International Journal of Scientific Research and Engineering Development-– Volume 8 Issue 4, July-Aug 2025
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Developing Base of the Modal SOC Integrated with Free & Open Source Software

W.M Ruwani Rathnayake
Graduate Student, School of Science and Technology, York St John University, London. & Security analyst @ Madhurashi Solutions Pvt (ltd)
Email: ruwani.rathna@gmail.com

----------**********----------

Abstract:

Emerging technologies are being developed in response to the increasing reliance on data and information across almost every company and organization today. As a result, it's become essential to monitor systems to identify policy violations, detect cyber threats, and assist with troubleshooting. Log data plays a crucial role in this process, as it provides visibility into system activities. Each application's logs give us insight into what's happening inside a system, and by collecting and analyzing these logs, we can effectively monitor for security events and anomalies. A Security Operation Center (SOC) is a centralized unit that is responsible for preventing, detecting, analyzing, and responding to cybersecurity incidents in real time. However, building a SOC from the ground up is a complex process involving the integration of multiple modules and functions. One of the biggest challenges is not just technical, it's financial. Traditional SOC solutions often require significant investment in commercial tools, licensing, infrastructure, and skilled personnel. The key focus of this research is to explore the possibility of developing a SOC using Free and Open-Source Software (FOSS). This approach offers a more cost-effective solution for organizations, especially those with limited budgets, the research highlights both the advantages and the challenges of using FOSS in SOC implementation, including aspects like integration complexity, performance tuning, and community support. Moreover, the study touches on the importance of Security Orchestration, Automation, and Response (SOAR). Finally, this research concludes with key findings and offers recommendations for future work, including how organizations can continue improving their SOC capabilities using open-source technologies while maintaining cost-effectiveness and operational resilience.

Keywords **— Security Operation Center, logs, prevent, detect, analyze, respond, cybersecurity incidents.**

----------**********----------

I. INTRODUCTION

The transformation of the world into the fourth industrial revolution can be observed daily, which almost completely depends on the use of modern technology. Technology has presented society with many advantages and opportunities that have enabled higher quality and improved convenience in many aspects of people's lives. Technology has become indispensable to the modern world. Nevertheless, there are pros and cons to this ongoing technological revolution.

The increased reliance on technology has led to the increase of cyber-threats and attacks. Conventional crime has evolved into cybercrime with the advent of computer technology. Cybercrime can be defined as illegal activities based on the Internet and Information Technology. A cyber-attack is a major part of a cybercrime which can maliciously disable computers, steal data or use a breached computer as

a launch point for further attacks. Bad actors use cyber-attacks to commit illegal activities and cyber-crimes which has resulted in cyber-attacks becoming a major threat to a world where data possess the highest value and impact on the society.

A. Security threats

Increasingly, businesses of all sizes are worried about cyber security concerns. These risks can manifest themselves in a variety of ways, including as malware, phishing scams, ransomware, and denial-of-service assaults. They may significantly affect a variety of organizations, such as:

- Financial losses: The cost of restoring damage, paying ransoms, or dealing with legal fees are just a few examples of the direct financial losses that can result from cyber assaults. Indirect financial costs like missed productivity, clients, and reputation can also result from them. [1]
- Cyber-attacks have the potential to interrupt corporate operations and result in lost profits and output. Additionally, they might harm a company's brand and make it challenging to draw in new clients and collaborators.
- Regulators may impose sanctions on companies for failing to safeguard sensitive data against cyberattacks. The penalties can be severe, and they can harm a company's reputation.
- Legal responsibility: Companies that have been hacked may be responsible for the harm that has been done. This can involve paying ransoms, fixing damage, and compensating victims.

B. SOC and SOAR

A SOC is a centralized mechanism within an organization that employs people, processes, and technology to continuously monitor for the purposes of preventing, detecting, analyzing, and responding to cybersecurity incidents and enhance an organization's security infrastructure. [2]

SOAR is based on the idea that an intelligent agent can use knowledge of the current state of the world, a set of operators that can be applied to that state, and a desired result or goal, to systematically generate a plan of action to achieve the goal. Here we use SOAR for automate the some of threat handling. [3]

Organizations would be able to detect threats quicker and react to them before they cause more harm to the organization because of the SOC. SOC also supports meeting regulatory requirements such as security monitoring, vulnerability management, and incident response.

SOC teams handle all the processes in a SOC environment. The SOC team aims to use a combination of technological solutions and a strong set of processes to identify, analyze, and respond to cybersecurity incidents. Security analysts, engineers, and managers manage security operations in SOCs [4]

C. Need for SOC solution

Security threats are a malicious act that seeks to damage data, steal data, or disrupt digital life in general. Different individuals and organizations have come up with different solutions to these cyber-attacks, because of the various range of security threats, varying desires and priorities, technological advancements, expertise and resources, collaboration, and the need for continuous adaptation contribute to the emergence of different solutions for different organizations and to the different cyber-attacks.

One of the main solutions is the method of identifying cyber threats and cyber-attacks and preventing and terminating the identified threats. That is, the threat hunting method can be introduced as the main method used for that. That is, a center set up by organizations and institutions based on a pre-threat identification system and methods to protect themselves from cyber-attacks and threats can be referred to as a SOC. A SOC is a centralized function within an organization employing people, processes, and technology to continuously monitor

and improve an organization's security posture while preventing, detecting, analyzing, and responding to cybersecurity incidents.

The success of a SOC depends largely on several factors including management, financial assistance, strategies, human, process, technology, environment, physical space, equipment, methods, tools, and continuous improvement. The purpose of this dissertation report is to identify the most critical factors for ensuring the success of SOCs and to highlight the critical elements for each of them. Also presenting the need to set up an SOC for an institution responsible for the security of a country and presenting the qualifications to be fulfilled for it. The research compares the research done before the establishment of the SOC and suggests the need for critical factors to ensure that the SOC works effectively and efficiently to provide protection against cyber-attacks. Employees at the SOC work closely with organizational incident management teams. Networks, servers, endpoints, databases, software, websites, and other systems are tracked and evaluated by SOCs, which search for suspicious behaviour that may indicate a security incident or compromise. The SOC is in charge of accurately detecting, examining, defending, investigating, and documenting possible security incidents. Cyber threat hunting is a major active defense strategy in a security operation environment. It is a security function that combines productive methodology, innovative technology, and threat intelligence to find and stop malicious activities.

Data is the key resource in the threat hunting method as data needs to be collected and analyzed for the threat hunters to hunt for threats. The data can be classified into three sections.

•Endpoint Data - Endpoint data comes from endpoint devices within the network.

•Network Data - This data got from Firewalls, Switches, Routers & DNS, Proxy Servers

•Security Data - This data got from security devices and solutions such as SIEM, IPS & IDS [5]

D. Research problem

Many organizations currently face challenges in effectively addressing cyber threats due to the absence of crucial fundamentals. The mentioned entities lack Risk Assessment and Management Incident Response Capabilities, neglect Third-Party Risk Management, and fail to implement Continuous Monitoring. SOCs play a vital role in addressing cyber challenges. To effectively address these areas of improvement, it is necessary to adopt a comprehensive and complete approach to cybersecurity. However, the majority of organizations lack the necessary financial resources and implies to establish and sustain a SOC. In this dissertation, I propose a solution to address the problem by creating a SOC platform exclusively utilising FOSS.

The research would be of interest to organizations, including private companies and government agencies, who are seeking to improve their cybersecurity posture and reduce their exposure to cyber threats. The findings of the research can contribute to the development of best practices and guidelines for the deployment and management of SOAR and SOC solutions using FOSS tools.

AI, Big data Analysis, and Machine Learning can be identified as one of the most improving technologies in the Cyber Security field. The field of study for this research is cybersecurity, particularly in the area of SOAR and SOC management. The research could involve the application of various technologies such as open-source software, scripting languages, automation tools, and cloud services in the cybersecurity domain. such big data analytics and AI techniques to develop Security Operation Centre with Security Orchestration.

The research could also cover areas such as incident response management, threat intelligence, vulnerability management, log management, and compliance monitoring. The field of study is interdisciplinary, involving aspects of computer science, information security, risk management, and business management.

E. Research Aim and Objectives

The research aims of "Developing SOC using FOSS tools" could be to investigate and evaluate the effectiveness of FOSS in building and managing a SOAR solution and a SOC. The research could involve designing and implementing a SOAR solution and SOC using FOSS and then evaluating the effectiveness of these tools in terms of their capabilities, performance, and scalability. The research may also compare the free and open source solution with proprietary solutions to identify the strengths and weaknesses of each approach.
Ultimately, the goal of this research would be to contribute to the development of cost-effective and efficient SOC solutions using FOSS tools, which can be deployed by organizations with limited budgets or those who prefer to avoid vendor lock-in.
The research objective of the proposed project could be to achieve the following goals:
1. Identify and evaluate suitable FOSS tools for building a SOAR solution and a SOC.
2. Develop and implement a SOAR solution and SOC using FOSS tools.
3. Evaluate the performance and effectiveness of the FOSS tools used in the SOAR solution and SOC in terms of their capabilities, scalability, and cost-effectiveness.
4. Compare the performance and effectiveness of the FOSS tools used in the SOAR solution and SOC with proprietary solutions.
5. Identify the limitations and challenges of using FOSS tools in building a SOAR solution and SOC and propose solutions to overcome these challenges.
6. The research objective is to contribute to the development of cost-effectiveness and efficiency.

F. Research questions that the project seeks to answer

The section of a dissertation that relates to the "research questions" the project intends to answer specifies the particular inquiries that the research project goals to address. The research questions in a dissertation are usually based on the research problem or objectives that have been identified.

1. How to develop this kind of platform only using open-source tools?
   In this section, it will be discussed in the chapter 3 of the tools that I intend on using in the process of developing the platform. Because a remedy for the affordability problem, all the technologies being used are open source.
2. How do we connect end nodes to this platform and get the network and system logs in real-time?
   Connecting the end nodes to the server is required in this scenario since it is impossible to perform live monitoring without doing so.
3. How to integrate threat intelligence?
   According to the proposal, it is essential for the SOC platform to integrate at least one or more forms of threat intelligence in order to improve the accuracy of threat discovery.
4. How do we create visualizations to see the output?
   The final step in the development process is the creation of the visualisation, and this platform will have the ability to generate visualisations based on the collected log data.
5. How do we alarm the third party in this prototype?
   The alarming function is essentially handled by visualisation, but I suggested adding a third-party alarming option in case someone is not present at the monitoring site. That would be helpful for getting immediate action in the event that severe attacks were recognised.

G. Research solution

The study aims to assess and analyse different open-source tools that are appropriate for specific parts of the platform. For instance, the research will consider the elasticsearch [6] tool for managing logs, the MITRE framework and NIST [7] for

security frameworks, Wazuh [8] for monitoring systems, and other relevant functionalities. The tools that have been chosen will be combined and adjusted in order to develop the platform, with a focus on making sure they work well together and perform at their best. Furthermore, this study aims to investigate techniques for effectively capturing and gathering system logs from the interconnected end nodes in real-time by utilising Wazuh agents. Appropriate logging frameworks or agents will be selected and incorporated into the platform architecture to enable easy retrieval and analysis of logs. The research will examine various open-source threat intelligence sources and APIs and assess their pertinence and efficacy for the platform. Additionally, this study will examine methods for effectively updating and managing threat intelligence data in order to ensure timely and comprehensive defence against new and evolving threats. The study aims to investigate whether open-source visualisation tools, such as Kibana [9], are appropriate for creating informative and easy-to-understand graphical displays. Different visualisation techniques, such as charts, graphs, and dashboards, will be explored in order to visually represent various aspects of the gathered data for monitoring purposes.

II. LITERATURE REVIEW

The literature review will consist of a thorough examination of existing literature on FOSS that are specifically relevant to the development of SOC. This chapter discusses the research conducted on the development of SOCs, the purpose of upgrading SOAR, the impact of severe cyber-attacks, as well as the solutions and challenges associated with developing SOCs using FOSS. The purpose of the literature review is to identify and deploy successful SOC using different FOSS tools that are capable of doing log management, threat detection, incident response, and visualisation.

A. Research was done on SOC development best practices

In recent years, there has been a significant increase in the number and sophistication of cyberattacks, making it challenging for organizations to defend against them. As a result, SOC and SOAR platforms have become essential for effective cybersecurity management. [10] This literature review aims to analyse and synthesize existing research on SOC and SOAR implementation to provide insight into the best practices for their effective deployment. [11]

The second paper to consider is "The Value of Security Operations Center" by Stamford, Conn (2017), which defines SOCs as "teams of security professionals that detect, analyse and respond to security incidents using a combination of technology solutions and a strong set of processes." The report highlights the benefits of having a SOC, including improved incident response time, reduced damage from security breaches, and increased visibility into the security posture of an organization. [12]

In "Security Orchestration, Automation and Response (SOAR) for Cybersecurity Operations" by National Institute of Standards and Technology [13], SOAR is defined as a "collection of software solutions and processes that enable organizations to streamline security operations and improve incident response by automating repetitive tasks and orchestrating workflows." The report suggests that SOAR can help organizations achieve better outcomes, including faster incident response, improved collaboration, and reduced human error.

One of the critical aspects of successful SOC and SOAR implementation is the integration of people, processes, and technology. The paper "Implementing a Security Operations Center: A Practical Guide" by Gartner [14] outlines the importance of building a cohesive team of skilled professionals, establishing well-defined processes, and using the right technology solutions to achieve an effective SOC.

Another critical aspect of SOC and SOAR implementation is selecting the right technology solutions. The paper "Security Orchestration and Automation: The Future of SecOps" by 451

Research (2018) suggests that SOAR solutions must be capable of integrating with various security tools, supporting a broad range of use cases, and providing robust reporting and analytics capabilities [15].

B. SOAR – Enhancing Security Operations

SOAR was developed by Allen Newell and Herbert A. Simon in the 1950s and 1960s to model human cognition and decision-making processes. SOAR is based on the idea that an intelligent agent can use knowledge of the current state of the world, a set of operators that can be applied to that state, and a desired result or goal, to systematically generate a plan of action to achieve the goal. The SOAR architecture is used in cognitive science and artificial intelligence research, as well as in the development of intelligent systems and robots.

In "Automation and Orchestration in Security: SOARing to Success" by Enterprise Strategy Group (ESG, 2019), the report highlights the importance of automating security processes to improve efficiency, reduce errors, and improve incident response time. [16] The paper suggests that SOAR can help organizations achieve these benefits by automating tasks such as incident triage, investigation, and containment.

In "A Review of SOAR Platforms: What are They, Why Do You Need One, and How to Choose?" by Cyber Security Hub (2020), the paper provides an overview of different SOAR solutions, highlighting their strengths and weaknesses. [17] The report suggests that organizations must choose a solution that aligns with their specific needs and security objectives.

The paper "Security Operations Centers and Automated Incident Response" by SANS Institute (2021) highlights the importance of integrating SOAR solutions with the overall incident response plan. The report suggests that SOAR can improve the efficiency of incident response processes by automating specific tasks, such as alert triage and remediation. [18]

In "Security Orchestration and Automation: Enhancing Security Operations" by Aberdeen Group (2019), the report highlights the potential ROI (Return On Investment) for organizations that implement SOAR solutions. The study found that companies that deployed SOAR solutions achieved a 23% reduction in time to remediate security incidents and a 20% reduction in the number of false-positive alerts. [19]

Another critical aspect of SOC and SOAR implementation is selecting the right metrics to measure the effectiveness of these solutions. 5 SOAR Myths Debunked paper mentioning the "Measuring the Success of Your SOC and IR Teams: A Guide to Security Operations Center Metrics" by Rapid7 [20]

C. Challenges of developing SOC platform only using Free and Open-source Tools.

As the field of cybersecurity continues to evolve, the need for effective security tools and processes is becoming increasingly important. One approach to enhancing cybersecurity is through the use of SOAR and SOC technologies, which help organizations to detect, analyse, and respond to security incidents more efficiently [21].

However, developing effective SOAR and SOC systems using FOSS tools presents a number of challenges. One challenge is the lack of interoperability between different FOSS tools. According to Löhr and colleagues [22], this can result in integration issues, which can make it difficult to achieve a seamless and effective system. They argue that this can be mitigated through the use of standardized data formats and application programming interfaces (APIs) to facilitate the exchange of data between different FOSS tools.

1) ***Skilled staff:*** Another challenge is the need for skilled personnel to manage and maintain FOSS-based SOAR and SOC systems. As noted by Rizk and Fakhereddine (2021), FOSS tools often require more technical solutions. This can result in higher staffing costs and a shortage of skilled personnel. *[23]*

2) ***Lack of documentation & support:*** Additionally, the lack of formal support and documentation for FOSS tools can make it difficult to troubleshoot issues and resolve problems in a timely manner. This is highlighted by Alqahtani and

colleagues (2021), who note that the lack of documentation and support for FOSS tools can lead to a slower response to security incidents and can limit the effectiveness of the system as a whole. *[1]*

3) ***Codes are available in public:*** Finally, there is a risk associated with the use of FOSS tools, as their source code is freely available and can be modified by anyone. This can make it difficult to ensure the security of the system, as malicious actors can potentially exploit vulnerabilities in the code. As noted by the National Institute of Standards and Technology *[24]*, it is important to carefully evaluate the security of FOSS tools and implement appropriate security measures to mitigate these risks.

D. Benefits of having SOC in a company

Implementing a SOC platform can provide several benefits to companies. Here are some of them:

1. Improved threat detection and response: SOC platforms can provide real-time monitoring and alerting of security incidents, which can help companies detect and respond to threats more quickly and effectively. By having a dedicated team of security analysts who can investigate and respond to incidents, companies can minimize the impact of security breaches and prevent future incidents.
2. Increased visibility and control: SOC platforms provide companies with greater visibility into their network and application infrastructure, allowing them to identify and address vulnerabilities more effectively. This can help companies reduce the risk of data breaches, malware infections, and other security incidents.
3. Enhanced compliance: SOC platforms can help companies meet regulatory compliance requirements by providing detailed reports on security incidents and demonstrating compliance with industry standards such as PCI-DSS and HIPAA.
4. Cost savings and Reduce Investigation Time: SOC platforms can help companies reduce the costs associated with security incidents by minimizing the impact of breaches and preventing future incidents. This can include reducing the cost of incident response, damage control, and legal fees.
5. Improved customer trust: By implementing a SOC platform, companies can demonstrate to their customers that they are taking security seriously and are committed to protecting their data. This can help build customer trust and enhance the company's reputation.

E. Research was done on SOC related software

In Resource Optimization for log shipper and processing pipeline in a large-scale logging system by Lertwuthikarn Described in his paper that filebeat is utilised as a log shipper to send the data to Logstash or to Elasticsearch, a server-side preprocessing pipeline. And the effectiveness of Filebeat and Logstash when used together can be influenced by a variety of customizable factors. [25]
A distributed full-text search engine called Elasticsearch expressly addresses the problems with scalability, huge data search, and performance that relational databases were never intended to handle. In 'Mining modern repositories with Elasticsearch' paper, reseachers examine Elasticsearch's benefits and drawbacks for using it to conduct research on contemporary mining software repositories. [26]
SCADA statistics monitoring using the Elastic stack by James Hamilton, In that paper, they highlighted that using a range of graphs, charts, and tables, Kibana is a web-based data visualisation application that connects with Elasticsearch to offer simple methods for navigating and visualising data. [27]
In paper of 'Agent-based ZeroLogon Vulnerability Detection' by S.V. Bezzateev and team describe that the Wazuh agent may respond to occurrences while adhering to legal requirements, such as group policies, and Wazuh is an open source system that carries out security monitoring and threat detection

duties. A Wazuh agent communicates with a Wazuh server, which it gives information about its configuration, detected intrusions, and abnormalities for additional investigation. [28]

III. RESEARCH METHODOLOGY

This chapter provides an overview of the framework, methodology, and the software, hardware, and technology that will be utilized in the development of the framework. Furthermore, it is important to place greater emphasis on the research solution that was briefly discussed in Chapter One. In his presentation, Mr. James Carder outlines a series of seven steps for establishing a SOC that can effectively identify and address threats, even without the presence of in-house staff. The framework will be further discussed in the this chapter. The discussion also covered various open-source tools such as Elasticsearch, Filebeat, Kibana, Wazuh, MISP, VirusTotal [29], and others.

A. Solutions for the research problems

The section will be discussed further of a dissertation that relates to the "research questions" the project intends to answer. it provides the framework, methodology, and the software, hardware, and technology that will be utilized in the development of the framework.

1) ***Develop platform only using FOSS:*** Adhering to a proper methodology and utilizing completely open-source tools can facilitate the creation and functionality of the prototype. The initial phase of data collection. It needs to be specify the requirements and objectives of the SOC platform. This may involve elements like managing logs, detecting threats, responding to incidents, and having reporting abilities. During the stage of tool selection, conduct an analysis of open-source tools that meet the specified criteria and several commonly used software options for SOC platforms are available like below.

- SIEM tools, such as ki Stack (which includes Elasticsearch, Logstash, and Kibana), Graylog, and OSSIM, are used for security information and event management.
- Tools such as OSSEC, elasticEDR, or Wazuh agent, or AV agent are utilized for Host based Security Monitoring.
- Tools for incident response and forensics, such as The Sleuth Kit, Volatility, and Autopsy.
- Tools for Threat Intelligence, such as MISP (Malware Information Sharing Platform) or TheHive.

When choosing tools, it is important to consider various factors such as community support, documentation, integration capabilities, and scalability. If the tools are not integrated or fully supported with our existing tools, we must explore other similar tools until we find a suitable one.

2) ***Connect end nodes to the platform:*** The process of integrating end nodes into a SOC platform requires creating a reliable and secure communication pathway between the nodes and the SOC platform. This facilitates the observation, examination, and reaction to security incidents. The approach used can differ based on the network structure, the characteristics of the end devices, and the needs of the SOC platform. I am utilizing the Wazuh agent, a lightweight tool, to facilitate host data transactions for the purpose of Host Intrusion Detection. When integrating other software with the platform, it may or may not be compatible with other tools. Therefore, I need to integrate different software that has similar functionalities.

3) ***Threat intelligence integration:***

The SOC platform uses threat intelligence data to create alerts and prioritize security incidents. Threat intelligence can aid in evaluating the severity, risk, and impact of detected events, allowing security analysts to prioritize the most significant threats. Incorporating threat intelligence into a SOC platform improves the organization's capacity to identify and address security threats with greater efficiency. The process of integration entails the inclusion of external threat intelligence feeds into the current infrastructure of the SOC platform. This is a brief summary of the procedures involved:

To obtain current information on emerging threats, vulnerabilities, indicators of compromise (IoCs), threat actors, and other security-related data, it is important to identify reliable and pertinent sources of threat intelligence. Various sources can provide threat intelligence, such as commercial providers, open-source intelligence, government agencies, industry-specific communities, and internal intelligence produced by the organization.

Data ingestion and processing refer to the procedures of collecting and analysing threat intelligence data. The SOC platform is set up to receive and handle this data. This refers to the process of creating data ingestion pipelines, converting and standardizing data formats, and assigning the relevant data to the corresponding fields in the platform. Custom parsers or pre-existing integration tools may be necessary to

accommodate the varying formats and structures of threat intelligence feeds.

Incorporate threat intelligence data into the detection and correlation mechanisms of the SOC platform through IoC matching and correlation. The process may include comparing IoCs, such as IP addresses, domains, hashes, or behavioural patterns, with network traffic, log files, or system events to detect possible security incidents or signs of compromise. The correlation rules may be tailored to meet the particular security needs of an organization.

To enhance the security data, it is recommended to incorporate enrichment processes that supplement the current security data in the SOC platform with significant threat intelligence information. Enrichment refers to the process of enhancing information by adding contextual details about known threats, associating threat actors or campaigns with detected activities, updating vulnerability information, or providing additional context for incident investigations.

4) ***Create visualizations:*** The Kibana visualization tool is utilized in the SOC to generate visualizations. The methodology comprises a series of steps.

- Establishing Kibana: Proceed with the installation and configuration of Kibana on the designated system. Kibana is commonly utilized as an integral component of the Elastic Stack, a comprehensive software suite that encompasses Elasticsearch for the purpose of data storage and retrieval.
- Kibana is dependent on Elasticsearch for the purpose of data storage and retrieval. Establish a linkage between Kibana and pertinent data sources, including but not limited to logs, events, or other data feeds. The process of enabling data indexing in Elasticsearch and subsequently facilitating its availability to Kibana can be achieved through configuration.
- To access the Kibana interface, one must open the Kibana web interface by navigating to the appropriate URL in a web browser. Kibana is typically configured to operate on port 5601, thereby enabling access to the application via the URL http://localhost:5601. An index pattern is a crucial component in Kibana that serves to identify the Elasticsearch indices that hold the data that is intended for visualization.
- Generate a graphical representation: Following the establishment of the index pattern, the process of crafting visualizations may commence. Kibana provides a diverse range of visualization options, including but not limited to bar charts, line charts, pie charts, maps, and other types. Choose the most suitable type of visualization according to the data that is intended to be represented.
- Adjust the visualization settings to match the requirements of the selected visualization type. The process entails the act of choosing the specific data fields to be represented in the plot, establishing the appropriate metrics or aggregations, defining filters, and personalizing the visual presentation and arrangement.
- Enhance the graphical representation. Refine the visual representation by modifying parameters such as axes, annotations, hues, and keys. Explore various arrangements to proficiently depict the data and communicate the intended information.

Upon achieving contentment with the visualization, proceed to preserve it within Kibana. This feature enables the user to employ the visualization repeatedly or disseminate it among their SOC team or wider organization. It is possible to generate dashboards by amalgamating various visual representations in order to offer a comprehensive perspective of the data.

5) ***Alarm the third party:*** To trigger an alert for a third party in a SOC platform, first identify the third party and identify the particular external entity or third party that requires notification or warning. Identify the conditions that initiate or activate a particular event or process. Specify the circumstances or occurrences that ought to activate the alarm for the third party. This can be determined by particular security events, risks, irregularities, or any other established standards. Then Identify the optimal method for notifying a third party. This may entail different modes of communication like email, phone, SMS, instant messaging, or specialised incident management systems. Here I use same platform and telegram chatbot. *[30]* To ensure prompt and consistent response times, it is advisable to automate the alarm process whenever feasible.

B. Research Framework

The process of developing a SOC could vary depending on the requirements of the organization. Mr. James Carder introduces seven steps to build a SOC to detect and respond to threats without in-house staff using commercial tools. As he's a view I am following possible methodology according to my scenario.

1. Build a plan for your SOC.
   - Limitation of scope
   - Core functions and non-core functions

2. Assemble the team.
   - Assign roles and responsibilities to each of member.
3. Establish policies and procedures.
   - Develop comprehensive security policies and procedures that govern SOC operations, data privacy, employee responsibilities. And ensure these policies comply with industry regulations and standards.
4. Select SOC tools and Technology.
   - Choose appropriate tools and technologies.
   - Integrate selected tools to enable data sharing and correlation.
5. Data collection and integration
   - Collect relevant data from the sources.
   - Integrate other tools and threat data feeds to the SIEM.
6. Implement use cases and detection rules.
7. Make Visualizations.
8. Training and Awareness.
9. Monitor & responds.
10. Maintain and improve your system.

For this prototype, FOSS were used instead of commercial tools. So, the scope of the project was somewhat narrowed by the capability of the tools used. But it was possible to get a significant output from this developed platform.

This report will present the main discoveries and insights obtained during the creation of a SOC through the utilization of open-source tools. The objective of this project was to create a budget-friendly SOC solution using open-source tools. This solution was intended to improve security monitoring, detect threats, and enhance incident response capabilities. Throughout the course of the project. In the first stage of my project, I methodically chose open-source tools that are appropriate for constructing a proficient SOC. The study involved thorough research and assessment of different tools based on factors such as functionality, community support, flexibility, and integration capabilities. The results showed that there are many open-source tools available, including popular ones like Suricata, beats, OSSEC, ELK Stack OSSIM, Alien Vault, The Hive, and others. I must explore all available tools to determine which one is suitable for my architecture.

I created a multi-tier architecture for our SOC that included data collection, analysis, and visualization components. The study showed that open-source tools can be used to combine various data sources, such as logs, system logs, and threat intelligence feeds. After deploying and configuring the chosen tools in the architecture, the data flow and processing of security events were made seamless and efficient.

The main goal of this SOC was to improve the ability to identify potential security risks. Through the use of open-source tools that have advanced detection mechanisms, such as Wazuh, OpenDistro, and Virus Total, it is possible to identify and analyze potential threats in an effective manner. The tools are capable of detecting both known and unknown threats in our monitored network environment through signature-based detection and behavioral analysis. Subsequently, the SOC endeavors to optimize incident response procedures. However, due to the absence of necessary hardware and a suitable testing setting, I am unable to execute this to its fullest extent.

Although I was able to proceed with the project using a cloud-based approach, it is important to recognize specific factors and constraints. The use of cloud services creates a dependence on internet connectivity, which can affect the availability and responsiveness of the SOC. Additionally, it is important to consider the expenses related to the use of cloud services, as there may be recurring costs based on the selected cloud provider and resource consumption. To sum up, the absence of tangible servers and limitations in networking impeded the creation of a physical model for our SOC at first. However, the use of cloud services offered a feasible substitute. By implementing the tools on the cloud, I was able to deploy and

configure them successfully. This enabled me to monitor security, detect threats, and respond to incidents effectively. This project showcases how open-source SOC solutions can be adapted to various deployment environments, allowing organizations to use cloud services to overcome infrastructure limitations. However, it is important to note that there are still considerations and limitations associated with the use of cloud services. SOCs usually comprise multiple specialized units that are accountable for different facets of cybersecurity and security operations. The composition of units within a SOC can differ based on the organization and its requirements. However, several standard units are commonly present in a typical SOC. The figure 4 is shown Gunther Pernul, Manfred Vielberth, Fabian Böhm, and Ines Fichtinger According to their findings, a SOC team consist of several levels of SOC analysts and specialized managers.

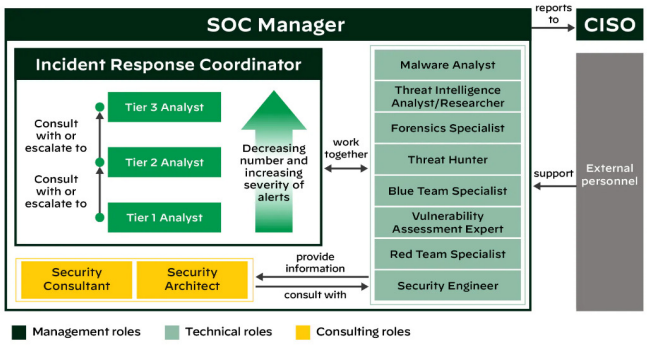

Fig 1: SOC team **[31]**

- The Security Incident Response Team (SIRT) is tasked with the responsibility of identifying, examining, and addressing security incidents. They are responsible for monitoring security events in real-time, analyzing possible threats, and coordinating activities related to responding to incidents.
- Security analysts are professionals who examine security alerts and logs produced by security systems. They also engage in threat-hunting activities and carry out vulnerability assessments. They collaborate closely with the Security Incident Response Team (SIRT) in order to identify and address security risks.
- The Threat Intelligence Team is responsible for gathering and examining data on existing and upcoming threats. This includes studying threat actors, their strategies, and any weaknesses they may exploit. These insights are valuable to the SOC as they assist in maintaining a proactive and well-informed approach to identifying potential risks.
- The forensics team has expertise in the field of digital forensics and incident response. The individuals in question engage in the examination of security incidents, the collection and examination of evidence, and the provision of assistance for legal and compliance objectives. Additionally, they can aid in the process of recovering and restoring systems that have been compromised.
- The Malware Analysis Team is tasked with the responsibility of examining and deciphering harmful software, commonly known as malware. Researchers analyze the behaviors, capabilities, and potential consequences of malware in order to create countermeasures and strengthen the organization's defensive strategies.
- The management of the SOC involves supervising its overall operations, team management, and ensuring that security operations are in line with the organization's goals and objectives. They are responsible for managing the distribution of resources, enhancing operational efficiency, and collaborating with other departments or external organizations.

- Security engineers are responsible for the design, implementation, and maintenance of security systems and technologies within the SOC. The individuals engage in the configuration of security tools, implementation of network and system monitoring solutions, and creation of security architectures in order to safeguard important assets.
- The Compliance and Audit Team is responsible for ensuring that the activities of the SOC are in accordance with applicable regulations, industry standards, and internal policies. Regular audits are conducted, security controls are assessed, and assistance is provided in maintaining compliance with data protection laws.
- The Threat Hunting Team adopts a proactive approach to detect and examine potential threats prior to their occurrence as security incidents. The organization actively seeks out signs of compromise, unusual behavior, or weaknesses in its network and systems.

It is worth mentioning that the structure and units within a SOC can differ depending on factors such as the organization's size, industry, and security needs. Different organizations may choose to reorganize these units in various ways based on their specific requirements and available resources.

C. System Development Methodology

Develop SOC prototype was selected as a software development methodology because it enables an iterative design process. Prototyping enables the program to be updated often while being examined for updates and features. Consumer testing follows the creation of a prototype, which is then updated once more before being used as the final product. The prototype-making process will involve users right from the start. It is possible to detect functionality that has not been developed and ensure user satisfaction thanks to user evaluation of the prototype.

To complete this dissertation, data, and information will be gathered through interviews, reports, web sources, and documentaries and then examined. To choose the optimal approach, information about earlier work will be examined in particular.

D. Hardware requirements

This table summarizes the overall requirements, infrastructure, and services that have been implemented in my project. In addition to this, it is possible to deploy it in more extensive environments by using the appropriate configurations and scalable infrastructure.

When it comes to end points, it is possible to link any device that generates logs with the server machine. For example, routers, switches, and so on are all examples of such devices. And in enterprise level we have expand the hardware requirement based on the scenario.

TABLE I

Physical servers/ cloud		
Requirement	**Server**	**agent**
Memory (min)	8 GB memory	4 GB memory
OS	Ubuntu 18.04	Windows 10
Open ports	Wazuh: TCP 1514, 1515, 55000 Kibana: TCP 5601, 443 Elasticsearch: 9200 SSH: TCP 22	RDP: TCP 3389
services	Filebeat Elasticsearch Wazuh-api Wazuh manager MITRE Attack framework plugin Virus total	Telegram Wazuh agent

E. Proposed design

The proposed or presented Model SOC was being carried out in accordance with the figure below. As per the figure 7 architecture, agents that have been deployed will gather logs from the host machines and then send them to the Wazuh

manager for analysis after parsing. Subsequently, elasticsearch retains the alerts and logs in the JSON data format. The network logs will be collected and stored in Elasticsearch. This will enable the collection and analysis of logs using Beats, Kafka, and Logstash. The Malware Information Sharing Platform (MISP) collaborates with Cortex and provides feeds on threat intelligence. It will analyze and respond to threats. The Hive is a platform designed for responding to security incidents. It helps manage and investigate these incidents in an efficient manner. The results will be generated utilizing the Kibana software.

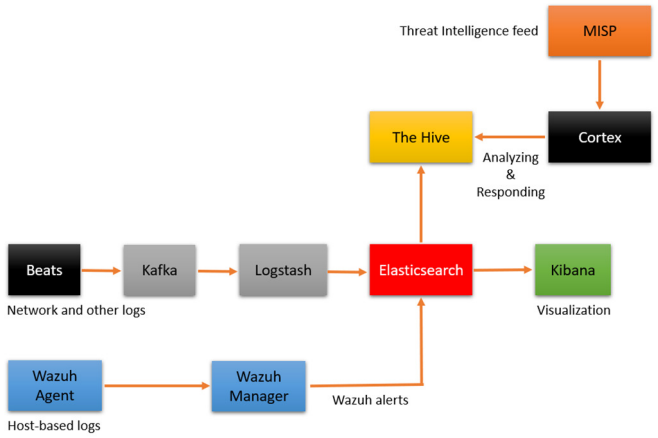

Fig 2: Proposed architecture

F. Developed design

According to the below design in figure 8, deployed agents will collect logs from the host machines and send them to Wazuh manager to be analyze d. Then, use beats to read the logs and send them to Elasticsearch. The alerts and logs should be saved in JSON format. Like Indicators of Compromise (IOCs), CDB list and MITRE attack frameworks are used here as threat data feeds. And it will look for threats and figure out how to stop them. Also, Virustotal checks the host machine for malicious material and files and sends the results to Elasticsearch. Using the Kibana tool, the results will be made at the end. By configuring yml files, I can also send an telegram message to a third party to alarm them.

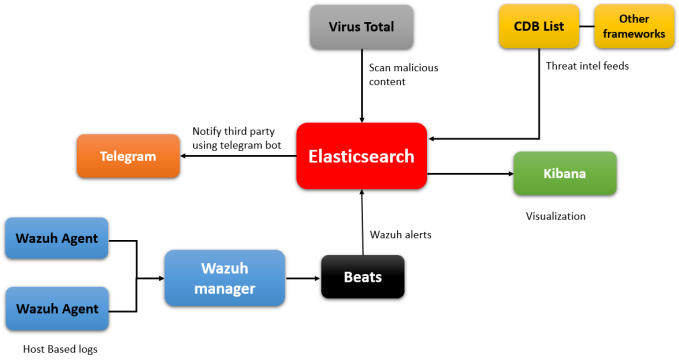

Fig 3: Developed architecture

G. Design Goals

The utilization of open-source tools in the implementation of SOC provides a multitude of benefits such as economic feasibility, adaptability, and communal backing. In this section, defines the fundamental objectives of design when employing open-source tools for a SOC implementation.

- The selected tools must possess the ability to manage the escalating quantity of security events and logs in tandem with the organization's expansion. The concept of scalability pertains to the ability of the SOC infrastructure to accommodate a growing number of monitored assets and efficiently manage the surge of data produced by diverse security systems.
- The utilization of open-source tools affords organizations the benefit of adaptability and personalization, enabling them to customize the SOC to their particular needs. It is imperative that the tools utilized possess the capability to seamlessly integrate with diverse security systems, network devices, and log sources, thereby facilitating an all-encompassing view of the organization's infrastructure. The availability of customization options empowers the SOC team to tailor the tools to their specific workflows and processes, thereby optimizing their efficiency and efficacy.
- The implementation of comprehensive monitoring is an essential component of a SOC

to ensure optimal performance. The chosen open-source tools must possess sturdy functionalities for the purpose of monitoring and scrutinizing security occurrences, records, network flow, and system conduct. It is recommended that the system provides instantaneous visibility into the assets of the organization, promptly detects potential security incidents, and facilitates the correlation and analysis of security data from various sources.

- The integration of threat intelligence feeds into the SOC infrastructure can significantly improve an organization's capacity to identify emerging threats in a proactive manner. It is imperative for open-source tools to facilitate the integration of threat intelligence platforms, thereby enabling prompt updates on the most recent threat indicators, vulnerabilities, and attack patterns. The integration facilitates the ability of the SOC team to make data-driven decisions and implement preemptive measures to address potential security threats.
- The provision of comprehensive reporting and analytics functionalities by open-source tools is imperative for the generation of significant insights from security data. SOC teams have the ability to utilize these technological resources to construct reports that are suitable for executive-level audiences, conduct trend analysis, and evaluate critical performance metrics, commonly referred to as key performance indicators (KPIs). The implementation of proficient reporting and analytics mechanisms is instrumental in facilitating the perpetual enhancement of the SOC activities and the organization's comprehensive security stance.

IV. RESULTS AND DEMOSTRATION

According to the methodology, sequential procedure used to set up the environment using VMware and performed the installation order by Ubuntu 18.04 [32], setting up the Wazuh server, installing Elasticsearch, Filebeat, and Kibana, authenticating the agents with the server, deploying agents to the server, integrating VirusTotal, and configuring a Telegram bot for third-party alerts. The software needs to be configured and integrated in order to obtain results from the platform. The provided figure in the results chapter will serve as evidence of the platform's successful installation up to this point.

This section is dedicated to evaluating the efficiency and effectiveness of the SOC in accomplishing its desired goals and objectives. Additionally, the results section provides further details regarding the process of collecting and analyzing the data in the study. This demonstrates the efficiency of the data collection methods that have been put into place, using agents, in gathering security event data from various sources. This section also discusses the findings of the data analysis, highlighting the SOC's capability to effectively correlate and analyze the gathered data in order to promptly identify potential threats. Furthermore, the results section of the study presents the findings obtained from the reporting and visualization activities. The generated security reports and metrics are presented to demonstrate their usefulness in providing valuable insights to analysts and aiding informed decision-making. This section also discusses the efficacy of data visualization tools in presenting security information in a clear and practical manner.

A. *Dashboard*

My Ubuntu server's IP address is 192.168.50.128, and it's also where I installed the Kibana server. Therefore, the above IP address can be used to visit the Kibana login page, and a "admin" user login is required to view all dashboards. The implementation dashboard will be explained and discussed in detail in the following sections.

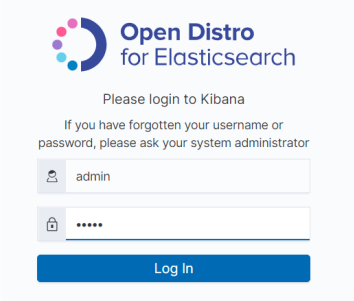

Fig 4: Login to Kibana

*1) **Skilled staff: Setting up new monitoring group to the platform:*** If there are multiple hosts that need to be investigated, it would be convenient to divide them into groups. This way, monitoring can be effectively managed on a group basis. Management panel give feature to add new groups to the platform as like the following figure 5.

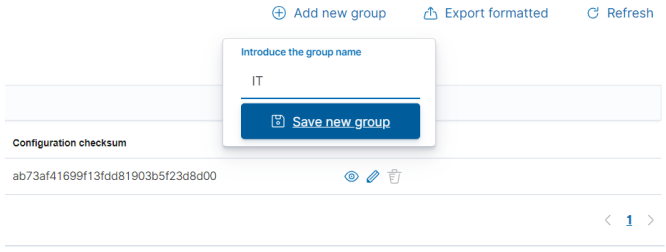

Fig 5: Add monitoring groups

According to the figure 6, IT (Information Technology) and HR (Human Resource) group were created as test groups for the monitoring purpose. Based on this groups, I can add few agents under those monitoring groups to do the test run.

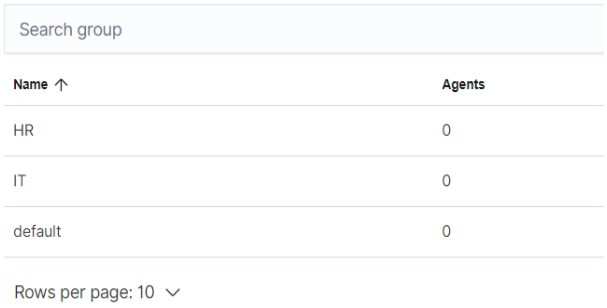

Fig 6: List of groups

*2) **Agent Overview:*** This dashboard figure 7 provides information on the status of agents, agents' evolution over time, agent connectivity details, and specific details about each agent such as IP address, given name, dedicated groups, and operating system version.

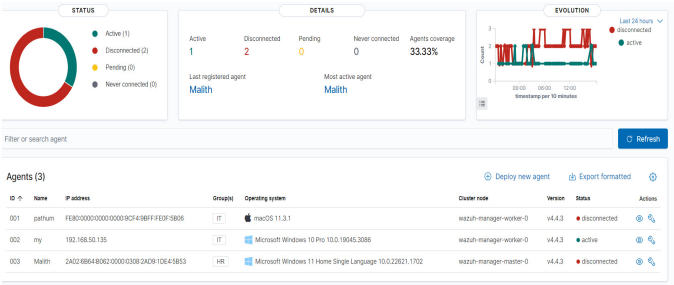

Fig 7: Agent overview dashboard

Authentication failures events can be identified in the analyze d log set, as the same logon failure is indicated by the visualization shown in Figure 8. This login failures indicate all the failure logging with the OS platform.

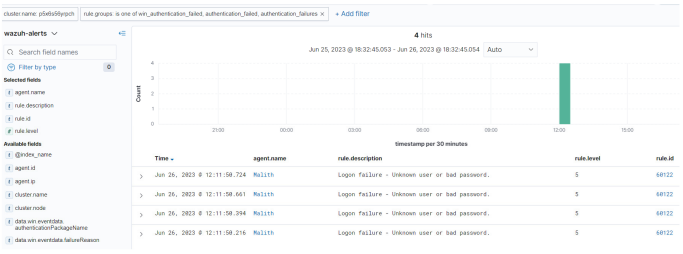

Figure 8: Logon failure events

Following figure 9 display the graphical approach of same event that figure 8 shows that specific to the agent called “Malith”. Following table discussed the logging familiar evet categorizing into time frame, agent, agent name, what MITRE tactics used to identify them, description of the event, threat level of the event and rule ID.

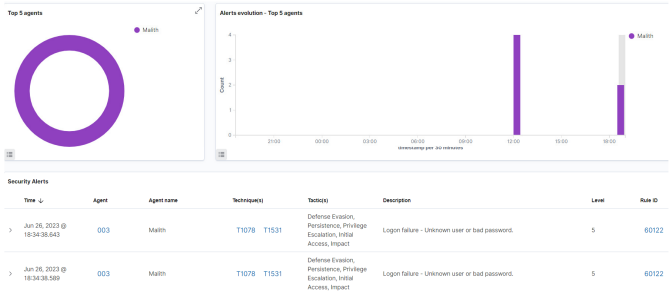

Fig 9: Logon failure visualizations

***3) Security event summary dashboard*:** Figure 10 and 12 shows the security event summary dashboard of the platform. According to the figure 67, dashboard displays that 6182 of total number of events, 445 of authentication success, 15 of authentication failures, and 90 of alert count above level 12. Then the alert level evolution chart shows the all the events according to the alert levels in same figure. The 'Top MITRE ATTACKS' doughnut chart shows the stored logs according to MITRE ATTACK framework.

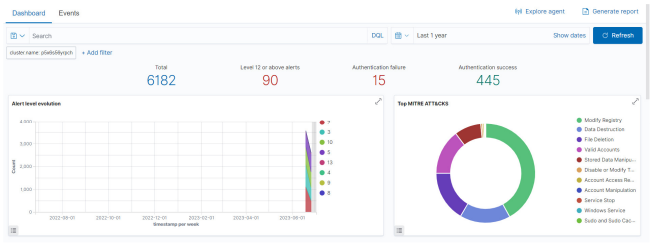

Figure 10: Event summary dashboard 1

Wazuh manager classify that it rules according to the threat levels as like in figure 11 [8]. According to their classification 0 to 11 level has low and moderate threat level and 12 to 15 has severe threat level and need to take immediate action to those alerts.

Level	Title	Description
0	Ignored	No action taken. Used to avoid false positives.
2	System low priority notification	System notification or status messages. These have no security relevance.
3	Successful/Authorized events	These include successful login attempts, firewall allow events, etc.
4	System low priority error	Errors related to bad configurations or unused devices/applications. These have no security relevance and are usually caused by default installations or software testing.
5	User generated error	These include missed passwords, denied actions, etc. By themselves, these have no security relevance.
6	Low relevance attack	These indicate a worm or a virus that have no affect to the system (like code red for apache servers, etc).
7	"Bad word" matching	These include words like "bad", "error", etc.
8	First time seen	Include first time seen events. First time an IDS event is fired or the first time a user logged in. It also includes security relevant actions (like the starting of a sniffer or something like that).
9	Error from invalid source	Include attempts to login as an unknown user or from an invalid source. May have security relevance (specially if repeated).
10	Multiple user generated errors	These include multiple bad passwords, multiple failed logins, etc. These may indicate an attack or may just be that a user just forgot his credentials.
11	Integrity checking warning	These include messages regarding the modification of binaries or the presence of rootkits (by Rootcheck). These may indicate a successful attack. Also included IDS events that will be ignored (high number of repet
12	High importance event	These include error or warning messages from the system, kernel, etc. These may indicate an attack against a specific application.
13	Unusual error (high importance)	Most of the times it matches a common attack pattern.
14	High importance security event	Most of the times done with correlation and it indicates an attack.
15	Severe attack	No chances of false positives. Immediate attention is necessary.

Fig 11: Alert level classification

Figure 12 is a also part of the Security event summary dashboard and it shows the alert evolution according to the registered agents, and the doughnut chart display the top 5 agents that collecting logs from the server. Following 'Security Alerts' table display the same events from the above graphical approaches that related to the deployed agents. Following table discussed the logging familiar evet categorizing into time frame, agent, agent name, what MITRE tactics used to identify them, description of the event, threat level of the event and rule ID.

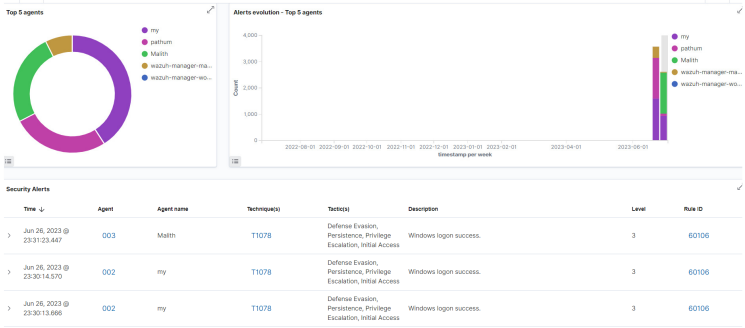

Fig 12: Event summary dashboard 2

*4) **Integrity monitoring dashboard:*** Integrity monitoring dashboard will display that the any changes or corruption of file or data like in figure 13. Event summary line chart displays the total number of increased events according to the time frame of all agents. Currently I only have logs from the 2 agents called 'Malith' & 'My'. Rule distribution chart shows the registry value entry changes in deferent agents. Actions pie chart shows the action that has been done during the time frame. It can be file added, modified, or deleted. Top 5 user table shows the user that responsible for above actions.

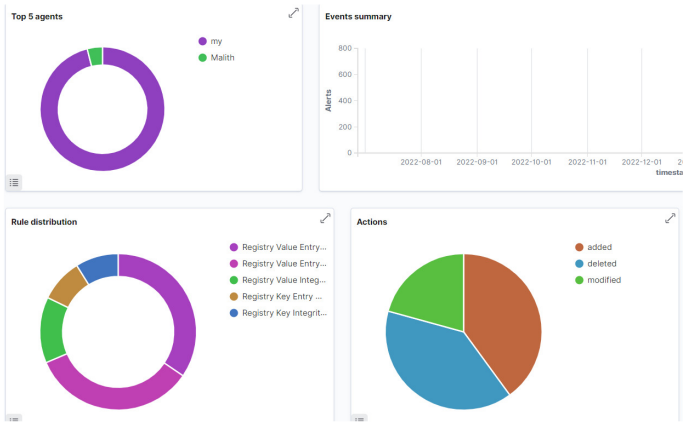

Fig 13: Integrity monitoring dashboard

I accidentally discovered a trojan version of a file on one of the agents using this platform. This finding is shown in Figure

14. A file that is linked to a Trojan horse, also known as a Trojan, is referred to as a Trojan-related file. A Trojan is a form of malicious software that pretends to be a genuine or harmless file or program in order to trick users and obtain unauthorized entry to their systems. Based on the figure provided, a trojan file has been detected on Pathum's laptop within the relevant folder.

Fig 14: Detected trojan file in agent "pathum"

5) Vulnerabilities of agents: Following figure 15, shows the identified vulnerabilities in deployed agents. First doughnut charts display the severity of the identified vulnerabilities as Critical, High, Medium, Low. Same details visualize from the details board. Summary chart displays the identified vulnerability names according to figure 15. I found on vulnerability VLC media player and 454 of vulnerabilities found on Windows 11 OS. The table shows the vulnerability details on related agent. According to the table I can identify what system is vulnerable, its version, architecture, threat severity, related CVE number, CVSS3 source and vulnerability detected time.

The acronym CVE stands for Common Vulnerabilities and Exposures. The Common Vulnerabilities and Exposures (CVE) system is a unique identifier utilized by various vendors, including Adobe and Microsoft, to categorize specific security vulnerabilities [33]. These vulnerabilities are addressed through the provision of patches as a means of resolution. For example, each page of a book is assigned a unique number, which facilitates the efficient retrieval of information from specific pages.

In general, CVE numbers are typically presented in the format of CVE-nnnn-nnnn. This will reveal numerous vulnerabilities. Given that the CVE number is not owned by any particular software vendor, clients can have confidence in receiving comprehensive protection. The database serves as a neutral platform where vendors can openly disclose their vulnerabilities without any bias.

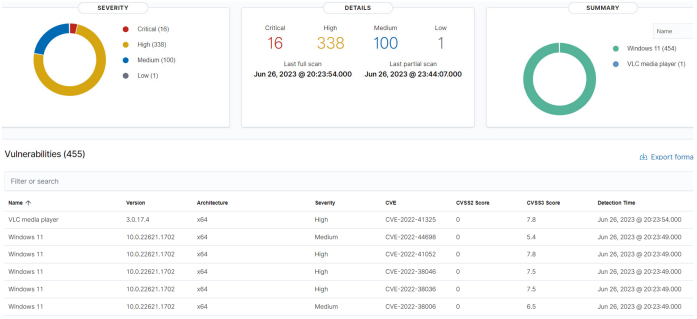

Fig 15: Vulnerabilities of the agents

6) MITRE attacks frameworks: As I discussed about the MITRE ATTACK framework in above chapter. Following figure 74 shows the identified MITRE ATTACK events in the resisted agents though the time. The purpose of this dashboard is to give a comprehensive summary of the current condition of my infrastructure in relation to recognized adversarial Tactics, Techniques, and Procedures (TTPs) as outlined in the MITRE ATT&CK framework. The dashboard presents important metrics, including the overall count of events and alerts, as well as a summary of the top 10 Tactics, Techniques, and Procedures (TTPs) identified in my system. These indicators can be utilized to evaluate the efficiency of current security measures and pinpoint areas that may necessitate additional focus.

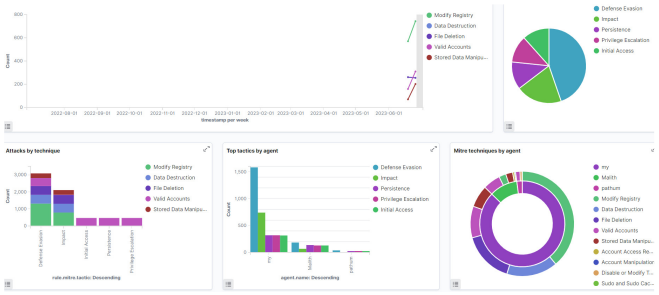

Fig 16: MITRE Framework dashboard

B. Virus total results

In following dashboard on figure 17, I identified 470 of positive malicious data and file in my registered agents though the virus total platform, based on the logs I identified that was from the same agent who had the Trojan files. Scanned file count can be seen in following doughnut chart and the table. Also, File directory can be located in the given table in figure 17.

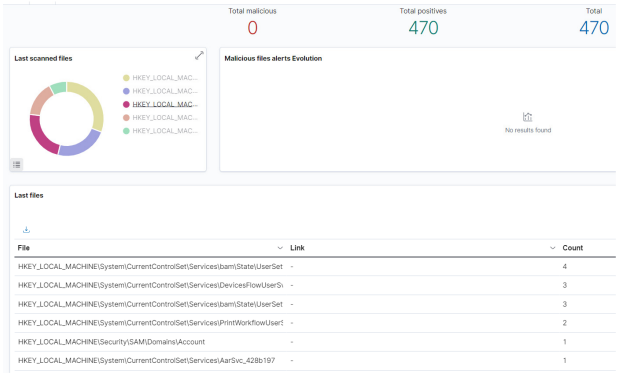

Fig 17: VirusTotal findings

C. *Telegram Alerts*

Purpose of the sending alerts to the chat application will be easy to act fast when some threats identified. The chat notification feature in a continuous monitoring system plays an important role for preserving the system's reliability and enabling prompt responses to incidents. It also encourages teamwork, facilitates communication between the monitoring team and other involved parties. Enhancing overall operational efficiency and minimizing the risk of extended periods of downtime or disruptions. here figure 18 displays the first few alerts that generated from the developed platform.

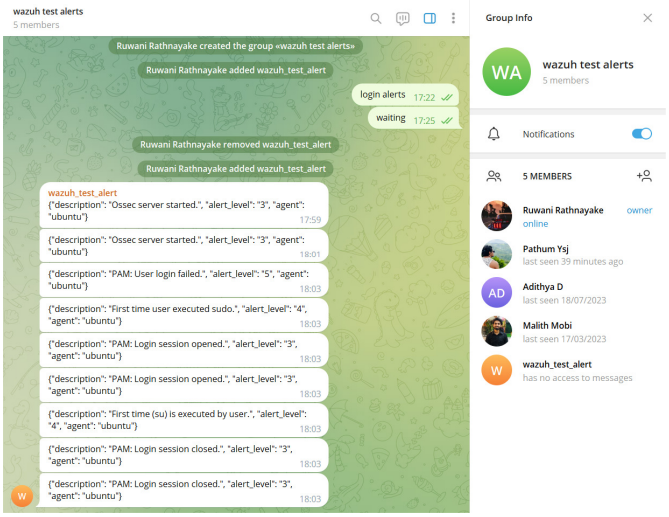

Fig 18: Telegram alerts

D. *Auto generated report*

The platform offers users the option to generate reports based on the aforementioned findings as a default feature. Figures 19 and 20 The exact same findings that appeared in the previously mentioned dashboard report will appear in this report as well.

Fig 19: Visualizations in generated report 1

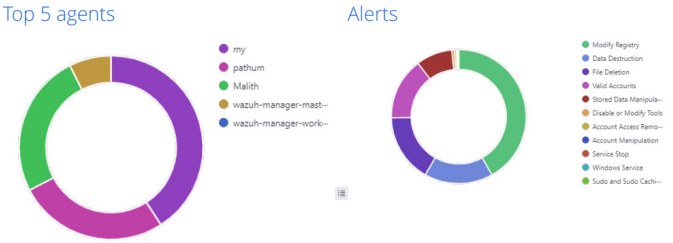

Fig 20: Visualizations in generated report 2

V. LIMITATION AND FUTURE IMPLEMENTATION

Relying exclusively on open-source tools to construct a SOC has a drawback in terms of insufficient vendor support and accountability. Open-source tools are usually created and managed by a group of unpaid individuals, which can result in insufficient resources for continuous improvement, error correction, and assistance. Possible delays may occur in obtaining prompt updates, patches, or support for addressing critical security incidents or emerging threats. The anticipated constraints of this project are outlined below.

- Tools and software that are not compatible with each other. Establishing a SOC that relies exclusively on open-source tools may present difficulties in terms of compatibility

with other proprietary tools or software utilized by the organization. Difficulties with integration and interoperability may occur when trying to link open-source tools with proprietary security solutions, which could restrict the SOC's capability to utilize the complete array of security technologies that are accessible.

- Open-source tools typically depend on development driven by the community, which can lead to inadequate vendor support and accountability. The SOC team may experience difficulties in obtaining timely support, bug resolutions, or updates when they encounter problems or vulnerabilities. This can affect the effectiveness of incident response and impede the SOC's capacity to keep abreast of new threats.
- Open-source tools may have insufficient documentation and unreliable sources, which can make it difficult to comprehend their functionalities, settings, or optimal methods. The absence of trustworthy sources may impede the SOC team's capacity to efficiently implement, set up, and sustain the tools, resulting in possible operational inefficiencies and security vulnerabilities.
- The construction and upkeep of a SOC utilizing open-source tools necessitates specific knowledge and expertise. The SOC team should have the essential abilities to set up, adapt, and enhance the tools for their particular setting. Recruiting and keeping skilled staff who are knowledgeable in open-source tools can be difficult due to the high demand for these professionals in the industry.
- The ecosystem is fragmented: Within the open-source landscape, there are various tools available, each having its unique advantages, disadvantages, and community backing. Selecting the appropriate tools for the SOC can be a complicated process that involves careful assessment and experimentation. The complexity of integration, interoperability, and maintenance efforts can be increased by managing a fragmented ecosystem of tools.
- Open-source tools may have limited features, functionality, and scalability compared to proprietary tools. Open-source solutions may not provide advanced capabilities or specialized security features, which can restrict the SOC's ability to handle complex or specific security requirements.

It is noteworthy that although there are certain limitations, organizations can overcome them by investing in skilled resources, establishing strong community support, implementing effective integration strategies, and ensuring comprehensive documentation and training. Open-source tools can be a useful basis for constructing a SOC, particularly when combined with thoughtful preparation and adaptation to meet specific needs.

As the project progresses, potential future implementations have been identified.

Develop protocols for various incidents and identify the appropriate tools to be used for each. Open-source automation frameworks can be utilized to streamline repetitive and time-consuming tasks, such as log analysis, threat hunting, and incident enrichment. This approach involves using open-source tools to proactively search for potential threats and perform advanced analytics. Utilize machine learning libraries, frameworks for log analysis, and tools for visualization to detect patterns, irregularities, and possible security threats. Create data analytics pipelines to analyze the security data and obtain useful information.

To enhance collaboration, it is recommended to work with open-source vendors and service

providers who provide commercial support for the chosen tools. Forming partnerships can guarantee access to professional support, timely software upgrades, and security fixes. Engage in partnerships with vendors and service providers to meet particular needs, make contributions to improve product features, and have an impact on the product's development plan. Organizations can construct a strong SOC by utilizing open-source tools through the implementation of forthcoming measures. This method allows for adaptability, affordability, and personalization while also guaranteeing adequate assistance, incorporation, and proficiency enhancement to optimize the efficiency of the security measures. Also create a complete documentation database for the open-source tools is important when utilizing the SOC. The document should contain instructions for installation, examples for configuration, tips for troubleshooting, and scenarios for usage. It is important to keep the documentation up-to-date and well-maintained in order to ensure its accuracy. A knowledge management system should be established to record insights, lessons learned, and standard operating procedures that are relevant to the SOC's environment.

Investing in training programs can enhance the skill set of the SOC team members with regards to the open-source tools used for skill development. Offer chances for ongoing education, official credentials, and practical training sessions. Establishing in-house knowledge and skills is crucial for optimizing the use and upkeep of tools.

In terms of future directions, this project has yielded valuable insights into the creation of a SOC through the utilization of open-source tools. It would be worthwhile to explore the potential of other SOCs in this context. In light of my discoveries, forthcoming research could concentrate on the subsequent elements:

- The SOC will be tested and evaluated in a real-world setting to determine how well it can handle increased demand, use resources efficiently, and accurately detect threats.
- The study aims to investigate the incorporation of supplementary open-source tools and the enhancement of SOC's capabilities. This includes the integration of machine learning methods for detecting anomalies and conducting advanced analytics.
- The SOC's automation capabilities will be improved by integrating orchestration tools to simplify incident response processes.
- Collaborating with external entities or communities can provide the SOC with diverse threat intelligence feeds, thereby enhancing its detection capabilities.

This research involves the analysis of data, hence that approach was chosen. Build precise security solutions that are used in this study to combine cyber solutions on a single platform with AI alternatives to automate them. Therefore, it is essential to test the system to find cyber risks in line with the structure. In the future, it is important to investigate whether it is possible to put the SOC into practice in a real-world setting when resources are accessible.

VI. CONCLUSIONS AND RECOMMENDATIONS

The final chapter of the project will discuss the achievements of the research goals and objectives, the knowledge gained from this project, the acquisition of new skills, and the enhancement of existing abilities. The objective of this study was to establish a SOC utilizing FOSS tools. This methodology has the potential to enhance the security infrastructure of an organization, even when faced with financial constraints.

The initial chapter of this study explored the importance of adopting a threat incident management platform to effectively address the increasing cyber threats that exist in our modern world. I also discussed the problem of companies facing a scarcity of resources and financial assistance. The second chapter discusses the

literature reviews on the development of effective SOAR and SOC systems using FOSS. These reviews highlight several challenges that arise in this process, such as problems with interoperability, high staffing costs, limited support and documentation, and potential security risks and their mitigation. In the third chapter, an overview is given of the framework, methodology, and the software, hardware, and technology that will be used in developing the framework. Additionally, it is crucial to prioritize the research solution that was briefly mentioned in Chapter One. The fourth chapter provides a detailed explanation of the step-by-step process used to set up the environment using VMware and install the software components, namely Elasticsearch, Kibana, Wazuh, and Filebeat. These components were configured and integrated to enable the platform to generate desired results. The fifth and final chapter of the study presented the effectiveness of the implemented data collection methods, which involved the use of agents, in acquiring security event data from different sources. This section also presents the results of the data analysis, emphasizing the ability of the SOC to accurately connect and analyze the collected data in order to quickly identify possible threats.

It is important to acknowledge that open-source SOC platforms have the potential for extensive customization. However, it is necessary to possess technical proficiency in order to successfully set up, configure, and maintain these platforms. Organizations or individuals who consider the utilization of these platforms should assess their particular needs and available resources in order determine the most suitable approach. Individuals or organizations have the option to implement open-source SOC platforms on their own infrastructure. Users have the ability to independently download and install the necessary tools, adapt them to suit their specific requirements, and establish their own SOC environment to handle cyber threats.

Open-source tools are generally accessible without fee, leading to significant cost reductions in the development and maintenance of a SOC platform. Organizations can avoid paying high costs for licensing fees typically associated with proprietary software by utilizing open-source solutions. Open-source SOC platforms provide the opportunity to customize the system according to the specific needs of an organization. Security teams have the ability to customize and enhance the features of the tools to meet their specific requirements. The ability to customize at this level enhances the efficiency and effectiveness of the security monitoring and response system. Even though open-source SOC systems have a lot of benefits, it's important to remember that implementing them up and functioning and managing them may require specialized skills and resources. Hence finally, the prototype that was implemented successfully achieved accuracy results, as stated in Chapter 4.

ACKNOWLEDGMENT

This paper is based on work I completed as part of my MSc dissertation at York St. John University. Therefore, I wish to express my sincere gratitude to Dr. Gayathri Karthick for her guidance, encouragement, and constructive criticism throughout this project. Her input was truly valuable and contributed significantly to producing a stronger final outcome. I would also like to extend my thanks to all the staff at York St John University, including Dr. Nalinda Somasiri, for their support during my academic journey.

I am grateful to my friend, Jayanga Palihena for his guidance and assistance during the publication process. Furthermore, I would like to give special thanks to my colleagues, friends, and family, whose constant support and encouragement helped me throughout this project.

REFERENCES

[1] M. Beller, “Modern Code Reviews in Open-Source Projects: Which Problems Do They Fix?,” Hyderabad, India, 2014.

[2] H. M. S. B. Naseer, “Demystifying analytical information processing capability: The case of cybersecurity incident response.,” *Decision Support Systems,* 2021.

[3] Johnson Kinyua and Lawrence Awuah, “AI/ML in Security Orchestration, Automation and Response: Future Research,” *Intelligent Automation & Soft Computing,* 2021.

[4] S. M. Milajerdi, “Poirot: Aligning Attack Behavior with Kernel Audit Records for Cyber Threat Hunting,” London, United Kingdom, 2019.

[5] V. M. a. A. Jøsang, “Data-Driven Threat Hunting Using Sysmon,” Guiyang, China, 2018.

[6] elastic.co, “elastic,” 2023. [Online]. Available: https://www.elastic.co/beats/. [Accessed 2023].

[7] M. ATT&CK, “MITRE ATT&CK,” 2023. [Online]. Available: https://attack.mitre.org/. [Accessed 2023].

[8] wazuh.com, “wazuh,” 2013. [Online]. Available: https://documentation.wazuh.com/current/user-manual/ruleset/rules-classification.html.

[9] elastic.co, “elastic,” 2023. [Online]. Available: https://www.elastic.co/kibana/. [Accessed 2023].

[10] R. Bidou, “Security Operation Center Concepts & Implementation,” *ResearchGate,* 2014 March 6.

[11] *. Johnson Kinyua1 and Lawrence Awuah2, “AI/ML in Security Orchestration, Automation and Response: Future Research Direction,” *Intelligent Automation & Soft Computing,* vol. vol 28, 2021.

[12] C. STAMFORD, “Security Operations Centers and Their Role in Cybersecurity,” Dubai, 2017.

[13] V. P. Ron Ross, “Enhanced Security Requirements for Protecting Controlled Unclassified Information,” NIST Special Publication, US, 2021.

[14] E. Ahlm, “How to Build and Operate a Modern Security Operations Center,” Gartner , 2021.

[15] Scott Crawford, “Security Orchestration and Automation: The Future of SecOps,” 451 Research, 2018.

[16] E. S. Group, “Automation and Orchestration in Security: SOARing to Success,” ESG by TechTarget, 2019.

[17] C. S. H. Editer, “A Review of SOAR Platforms: What are They, Why Do You Need One, and How to Choose?,” Cyber Security Hub, 2020.

[18] S. Institute, “Security Operations Centers and Automated Incident Response,” SANS Institute , 2021.

[19] b. A. Group, “Security Orchestration and Automation: Enhancing Security Operations,” Aberdeen Research Group, 2019.

[20] M. Gardiner, “https://www.rapid7.com/,” July 2022. [Online]. Available: https://www.rapid7.com/blog/post/2022/07/27/5-soar-myths-debunked/. [Accessed 05 2023].

[21] A. W. M. a. R. K. Ramachandran, “Implementation of Security Orchestration, Automation and Response (SOAR) in Smart Grid-Based SCADA Systems .,” India, 2020.

[22] H. W. T. &. R. C. Löhr, “A Survey on Security Orchestration, Automation, and Response Systems,” ACM Computing Surveys, 54(1), 1-37., 2021.

[23] Rizk, R., & Fakhereddine, M., “Security Operations Center (SOC) and Security Orchestration Automation and Response (SOAR) Using Open Source Tools,” 2021.

[24] NIST, “Enhanced Security Requirements for Protecting Controlled Unclassified Information,” NIST Special Publication, 2020.

[25] T. Lertwuthikarn, V. C. Barroso and K. Akkarajitsakul, “Resource Optimization for Log Shipper and Preprocessing Pipeline in a Large-Scale Logging System,” Hualien, Taiwan,, 2022.

[26] O. B. R. H. a. M. W. G. Oleksii Kononenko, “Mining Modern Repositories with Elasticsearch,” University of Waterloo, Waterloo, ON, Canada, 2014.

[27] B. S. M. G. B. J.-C. T. James Hamilton, “SCADA STATISTICS MONITORING USING THE Elastic Stack,” CERN, Geneva, Switzerland, 2017.

[28] S. V. Bezzateev, S. G. Fomicheva and G. A. Zhemelev, “Agent-based ZeroLogon Vulnerability Detection,” St. Petersburg, Russia, 07 July 2021.

[29] Virustotal, “Virustotal,” 2023. [Online]. Available: https://www.virustotal.com/gui/home/upload. [Accessed 2023].

[30] telegram.org, “telegram,” 2013. [Online]. Available: https://telegram.org/. [Accessed 07 2023].

[31] P. A. Networks, “Palo Alto Networks,” 2023. [Online]. Available: https://www.paloaltonetworks.com/cyberpedia/soc-roles-and-responsibilities.

[32] C. Ltd, “Ubuntu,” 2023. [Online]. Available: https://ubuntu.com/. [Accessed 01 2023].

[33] vicarius.io, “vicarius,” 2020. [Online]. Available: https://www.vicarius.io/blog/cve-and-cvss-whats-the-difference.

[34] A. Greenberg, “ https://www.wired.com/,” 2014. [Online]. Available: https://www.wired.com/story/how-target-blew-it/. [Accessed 15 05 2023].

