuh agent logs

```
root@cyberdojo:/home/cyberdojo# cd /var/ossec/logs/
root@cyberdojo:/var/ossec/logs# ls
active-responses.log  alerts  api  api.log  archives  cluster  cluster.log  firewall  integrations.log  ossec.log  wazuh
root@cyberdojo:/var/ossec/logs# cd archives/
root@cyberdojo:/var/ossec/logs/archives#
root@cyberdojo:/var/ossec/logs/archives# ls
2024  archives.log
root@cyberdojo:/var/ossec/logs/archives# cd 2024
root@cyberdojo:/var/ossec/logs/archives/2024# ls
Mar
root@cyberdojo:/var/ossec/logs/archives/2024# cd Mar/
root@cyberdojo:/var/ossec/logs/archives/2024/Mar#
root@cyberdojo:/var/ossec/logs/archives/2024/Mar# ls
ossec-archive-22.json.sum  ossec-archive-22.log.sum  ossec-archive-23.log
root@cyberdojo:/var/ossec/logs/archives/2024/Mar# cat ossec-archive-23.log
root@cyberdojo:/var/ossec/logs/archives/2024/Mar# cat ossec-archive-22.log.sum
Current checksum:
MD5  (logs/archives/2024/Mar/ossec-archive-22) = none
SHA1 (logs/archives/2024/Mar/ossec-archive-22) = none
SHA256 (logs/archives/2024/Mar/ossec-archive-22) = none

Chained checksum:
MD5  (logs/archives/2024/Mar/ossec-archive-21.log.sum) = none
SHA1 (logs/archives/2024/Mar/ossec-archive-21.log.sum) = none
SHA256 (logs/archives/2024/Mar/ossec-archive-21.log.sum) = none

root@cyberdojo:/var/ossec/logs/archives/2024/Mar#
```

5.2.3 Wazuh Dashboard

The Wazuh Dashboard is a web-based interface that provides visibility into security events and alerts generated by the Wazuh server. In this implementation, we focus on deploying and configuring the Wazuh Dashboard to provide a user-friendly interface for security monitoring and analysis.

Deployment: The Wazuh Dashboard is typically deployed alongside the Wazuh server, using the same server infrastructure. It requires a web server, such as Apache or Nginx, to serve the dashboard interface to users.

Configuration: Once deployed, the Wazuh Dashboard is configured to connect to the Wazuh server and retrieve security event data. Configuration includes specifying the Wazuh server's IP address and port, as well as configuring authentication mechanisms to ensure secure access to the dashboard.

User Access: Access to the Wazuh Dashboard is restricted to authorized users only. Users are required to authenticate using their credentials before accessing the dashboard interface. Role-based access controls can be configured to limit access based on user roles and permissions.

Dashboard Features: The Wazuh Dashboard provides a range of features to help users monitor and analyze security events. These include real-time event monitoring, event correlation, alert management, and customizable dashboards and reports.

Alert Management: The Wazuh Dashboard allows users to view and manage alerts generated by the Wazuh server. Alerts can be filtered based on severity, source, or timestamp, allowing users to focus on critical alerts that require immediate attention.

Visualization: The Wazuh Dashboard provides visualizations, such as charts and graphs, to help users understand security event trends and patterns. Visualizations can be customized to display data relevant to the organization's security posture.

Integration with SIEM: The Wazuh Dashboard can be integrated with a SIEM system, such as the ELK stack, for centralized log management and analysis. Integration allows users to correlate security events from different sources and gain a comprehensive view of the organization's security posture.

Steps:

IP address – 192.168.1.49

```
root@cyberdojo:/home/cyberdojo# apt-get update
Hit:1 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:3 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease
0% [Working]
```

```
Reading package lists... Done
root@cyberdojo:/home/cyberdojo# apt-get install debhelper tar curl libcap2-bin
```

```
root@cyberdojo:/home/cyberdojo# curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --no-default-keyring --keyring gnupg-ring
:/usr/share/keyrings/wazuh.gpg --import && chmod 644 /usr/share/keyrings/wazuh.gpg
gpg: keyring '/usr/share/keyrings/wazuh.gpg' created
gpg: directory '/root/.gnupg' created
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key 96B3EE5F29111145: public key "Wazuh.com (Wazuh Signing Key) <support@wazuh.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1
root@cyberdojo:/home/cyberdojo# echo "deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/apt/ stable main"
 | tee -a /etc/apt/sources.list.d/wazuh.list
deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/apt/ stable main
root@cyberdojo:/home/cyberdojo# apt-get update
Get:1 https://packages.wazuh.com/4.x/apt stable InRelease [17.3 kB]
```

```
root@cyberdojo:/home/cyberdojo# apt-get -y install wazuh-dashboard
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  wazuh-dashboard
0 upgraded, 1 newly installed, 0 to remove and 202 not upgraded.
Need to get 179 MB of archives
```

Configuration

```
root@cyberdojo:/home/cyberdojo# nano /etc/wazuh-dashboard/opensearch_dashboards.yml
```

```
server.host: 0.0.0.0
server.port: 443
opensearch.hosts: https://192.168.1.30:9200
opensearch.ssl.verificationMode: certificate
#opensearch.username:
#opensearch.password:
opensearch.requestHeadersAllowlist: ["securitytenant","Authorization"]
opensearch_security.multitenancy.enabled: false
opensearch_security.readonly_mode.roles: ["kibana_read_only"]
server.ssl.enabled: true
server.ssl.key: "/etc/wazuh-dashboard/certs/dashboard-key.pem"
server.ssl.certificate: "/etc/wazuh-dashboard/certs/dashboard.pem"
opensearch.ssl.certificateAuthorities: ["/etc/wazuh-dashboard/certs/root-ca.pem"]
uiSettings.overrides.defaultRoute: /app/wazuh
```

First make sure the tar file we sent from indexer is there.

```
root@cyberdojo:/home/cyberdojo# ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  wazuh-certificates.tar
```

Then,

```
root@cyberdojo:/home/cyberdojo# NODE_NAME=dashboard
root@cyberdojo:/home/cyberdojo# mkdir /etc/wazuh-dashboard/certs
root@cyberdojo:/home/cyberdojo# tar -xf ./wazuh-certificates.tar -C /etc/wazuh-dashboard/certs/ ./$NODE_NAME.pem ./$NODE_NAME-key.pe
m ./root-ca.pem
root@cyberdojo:/home/cyberdojo# mv -n /etc/wazuh-dashboard/certs/$NODE_NAME.pem /etc/wazuh-dashboard/certs/dashboard.pem
root@cyberdojo:/home/cyberdojo# mv -n /etc/wazuh-dashboard/certs/$NODE_NAME-key.pem /etc/wazuh-dashboard/certs/dashboard-key.pem
root@cyberdojo:/home/cyberdojo# chmod 500 /etc/wazuh-dashboard/certs
root@cyberdojo:/home/cyberdojo# chmod 400 /etc/wazuh-dashboard/certs/*
root@cyberdojo:/home/cyberdojo# chown -R wazuh-dashboard:wazuh-dashboard /etc/wazuh-dashboard/certs
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# systemctl daemon-reload
root@cyberdojo:/home/cyberdojo# systemctl enable wazuh-dashboard
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-dashboard.service → /etc/systemd/system/wazuh-dashboard.service.
root@cyberdojo:/home/cyberdojo# systemctl start wazuh-dashboard
root@cyberdojo:/home/cyberdojo# systemctl status wazuh-dashboard
● wazuh-dashboard.service - wazuh-dashboard
     Loaded: loaded (/etc/systemd/system/wazuh-dashboard.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-03-22 16:11:24 UTC; 1min 14s ago
   Main PID: 13797 (node)
      Tasks: 11 (limit: 10416)
     Memory: 146.3M
     CGroup: /system.slice/wazuh-dashboard.service
             └─13797 /usr/share/wazuh-dashboard/node/bin/node --no-warnings --max-http-header-size=65536 --unhandled-rejections=war>
```

The next step is only for distributed deployments

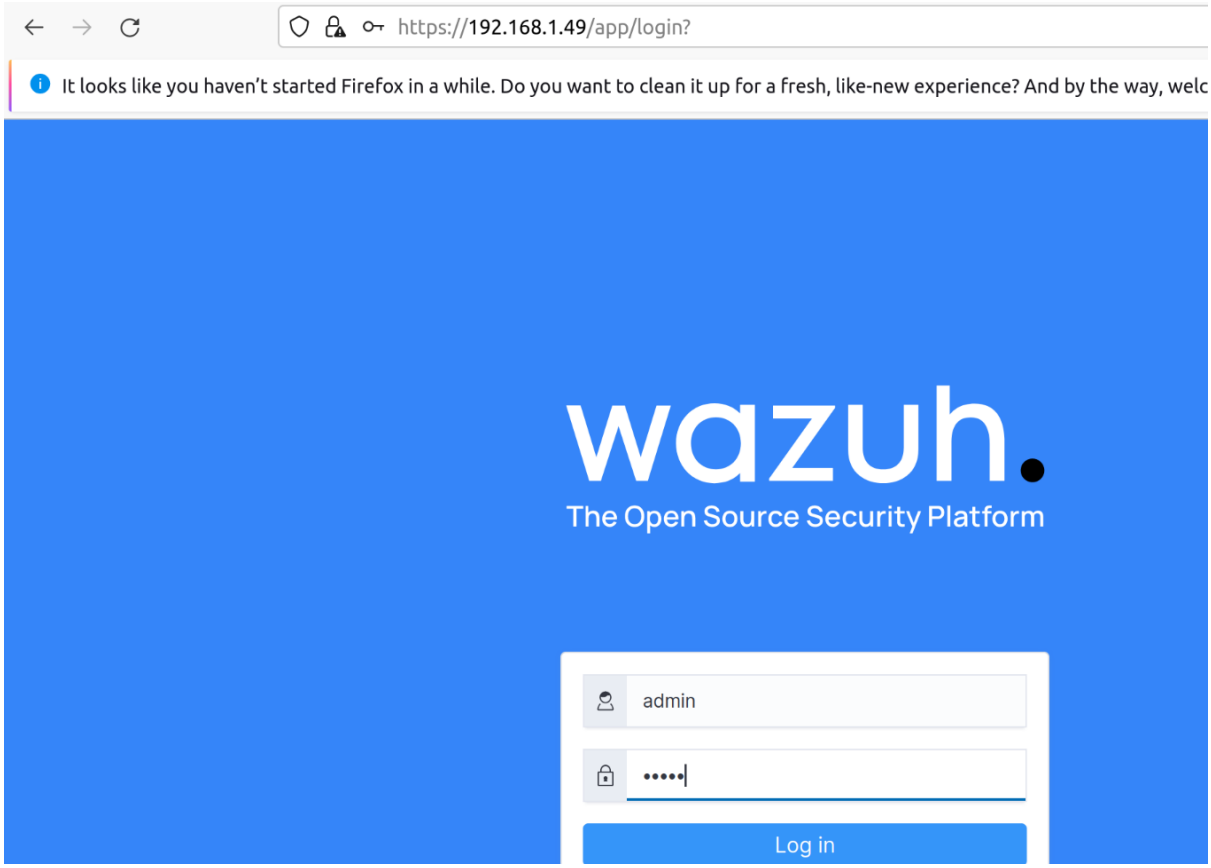
```
root@cyberdojo:/home/cyberdojo# nano /usr/share/wazuh-dashboard/data/wazuh/config/wazuh.yml
```

Change url only in the 2nd stanza, made a mistake by changing it in the 1st stanza also.

```
#
# hosts:
#   # Host ID / name,
#   - env-1:
#       # Host URL
#       url: https://192.168.1.22
#       # Host / API port
#       port: 55000
#       # Host / API username
#       username: wazuh-wui
#       # Host / API password
#       password: wazuh-wui
#       # Use RBAC or not. If set to true, the username must be "wazuh-wui".
#       run_as: true
#   - env-2:
#       url: https://env-2.example
#       port: 55000
#       username: wazuh-wui
#       password: wazuh-wui
#       run_as: true

hosts:
  - default:
      url: https://192.168.1.22
      port: 55000
      username: wazuh-wui
      password: wazuh-wui
      run_as: false
```

Access the Wazuh web interface with your credentials.

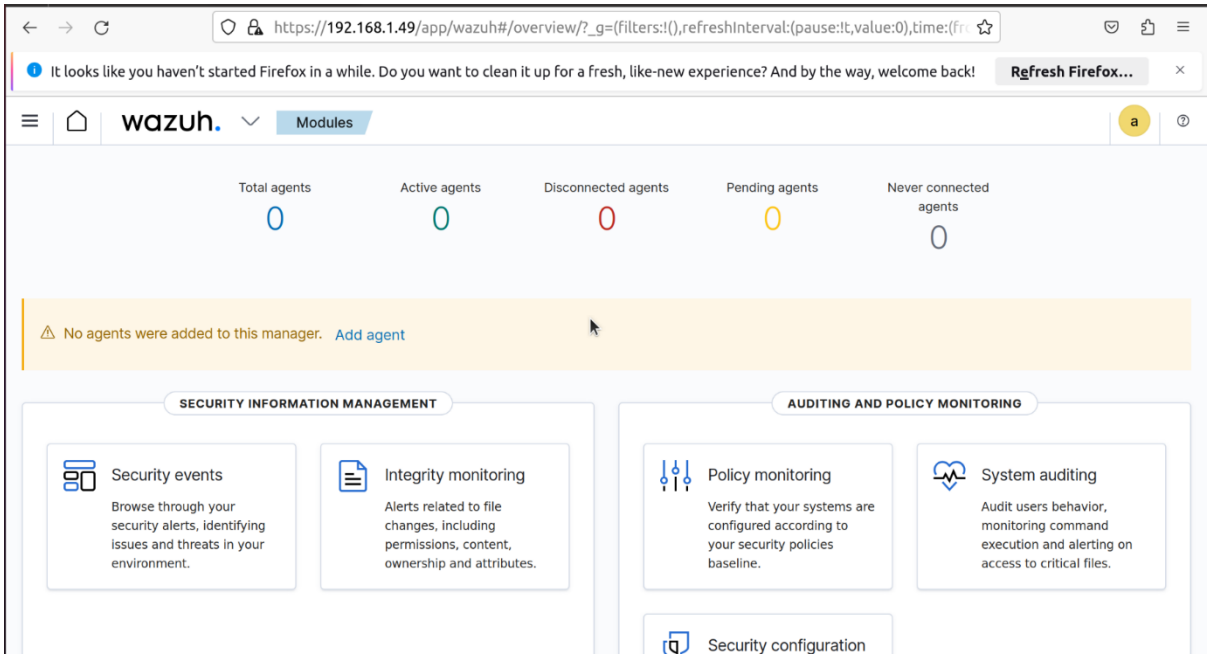
- URL: *https://<wazuh-dashboard-ip>*
- **Username**: *admin*
- **Password**: *admin*



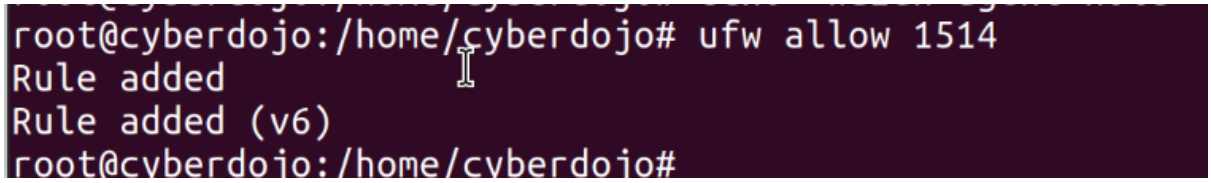
Open port 55000 so that wazuh dashboard can access wazuh API on wazuh server

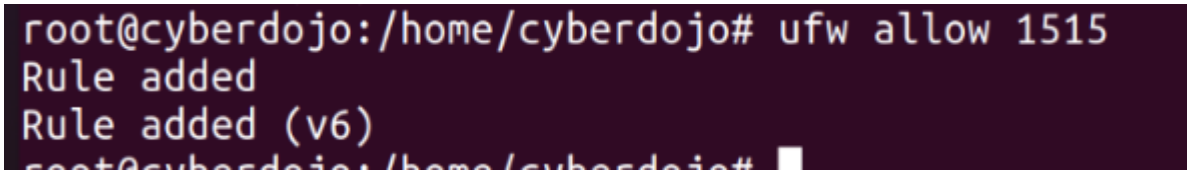
Open this port on wazuh server as well as wazuh dashboard.

```
root@cyberdojo:/home/cyberdojo# ufw allow 55000
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```



Wazuh agent communicates on port 1514 and 1515





5.2.4 Wazuh Agent

The Wazuh Agent is a lightweight endpoint security agent that collects and sends security event data to the Wazuh server for analysis. In this implementation, we focus on deploying and configuring the Wazuh Agent on endpoint devices to enhance security monitoring and threat detection.

Deployment: The Wazuh Agent is deployed on endpoint devices, such as workstations, servers, and cloud instances, to collect security event data. The agent is installed using a package manager or installer specific to the operating system of the endpoint device.

Configuration: Once deployed, the Wazuh Agent is configured to communicate with the Wazuh server and send security event data. Configuration includes specifying the Wazuh server's IP address and port, as well as configuring encryption and authentication settings for secure communication.

Active Response: The Wazuh Agent can be configured to perform active response actions, such as blocking IP addresses or killing processes, in response to detected security threats. Active response actions are configured based on predefined rules and can help mitigate the impact of security incidents.

File Integrity Monitoring: The Wazuh Agent includes file integrity monitoring (FIM) capabilities to detect unauthorized changes to files and directories. FIM rules are configured to monitor specific files and directories for changes and trigger alerts when unauthorized modifications are detected.

Rootkit Detection: The Wazuh Agent includes rootkit detection capabilities to identify and remove rootkits from endpoint devices. Rootkit detection rules are configured to scan for known rootkit signatures and behavior patterns and trigger alerts when rootkits are detected.

Log Collection: In addition to security event data, the Wazuh Agent can also collect and send log data from endpoint devices to the Wazuh server. Log collection rules are configured to specify which log files to monitor and send to the server for analysis.

Steps:

(WE WILL BE USING 2-3 VMS TO MONITOR USING WAZUH)

Wazuh server – 192.168.1.22

Important points

- Logs from wazuh agents are stored in /var/ossec/logs/archives on wazuh server.
  - Question – why not wazuh indexer, why wazuh server?
  - If logs are stored on wazuh server, then what's indexer's job?
  - Answer is wazuh agent logs are stored in wazuh server. Wazuh indexer serves as buffer between wazuh manager and elasticsearch.
  - Wazuh indexer parses logs before sending to elasticsearch, similar to logstash or HF in splunk.
-

```
cyberdojo@cyberdojo:~$ sudo su
[sudo] password for cyberdojo:
root@cyberdojo:/home/cyberdojo# curl -s https://packages.wazuh.com/key/GPG-KEY-W
AZUH | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/wazuh.g
pg --import && chmod 644 /usr/share/keyrings/wazuh.gpg
gpg: keyring '/usr/share/keyrings/wazuh.gpg' created
gpg: directory '/root/.gnupg' created
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key 96B3EE5F29111145: public key "Wazuh.com (Wazuh Signing Key) <support@wa
zuh.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# echo "deb [signed-by=/usr/share/keyrings/wazuh.g
pg] https://packages.wazuh.com/4.x/apt/ stable main" | tee -a /etc/apt/sources.l
ist.d/wazuh.list
deb [signed-by=/usr/share/keyrings/wazuh.gpg] https://packages.wazuh.com/4.x/apt
/ stable main
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# apt-get update
Hit:1 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:2 https://packages.wazuh.com/4.x/apt stable InRelease [17.3 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:4 https://packages.wazuh.com/4.x/apt stable/main amd64 Packages [39.5 kB]
0% [Waiting for headers]
```

```
root@cyberdojo:/home/cyberdojo# WAZUH_MANAGER="192.168.1.22" apt-get install waz
uh-agent
```

```
root@cyberdojo:/home/cyberdojo# systemctl daemon-reload
root@cyberdojo:/home/cyberdojo# systemctl enable wazuh-agent
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-agent.service
→ /lib/systemd/system/wazuh-agent.service.
root@cyberdojo:/home/cyberdojo# systemctl start wazuh-agent
root@cyberdojo:/home/cyberdojo# systemctl status wazuh-agent
● wazuh-agent.service - Wazuh agent
     Loaded: loaded (/lib/systemd/system/wazuh-agent.service; enabled; vendor p>
     Active: active (running) since Fri 2024-03-22 17:15:17 UTC; 16s ago
    Process: 6436 ExecStart=/usr/bin/env /var/ossec/bin/wazuh-control start (co>
      Tasks: 33 (limit: 10416)
     Memory: 17.8M
     CGroup: /system.slice/wazuh-agent.service
             ├─6458 /var/ossec/bin/wazuh-execd
             ├─6490 /var/ossec/bin/wazuh-agentd
             ├─6503 /var/ossec/bin/wazuh-syscheckd
             ├─6513 /var/ossec/bin/wazuh-logcollector
             └─6530 /var/ossec/bin/wazuh-modulesd
```

In the next 3 screenshots we will be disabling wazuh agent updates since wazuh agent version should always be equal to or lower than wazuh manager version

```
root@cyberdojo:/home/cyberdojo# sed -i "s/^deb/#deb/" /etc/apt/sources.list.d/wa
zuh.list
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# apt-get update
Hit:1 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:3 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu focal-security InRelease
0% [Working]
```

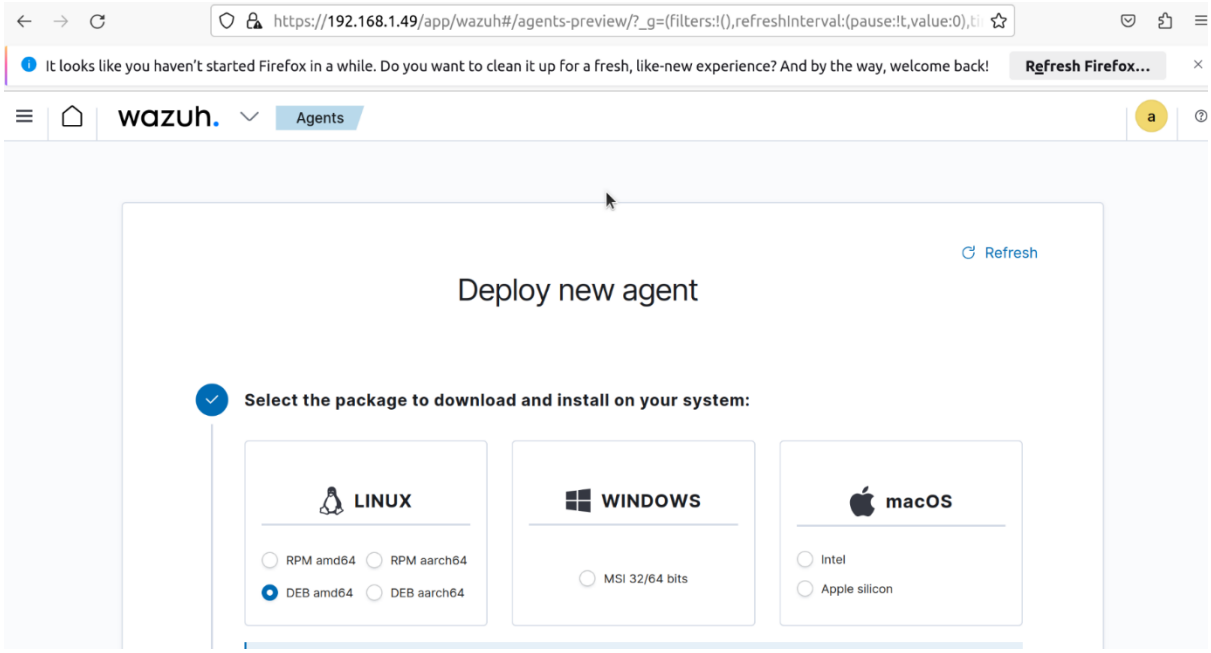
```
root@cyberdojo:/home/cyberdojo# echo "wazuh-agent hold" | dpkg --set-selections
root@cyberdojo:/home/cyberdojo#
```

Wazuh agent communicates on port 1514 and 1515

```
root@cyberdojo:/home/cyberdojo# ufw allow 1514
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# ufw allow 1515
Rule added
Rule added (v6)
```

ALTERNATIVELY, WE CAN ADD AGENT FROM WAZUH DASHBOARD



Server address:

This is the address the agent uses to communicate with the server. Enter an IP address or a fully qualified domain name (FDQN).

Assign a server address:

192.168.1.52

Optional settings:

By default, the deployment uses the hostname as the agent name. Optionally, you can use a different agent name in the field below.

Assign an agent name:

agent-1

Run the following commands to download and install the agent:

```
wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.3-1_amd64.deb &&
sudo WAZUH_MANAGER='192.168.1.52' WAZUH_AGENT_NAME='agent-1' dpkg -i ./wazuh-
agent_4.7.3-1_amd64.deb
```

Requirements

- You will need administrator privileges to perform this installation.

5 Start the agent:

```
sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
```

Then on the agent vm just copy paste

```
cyberdojo@cyberdojo:~$ wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.3-1_amd64.deb && sudo WAZUH_M
ANAGER='192.168.1.22' WAZUH_AGENT_NAME='agent-1' dpkg -i ./wazuh-agent_4.7.3-1_amd64.deb
--2024-03-22 17:39:59--  https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.3-1_amd64.deb
Resolving packages.wazuh.com (packages.wazuh.com)... 2600:9000:2178:d600:8:fed3:b0c0:93a1, 2600:9000:2178:4c00:8:fed3:b0c0:93a1, 260
0:9000:2178:8c00:8:fed3:b0c0:93a1, ...
Connecting to packages.wazuh.com (packages.wazuh.com)|2600:9000:2178:d600:8:fed3:b0c0:93a1|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9362524 (8.9M) [binary/octet-stream]
Saving to: 'wazuh-agent_4.7.3-1_amd64.deb'
```

```
cyberdojo@cyberdojo:~$ systemctl daemon-reload
```

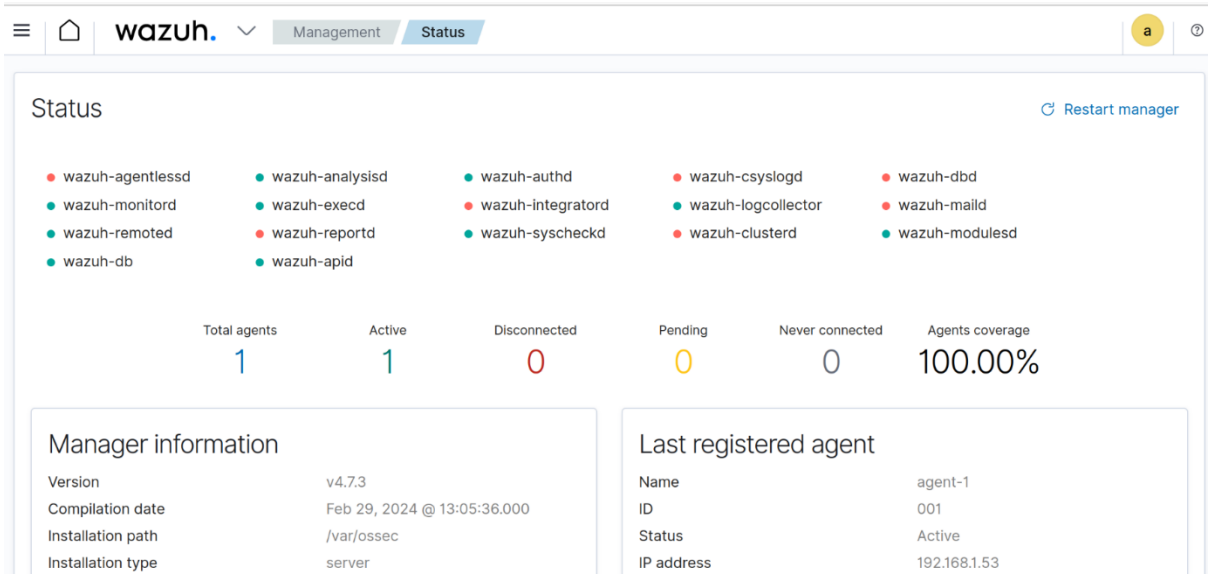
```
root@cyberdojo:/home/cyberdojo# systemctl enable wazuh-agent
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-agent.service → /lib/systemd/system/wazuh-agent.service.
root@cyberdojo:/home/cyberdojo# systemctl start wazuh-agent
root@cyberdojo:/home/cyberdojo# systemctl status wazuh-agent
● wazuh-agent.service - Wazuh agent
     Loaded: loaded (/lib/systemd/system/wazuh-agent.service; enabled; vendor p>
     Active: active (running) since Fri 2024-03-22 17:41:53 UTC; 7s ago
    Process: 5859 ExecStart=/usr/bin/env /var/ossec/bin/wazuh-control start (co>
      Tasks: 31 (limit: 10416)
     Memory: 126.4M
     CGroup: /system.slice/wazuh-agent.service
             ├─5881 /var/ossec/bin/wazuh-execd
             ├─5907 /var/ossec/bin/wazuh-agentd
             ├─5920 /var/ossec/bin/wazuh-syscheckd
             ├─5930 /var/ossec/bin/wazuh-logcollector
             └─5947 /var/ossec/bin/wazuh-modulesd
```

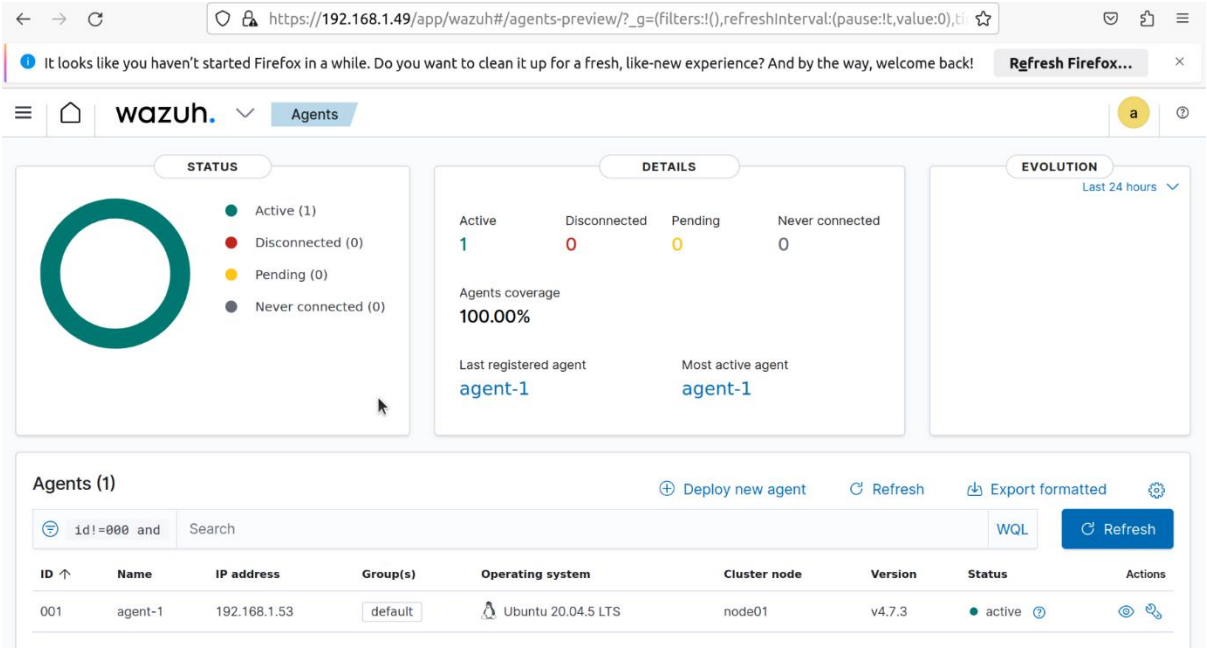
Wazuh agent communicates on port 1514 and 1515

```
root@cyberdojo:/home/cyberdojo# ufw allow 1514
Rule added
Rule added (v6)
root@cyberdojo:/home/cyberdojo#
```

```
root@cyberdojo:/home/cyberdojo# ufw allow 1515
Rule added
Rule added (v6)
```

Now, to verify, check on wazuh dashboard





If it still does not work, go to wazuh server vm and restart wazuh-manager. Systemctl restart wazuh-manager.

CHAPTER – 6

RESULT & DISCUSSIONS

6.1 Overview

The implementation of our SIEM solution using the ELK stack and Wazuh has enabled us to address several critical use cases that are essential for enhancing our organization's cybersecurity posture. By leveraging the capabilities of the ELK stack and Wazuh, we have been able to effectively monitor, detect, and respond to various security threats, ensuring the protection of our organization's assets and data.

Some of the key use cases that we have implemented is the Brute Force Detection and File Integrity Monitoring.

File Integrity Monitoring (FIM)

FIM is a key component of our SIEM solution, allowing us to monitor changes to files and directories across our IT infrastructure. By continuously monitoring file integrity, we can quickly detect unauthorized modifications that may indicate a security breach. FIM enables us to maintain the integrity of our systems and data, ensuring the confidentiality and availability of our organization's information assets.

Brute Force Detection

Brute force attacks are a common method used by threat actors to gain unauthorized access to systems and applications. To combat this threat, we have implemented brute force detection as part of our SIEM solution. By analyzing login attempts and identifying patterns indicative of brute force attacks, we can automatically block malicious IP addresses and prevent unauthorized access attempts. This proactive approach helps us protect our systems and data from unauthorized access, ensuring the security of our IT infrastructure.

In conclusion, the implementation of File Integrity Monitoring and Brute Force Detection as part of our SIEM solution has significantly strengthened our organization's cybersecurity defenses. These use cases enable us to detect and respond to security incidents more effectively, ensuring the integrity and security of our systems and data.

6.2 Use Case Implementation / Output

6.2.1 File Integrity Monitoring (FIM)

Wazuh agent : making directory in wazuh agent

```
cyberdojo@cyberdojo:~$ pwd
/home/cyberdojo
cyberdojo@cyberdojo:~$ mkdir siem
```

Wazuh agent: adding rule to ossec.conf file

<directories check_all="yes" report_changes="yes" realtime="yes">/home/cyberdojo/siem</directories>

```
root@cyberdojo:/home/cyberdojo# vim /var/ossec/etc/ossec.conf
root@cyberdojo:/home/cyberdojo#
```

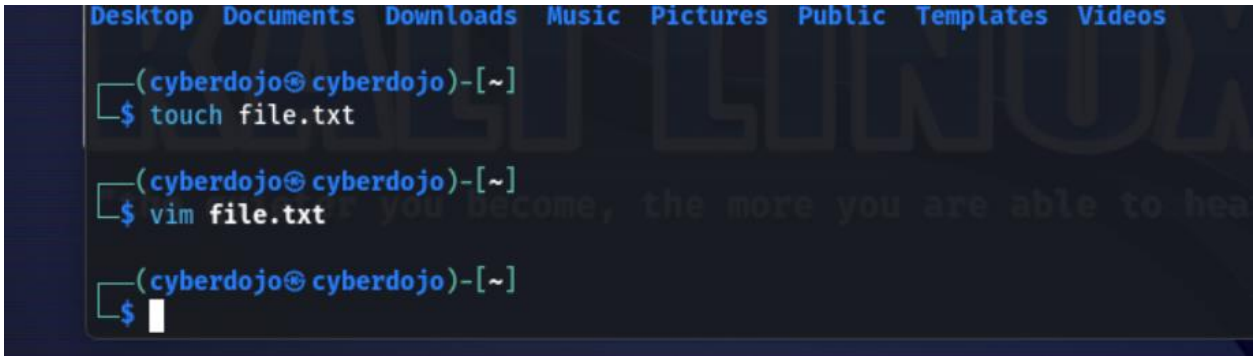
```
  <!-- File integrity monitoring -->
  <syscheck>
    <disabled>no</disabled>

    <!-- Frequency that syscheck is executed default every 12 hours -->
    <frequency>43200</frequency>

    <scan_on_start>yes</scan_on_start>

    <!-- Directories to check  (perform all possible verifications) -->
    <directories>/etc,/usr/bin,/usr/sbin</directories>
    <directories>/bin,/sbin,/boot</directories>
    <directories check_all="yes" report_changes="yes" realtime="yes">/home/cyberdojo/siem</directories>

    <!-- Files/directories to ignore -->
    <ignore>/etc/mtab</ignore>
    <ignore>/etc/hosts.deny</ignore>
    <ignore>/etc/mail/statistics</ignore>
    <ignore>/etc/random-seed</ignore>
-- INSERT --                                                    109,5         45%
```

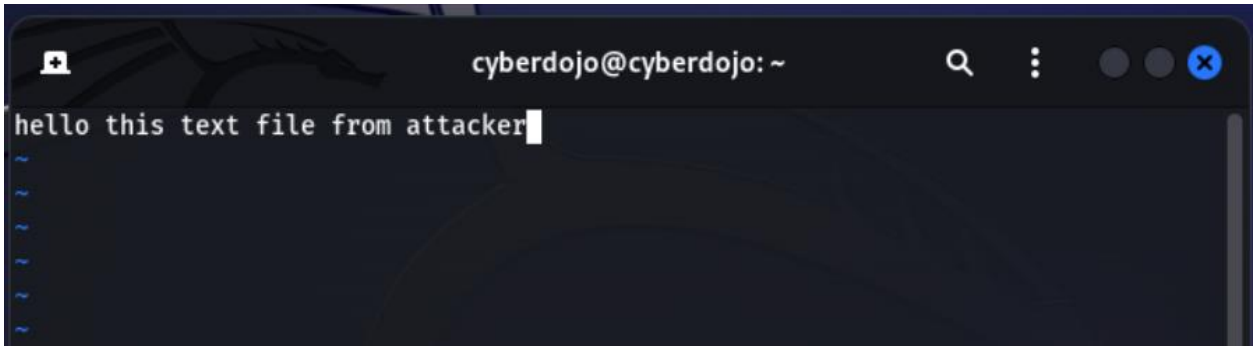
Wazuh agent : restarting the agent

```
root@cyberdojo:/home/cyberdojo# systemctl restart wazuh-agent
root@cyberdojo:/home/cyberdojo#
```

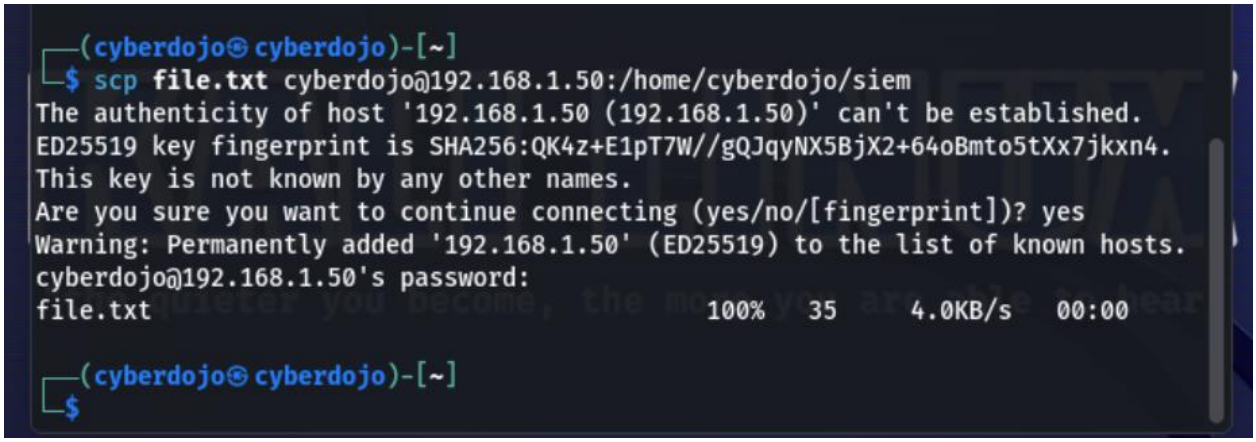
On Attacker side

Attacker : creating file.txt and edit using vim





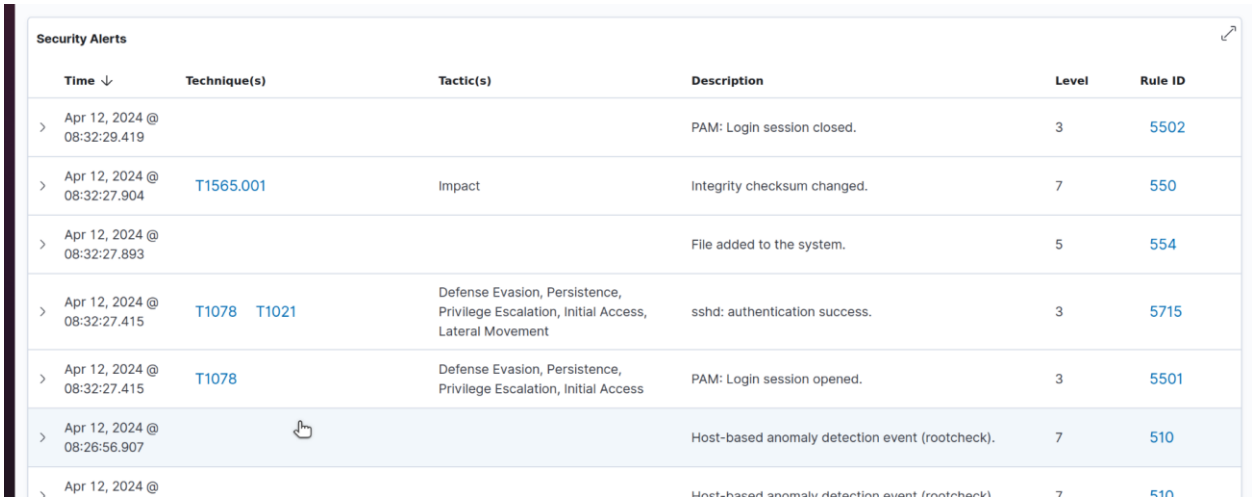
Attacker : sending to wazuh agent into /home/cyberdojo/siem folder using scp command



On Wazuh dashboard

Wazuh Dashboard: we can see here alert has been generated 5 alerts are here

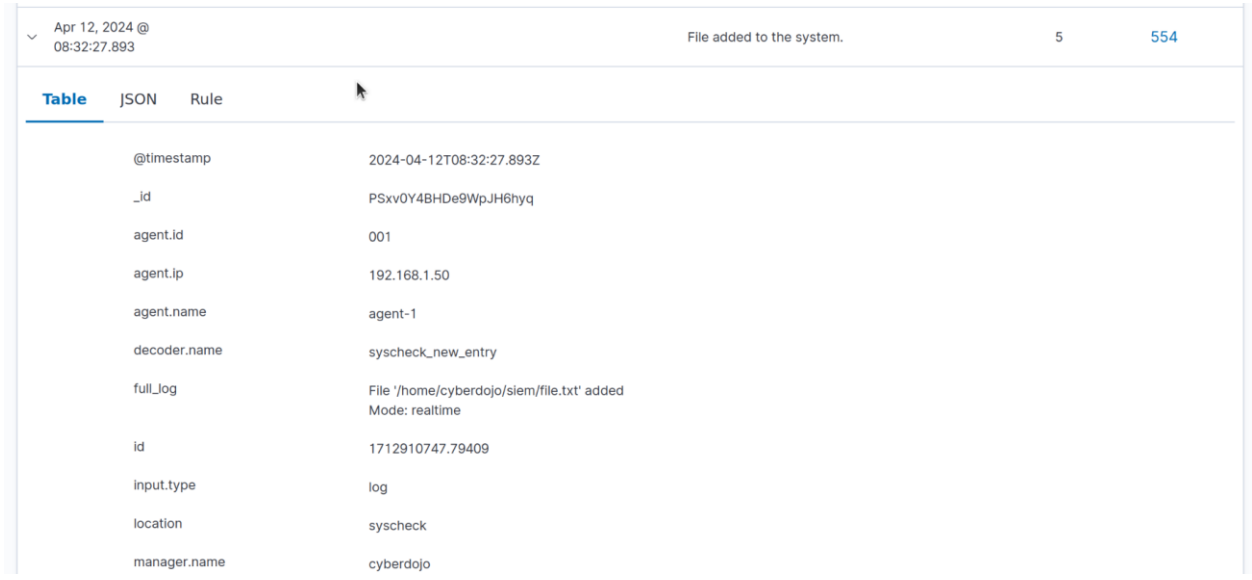
1. Pam:login session opened
2. Sshd:authentication
3. File added to system
4. Integrity checksum changed
5. Pam:login session closed



Security Alerts

Time	Technique(s)	Tactic(s)	Description	Level	Rule ID
Apr 12, 2024 @ 08:32:29.419			PAM: Login session closed.	3	5502
Apr 12, 2024 @ 08:32:27.904	T1565.001	Impact	Integrity checksum changed.	7	550
Apr 12, 2024 @ 08:32:27.893			File added to the system.	5	554
Apr 12, 2024 @ 08:32:27.415	T1078 T1021	Defense Evasion, Persistence, Privilege Escalation, Initial Access, Lateral Movement	sshd: authentication success.	3	5715
Apr 12, 2024 @ 08:32:27.415	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	PAM: Login session opened.	3	5501
Apr 12, 2024 @ 08:26:56.907			Host-based anomaly detection event (rootcheck).	7	510
Apr 12, 2024 @					

File added to the system alert in details



Apr 12, 2024 @ 08:32:27.893 — File added to the system. — 5 — 554

Table JSON Rule

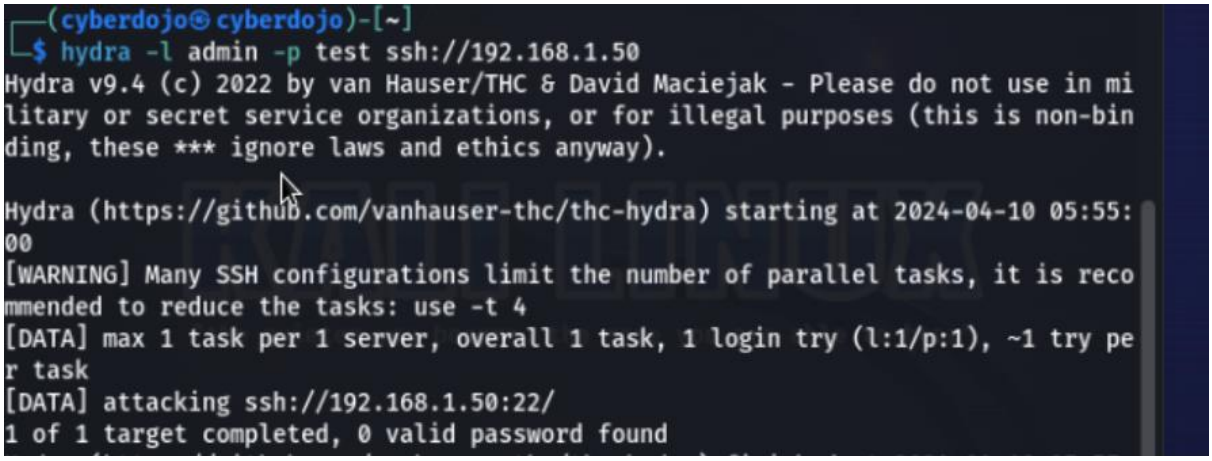
@timestamp	2024-04-12T08:32:27.893Z
_id	PSxv0Y4BHDe9WpJH6hyq
agent.id	001
agent.ip	192.168.1.50
agent.name	agent-1
decoder.name	syscheck_new_entry
full_log	File '/home/cyberdojo/siem/file.txt' added Mode: realtime
id	1712910747.79409
input.type	log
location	syscheck
manager.name	cyberdojo

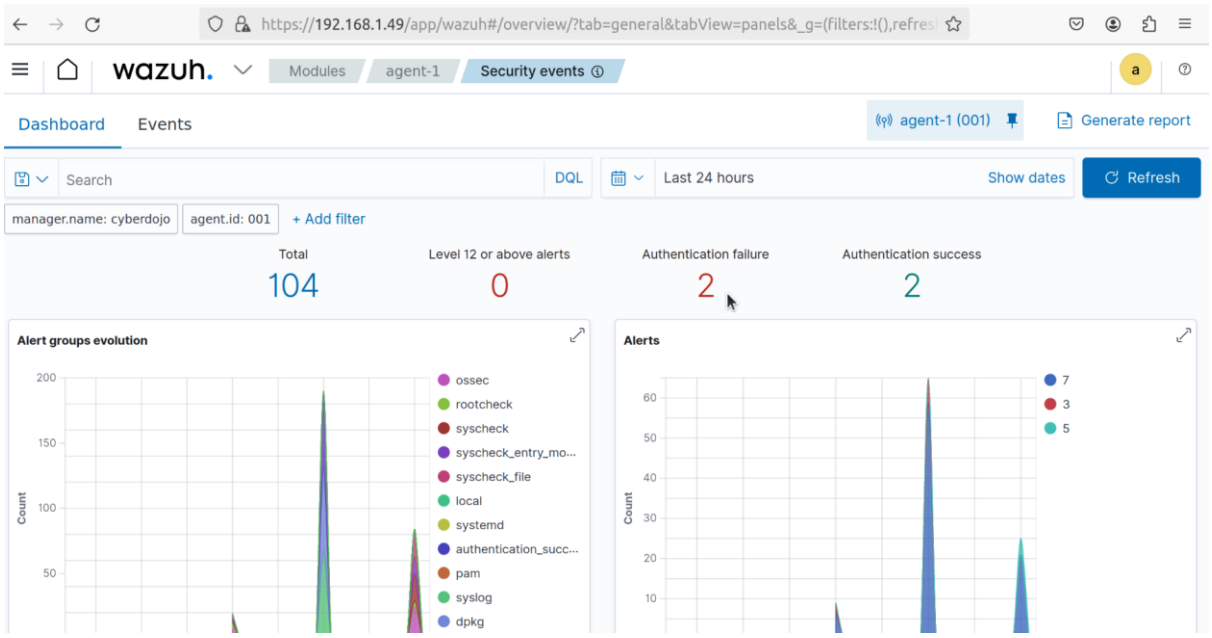
input.type	log
location	syscheck
manager.name	cyberdojo
rule.description	File added to the system.
rule.firedtimes	1
rule.gdpr	II_5.1.f
rule.gpg13	4.11
rule.groups	ossec, syscheck, syscheck_entry_added, syscheck_file
rule.hipaa	164.312.c.1, 164.312.c.2
rule.id	554
rule.level	5
rule.mail	false
rule.nist_800_53	SI.7
rule.pci_dss	11.5
rule.tsc	PI1.4, PI1.5, CC6.1, CC6.8, CC7.2, CC7.3

syscheck.event	added
syscheck.gid_after	1000
syscheck.gname_after	cyberdojo
syscheck.inode_after	657012
syscheck.md5_after	d41d8cd98f00b204e9800998ecf8427e
syscheck.mode	realtime
syscheck.mtime_after	2024-04-12T08:32:27
syscheck.path	/home/cyberdojo/siem/file.txt
syscheck.perm_after	rw-r--r--
syscheck.sha1_after	da39a3ee5e6b4b0d3255bfef95601890afd80709
syscheck.sha256_after	e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
syscheck.size_after	0
syscheck.uid_after	1000
syscheck.uname_after	cyberdojo
timestamp	2024-04-12T08:32:27.893+0000

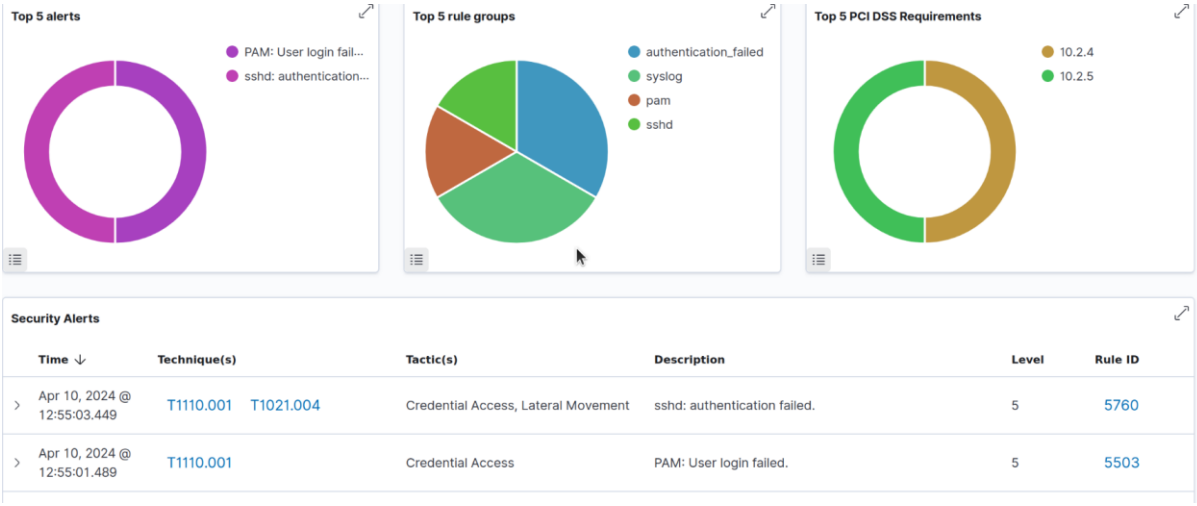
6.2.2 Brute Force Detection

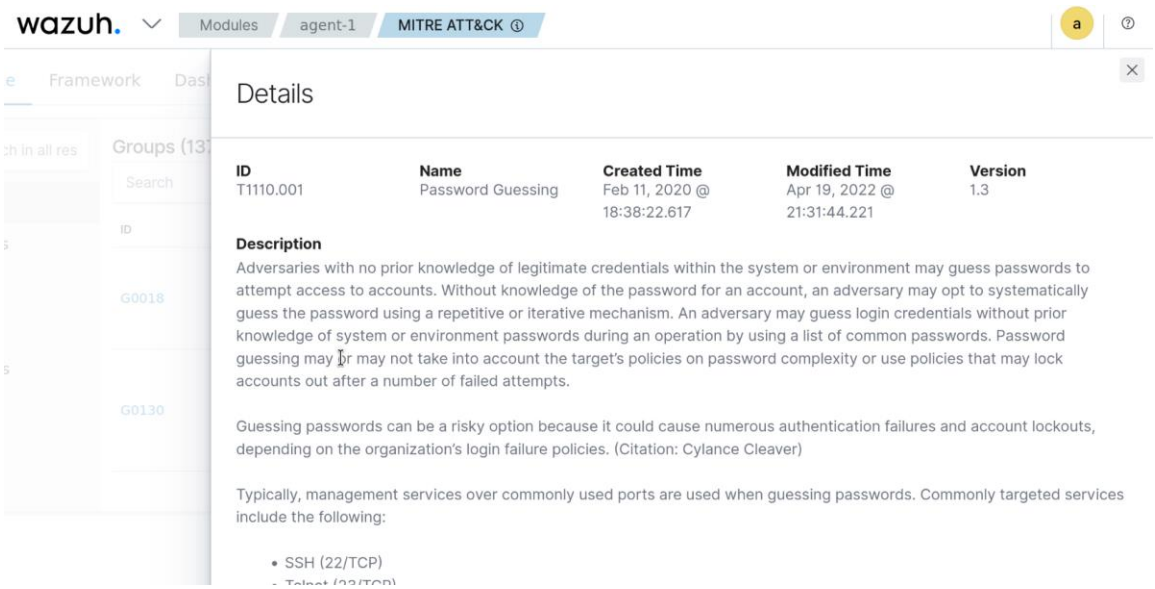
STEPS:

- Launch attack on wazuh agent vm from kali linux vm.
  - 

- Now, go to wazuh dashboard and checks security events related to particular agent.
  - 

- Now, click on authentication failure to see details regarding the failed ssh attempts.





In the future, you can also create custom rules.

Add the following rule to `/var/ossec/etc/rules/local_rules.xml` .

```
<group name="custom_rules_example,">
  <rule id="100010" level="0">
    <program_name>example</program_name>
    <description>User logged</description>
  </rule>
</group>
```

https://documentation.wazuh.com/current/user-manual/ruleset/custom.html

6.3 SUMMARY

Our project journey to implement a Security Information and Event Management (SIEM) solution using the ELK stack and Wazuh has been transformative, marking a significant shift in our cybersecurity approach. This initiative has not only revamped our log management, threat detection, and incident response strategies but has also fundamentally strengthened our overall security posture.

The adoption of the ELK stack has revolutionized our log management practices by centralizing the management and analysis of log data from various sources. This centralized approach has streamlined our log management processes, providing us with a unified platform to monitor and analyse security events. Additionally, the ELK stack has offered valuable insights into our security landscape, enabling us to proactively identify and address potential security risks.

Furthermore, the integration of Wazuh agents on our endpoints has empowered us with real-time threat detection capabilities. These agents continuously monitor our endpoint devices for suspicious activity, allowing us to swiftly identify and mitigate potential security incidents. This proactive approach has been instrumental in enhancing our organization's ability to respond to security threats promptly and effectively.

The user-friendly dashboard provided by Wazuh has further improved our security operations by simplifying the monitoring and analysis of security events. This intuitive interface has enabled our security team to monitor and analyse security events with ease, enhancing our overall security posture.

Looking ahead, we are committed to exploring advanced threat detection techniques, integrating our SIEM solution with cloud services, and further refining our incident response strategies. We believe that these efforts will further enhance our organization's cybersecurity defenses, ensuring the continued protection of our assets and data against evolving cyber threats.

CHAPTER – 7

CONCLUSIONS & FUTURE SCOPE

The implementation of our Security Information and Event Management (SIEM) solution using the ELK stack and Wazuh has been a significant milestone in fortifying our organization's cybersecurity posture. This project has not only enhanced our ability to monitor and detect security threats but has also improved our overall security readiness.

7.1 Key Achievements

Centralized Log Management: The ELK stack has provided us with a centralized platform for managing and analysing log data from various sources. This centralized approach has streamlined our log management process, enabling us to easily monitor and analyze security events.

Real-time Threat Detection: The integration of Wazuh agents on our endpoint devices has significantly improved our ability to detect and respond to security threats in real-time. The agents' capabilities for active response, file integrity monitoring, and rootkit detection have added layers of security to our critical assets, ensuring that we are promptly alerted to any potential security incidents.

Intuitive Dashboard: The Wazuh Dashboard has proven to be an invaluable tool for monitoring and analyzing security events. Its user-friendly interface and customizable dashboards have empowered our security team to quickly identify and respond to potential threats, enabling us to proactively address security issues before they escalate.

Enhanced Incident Response: The implementation of our SIEM solution has greatly improved our incident response capabilities. With real-time alerting and monitoring, we can quickly identify and respond to security incidents, reducing the risk of data breaches and other security breaches. The ability to centrally manage and analyze security events has also enabled us to conduct thorough investigations into security incidents, allowing us to identify and remediate vulnerabilities in our IT environment.

7.2 Conclusion

In conclusion, the implementation of our SIEM solution using the ELK stack and Wazuh has been a game-changer for our organization's cybersecurity posture. This project has not only significantly enhanced our ability to monitor and detect security threats but has also revolutionized our incident response capabilities. The centralized log management provided by the ELK stack has allowed us to efficiently manage and analyze log data from various sources, providing us with a comprehensive view of our security landscape. Moreover, the real-time threat detection capabilities of Wazuh have empowered us to swiftly identify and mitigate security incidents, minimizing their impact on our organization. Additionally, the intuitive dashboard has streamlined our security operations, offering a user-friendly interface for monitoring and analyzing security events, thereby enabling us to proactively address potential threats. Overall, this project has been a resounding success, and we are confident that our SIEM solution will continue to be a cornerstone of our cybersecurity strategy, ensuring the protection of our organization's assets in the face of evolving cyber threats.

In summary, the successful deployment of our SIEM solution using the ELK stack and Wazuh has significantly bolstered our organization's cybersecurity defenses. By centralizing log management and leveraging real-time threat detection capabilities, we have greatly enhanced our ability to detect, respond to, and mitigate security threats. The ELK stack's centralized log management has enabled us to effectively manage and analyze log data from diverse sources, providing us with valuable insights into potential security incidents. Furthermore, the real-time threat detection capabilities of Wazuh have allowed us to quickly identify and address security threats, minimizing their impact on our organization. Additionally, the intuitive dashboard has simplified our security operations, offering a user-friendly interface for monitoring and analyzing security events, enabling us to proactively address potential threats. This project has been instrumental in strengthening our overall security posture, and we are confident that our SIEM solution will continue to play a crucial role in protecting our organization's assets against cyber threats.

7.2 Future Scope

The successful implementation of our SIEM solution using the ELK stack and Wazuh has paved the way for several future enhancements and expansions that will further strengthen our organization's cybersecurity posture. As we continue to evolve our security strategy, the following areas present exciting opportunities for growth and improvement:

Advanced Threat Detection: We plan to leverage advanced threat detection techniques, such as machine learning and behavioral analytics, to enhance our ability to detect and respond to sophisticated cyber threats. By analyzing patterns and anomalies in our security data, we aim to identify threats before they can cause significant damage.

Integration with Cloud Services: With the increasing adoption of cloud services, we recognize the importance of integrating our SIEM solution with cloud platforms to monitor and secure our cloud-based assets effectively. This integration will allow us to extend our security monitoring capabilities to our cloud environments, ensuring comprehensive protection across all our IT infrastructure.

Enhanced Incident Response: Our goal is to further enhance our incident response capabilities by automating response actions and developing more efficient incident response workflows. By automating repetitive tasks and streamlining our response processes, we can minimize the impact of security incidents and reduce the time taken to resolve them.

Improved Compliance: Compliance with regulatory requirements and industry standards is critical for maintaining the trust of our customers and partners. We will continue to improve our compliance monitoring and reporting capabilities to ensure that we meet these requirements effectively. This includes regular audits, documentation of security policies and procedures, and ongoing training for our staff.

Threat Intelligence Integration: Integration with threat intelligence feeds will enable us to stay updated on the latest threats and trends in the cybersecurity landscape. By leveraging threat intelligence, we can proactively defend against cyber-attacks and better understand the tactics, techniques, and procedures (TTPs) used by threat actors.

Scalability and Performance: As our organization grows, we need to ensure that our SIEM solution can scale to handle the increasing volume of security data generated by our IT infrastructure. We will focus on enhancing the scalability and performance of our SIEM solution to meet these growing demands without compromising on security.

CHAPTER – 8

Reference

https://www.manageengine.com/log-management/siem/siem-components.html

https://www.manageengine.com/log-management/siem/siem-functions.html

https://www.exabeam.com/explainers/siem/what-is-siem/

https://www.paloaltonetworks.co.uk/cyberpedia/what-is-siem-logging

https://www.strongdm.com/what-is/siem-vs-log-management

https://www.aricoma.com/solutions/enterprise-cybersecurity/security-information-%C2%A0event-management/complex-security-management-systems-siem

https://www.rippling.com/blog/engineering-siem-part-1

https://infosecwriteups.com/building-a-siem-combining-elk-wazuh-hids-and-elastalert-for-optimal-performance-f1706c2b73c6

https://www.youtube.com/watch?v=2tnSfuYF60A&list=PLgfKCD7KgSTKFEXDIB9i203FY0o18rgB3&index=9&pp=gAQBiAQB

https://www.youtube.com/watch?v=FPsv9FEhVd8&list=PLgfKCD7KgSTKFEXDIB9i203FY0o18rgB3&index=14&pp=gAQBiAQB

https://www.youtube.com/watch?v=rXjm-iKeIlM&list=PLgfKCD7KgSTKFEXDIB9i203FY0o18rgB3&index=17&pp=gAQBiAQB

https://www.youtube.com/watch?v=0OzIMgcpVMY&list=PLgfKCD7KgSTKFEXDIB9i203FY0o18rgB3&index=27&pp=gAQBiAQB

https://www.youtube.com/watch?v=P4wXGKiSXvE&list=PLgfKCD7KgSTKFEXDIB9i203FY0o18rgB3&index=37&pp=gAQBiAQB

https://www.youtube.com/watch?v=0XAFewziv5I&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=1&pp=gAQBiAQB

https://www.youtube.com/watch?v=1MPEBDP12Oc&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=2&pp=gAQBiAQB

https://www.youtube.com/watch?v=-WhK4eHQTM0&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=4&pp=gAQBiAQB

https://www.youtube.com/watch?v=gS_nHTWZEJ8&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=5&t=2212s&pp=gAQBiAQB

https://www.youtube.com/watch?v=UPkqFvjN-yI&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=9&pp=gAQBiAQB

https://www.youtube.com/watch?v=gQ1c1uILyKI&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=10&pp=gAQBiAQB

https://www.youtube.com/watch?v=MRMgd6E9AXE&list=PLgfKCD7KgSTJAWWZtVXG2CBQLl

BVt9Vvq&index=11&t=1s&pp=gAQBiAQB

https://www.youtube.com/watch?v=LDAIzpUJItg&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=12&t=16s&pp=gAQBiAQB

https://www.youtube.com/watch?v=1EnvkPf7t6Y&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=13&t=1158s&pp=gAQBiAQB

https://www.youtube.com/watch?v=fSpprRMqA_I&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=14&pp=gAQBiAQB

https://www.youtube.com/watch?v=jLRxwqtVDCY&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=15&pp=gAQBiAQB

https://www.youtube.com/watch?v=s6p1pIkJC6E&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=17&pp=gAQBiAQB

https://www.youtube.com/watch?v=P38H4g938rE&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=20&pp=gAQBiAQB

https://www.youtube.com/watch?v=Kqs7UcCJquM&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=21&pp=gAQBiAQB

https://www.youtube.com/watch?v=tGVAkwupH4g&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=30&pp=gAQBiAQB

https://www.youtube.com/watch?v=JAPnhroRROs&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=32&pp=gAQBiAQB

https://www.youtube.com/watch?v=FhTDkg4346U&list=PLgfKCD7KgSTJAWWZtVXG2CBQLlBVt9Vvq&index=35&pp=gAQBiAQB

https://www.youtube.com/watch?v=Hq58_yGJwHk&list=PLgfKCD7KgSTLx3u-nnLpCNhdyCXns4nHC&index=1&t=3s&pp=gAQBiAQB

https://www.youtube.com/watch?v=yBVyhBv9I0A&list=PLgfKCD7KgSTLx3u-nnLpCNhdyCXns4nHC&index=5&pp=gAQBiAQB

https://www.youtube.com/watch?v=Dc5VIf7hK9s&list=PLgfKCD7KgSTLx3u-nnLpCNhdyCXns4nHC&index=17&pp=gAQBiAQB

https://www.youtube.com/watch?v=Tje7ngjKRuQ&list=PLgfKCD7KgSTLx3u-nnLpCNhdyCXns4nHC&index=19&pp=gAQBiAQB

https://playbooks.flexibleir.com/soc-siem-use-cases/

https://drertugrulakbas.medium.com/is-siem-rocket-science-extraordinary-siem-use-cases-7ee2af4e2d5

https://drertugrulakbas.medium.com/detecting-unusual-activities-using-a-next-generation-siem-use-cases-d91f4e24b0f2

https://www.exclusive-networks.com/ie/wp-content/uploads/sites/19/2021/07/The-Essential-Guide-to-SIEM.pdf