National College of Ireland

BOLSTERING CLOUD SECURITY WITH REAL-TIME SIEM USING HYBRID-RULE BASED AND ML INSIGHTS.

MSc Research Project
MSc. In Cybersecurity

Arbaz Adib Dalwai
Student ID: x23161795

School of Computing
National College of Ireland

Supervisor: Dr. Rohit Verma

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Arbaz Adib Dalwai.

Student ID: x23161795

Programme: MSc. In Cybersecurity **Year:** 2024-25

Module: MSc Practicum Part-2

Supervisor: Dr. Rohit Verma

Submission Due Date: 29.01.2025

Project Title: BOLSTERING CLOUD SECURITY WITH REAL-TIME SIEM USING HYBRID-RULE BASED AND ML INSIGHTS.

Word Count: **10527** **Page Count**...22.....

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Arbaz Adib Dalwai.

Date: 28.01.2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

BOLSTERING CLOUD SECURITY WITH REAL-TIME SIEM USING HYBRID-RULE BASED AND ML INSIGHTS.

Arbaz Adib Dalwai
x23161795

Abstract

The dynamic cloud environments often rely on traditional rule-based detection systems alone to detect the sophisticated threats like distributed denial of service (DDoS) and phishing which sometimes fall inadequate in terms of adaptability required in modern day detection. The research tries to bridge the gap between the academic researches and practical applications by contributing a scalable and robust detection framework in modern cloud infrastructure. This research aims to design and implement a real-time hybrid detection mechanism on a cloud platform- Amazon Web Services (AWS) that would integrate the security information and event management (SIEM), intrusion detection system (IDS) and machine learning (ML) models to detect and classify the cyber threats efficiently. Attack simulations were conducted to generate real-time logs which were monitored through the Suricata IDS, followed by the log processing in elasticsearch, logstash and kibana (ELK) stack. Ensemble learning models like Random Forest (RF) and XGBoost were deployed to complement the rule-based detections and all this was presented in visual forms in real-time without significant delays. This was proven by an average detection time of 0.5 milliseconds, demonstrating the systems suitability for real-world conditions. The framework tends to bridge the gap between conceptual and practical deployments with the implementation of real-time hybrid detection system in a cloud environment.

1 Introduction

The businesses worldwide have seen a rapid growth in shifting towards a cloud-based infrastructure due to its scalable approach and unmatched convenience (Morkos, 2023). However, with this increasing reliance has also opened the doors for significant security challenges within the environment by the rise of cyber threats like DDoS and phishing (Waldman, 2024; Proofpoint, 2024). There a lot of traditional tools and technologies such as Firewalls and antivirus software put in place to identify these threats and ultimately mitigating the risks associated with it, but often these traditional based systems struggle to adapt to evolving attack patterns when working independently, leaving behind critical gaps in the detection mechanisms. Firewalls are typically used for blocking the unauthorised access but sometimes fall short in dealing with complex attacks that bypass the network defences, while antivirus software may detect malicious files but cannot detect a phishing activity. These limitations motivated the necessity of building a hybrid system which would combine the strengths of advanced ML models with techniques of systems like IDS and SIEM. By integrating the rule-based and ML-based methods into the cloud environment the system's overall detection capabilities can be increased by a large extent.

To encounter the challenges posed by the ever-evolving threats landscape in a dynamic cloud environment, this research investigates the following question:

"How can a cloud-based real-time security monitoring system using SIEM, IDS and machine learning models adaptively and effectively analyse and classify DDoS and Phishing attacks?"

The foremost objective of this research is designing and implementing a hybrid security solution that would integrate the rule-based detection with ML-models for enhancing the adaptability of the system in identifying the attack patterns accurately. The research will then assess the system's performance by testing it under simulations which will replicate the real-world environments to determine whether it is able to detect and classify the threats in real-time.

Figure 1 below illustrates the integral components of the proposed system-

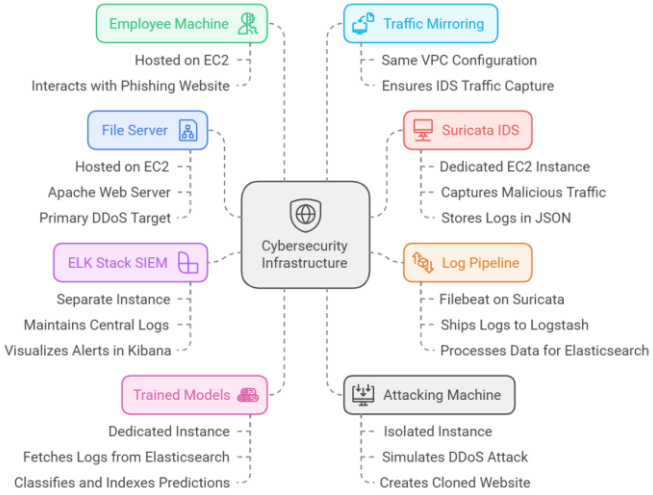

Figure 1: Cybersecurity Threat Detection Layout

This research aims to contribute in the field of cybersecurity through several key points such as demonstrating a practical integration of SIEM, IDS and ML models in order to create a hybrid detection framework which would be capable of identifying threats like phishing and DDoS in real-time. By providing visual alerts through kibana dashboards, the research also offers to provide real-time insights helping the analysts in better decision making. Despite the contributions, no research is without any limitation including this. The experimental setup relies on simulated attack traffic, which sometimes may not capture the complexities of the attack fully and this arises the need for future research in this area to test the framework in more complex settings.

The report is structured as follows: Section 2 depicts the related work detailing the analysis of existing researches in this area. Overview of the methodology is presented in section 3, containing the information on research framework and experimental plan. Section 4 presents the design specifications, while section 5 is the implementation, focusing on the technical aspects of the research. Evaluation consisting of the experimental findings and discussion is showcased in the section 6. Finally, section 7 concludes the research paper while proposing future enhancements.

2 Related Work

The world of cloud security has witnessed a lot of change in the past years with the integration of advanced tools such as SIEM and IDS for real-time threat detection (Ahmadi, 2024; Oracle, 2023). Also, a lot of advancements have been made to traditional cybersecurity

practices with the use of ML models into it. These technologies are playing a very vital role in the overall enhancements in the detection and monitoring techniques of different attacks. The following subsections will discuss about the literature review conducted for this research. Key search areas included the integration of SIEM and IDS for real-time monitoring, implementation of ML models for DDoS and phishing detections. This literature review helped in forming the base foundation for the methodology and implementation sections of this research.

2.1 SIEM and IDS in cloud security

The usage of SIEM and IDS solutions has been thoroughly investigated in many of the recent researches defining the importance of these tools in the field of cloud security. Research by (Tuyishime et al., 2023) has come up with a SIEM-based approach for threat monitoring in cloud environments through components such as virtual machines, load balancers, web application firewalls in order to have a central point for logging and analysis. It specifically tries to highlight the importance of SIEM in generating automated alerts to enhance the response times while also trying to evaluate the frameworks scalability in handling large volumes of data. However, it mainly focuses only on the system architecture and operational efficacy with very limited work on identifying specific attack types. (Granadillo, Zarzosa and Diaz, 2021) further extends this understanding by providing a comprehensive analysis of existing SIEM solutions including their functionalities, benefits and limitations. It mainly focuses on the necessity of integrating SIEM in modern cybersecurity infrastructures while also pointing out the challenges the solution has to offer. Likewise, the research by (Lee et al., 2017) extends these ideas with a SIEM architecture specially designed for Security-as-a-Service (SECaaS) in the cloud. This research describes the key factors which include scalability, flexibility and multi-tenancy when adapting SIEM systems to cloud native applications. This study however does not have an evaluation setup to test the system in real-world scenarios.

In order to make further improvements in the SIEM technologies, (Ayu et al., 2024) have presented the concept integrating ML with SIEM to improve the detection capabilities and thus showing the potential of ML models in solving the gaps in traditional rule-based systems. Here the researchers have implemented anomaly detection using the RF classifier on CSE-CID-IDS2018 dataset collected from an IDS which is pre-processed using the principal component analysis. Despite its promising results the study lacks on evaluating the model's performance on real-time data and only focuses on one particular dataset. Similarly (Saeed et al., 2022) presents a ML-based IDS which is customised for usage in the cloud environments through the incorporation of a number of algorithms such as Naive Bayes classifier, decision tree classifier, supporting classifier, logistic regression and RF classifier to analyse network traffic and detect anomalies. The study stresses on how the implementation of ML techniques can lower down the false positive and enhance the detection capabilities of an IDS in cloud infrastructure. While, (Innab et al., 2024) present the different security attacks in cloud environment while evaluating the different IDS types and techniques which can tackle it categorizing the IDS into signature-based, anomaly-based, hybrid based and other approaches and analyses their effectiveness in handling and detecting the security events. The research states that the signature-based attacks can effectively detect known threats but fail in unknown threats, anomaly-based are well-known for detecting unknown threats but have a greater false positive alert, while hybrid approach balances the two and provides improved detection accuracy. While this research was comprehensive it lacked any practical implementation to showcase the findings.

2.2 Phishing detection mechanisms

Various researches on phishing detection have explored several uses of ML techniques in order to identify and mitigate these attacks by analysing different data features. (Dutta, 2021) proposed a ML-approach by using recurrent neural network (RNN) for detecting phishing attempts by analysing the URL features. This research focuses on describing how the ML model identifies the phishing URLs through sequential patterns in URL structures. But the study focuses only URL-based approach without considering other factors such as webpage content or DNS information. Also, the researcher here doesn't test the model's performance in real-time scenario. Thus, opening the future opportunities of testing the ML capabilities in detection of phishing attacks in real-time. Similarly, (Salahdine, Mrabet and Kaabouch, 2022) introduced a ML-based approach for analysing and detecting the phishing attacks by working on the email features. This experimentation was done on a sample of 4,000 phishing emails by selecting 10 relevant features. They trained, validated and tested the dataset on 3 models-support vector machine (SVM), logistic regression and artificial neural network (ANN). However, this research only focuses on emails and not the network traffic but has emphasised greatly on feature selection to enhance the detection. Such detection systems if combined with the predefined rule-based techniques like SIEM or IDS can strengthen the threat monitoring endpoint infrastructures by margins.

2.3 DDoS detection mechanisms

Extensive researches conducted on DDoS detection have shown various innovative approaches that use SIEM and ML techniques to address the growing complexities of this attack. (Çakmakçı et al., 2021) proposes a framework for detecting various types of DDoS attacks through SIEM. It encompasses an incident detection engine within the SIEM in order to identify the attack patterns. It also has an inference engine which automatically suggests the response and recovery steps after detection thus reducing the reacting time for ongoing attacks. Future research on this framework could enhance the detection capabilities of the system when tested in diverse network environments (Çakmakçı et al., 2021). Complementing this, research by (Dhahir, 2024) introduces hybrid technique which combines the clustering-based local outlier factor for feature engineering with extreme gradient boosting (XGBoost) for the detection of DDoS attacks. This model was implemented and tested on the CIC-IDS 2017 and CIC-IDS 2018 achieving an accuracy of 99.99% and a precision score of 100%. However, its effectiveness has not been tested in dynamic settings. Similarly, (Chen et al., 2018) uses the XGBoost algorithm for detecting the DDoS attack in a Software defined networking-based cloud environment. The proposed model had a high level of performance in detecting normal and malicious traffic hence proving the effectiveness of using the XGBoost. It also depicted that the XGBoost can effectively handle large volumes of data in dynamic cloud environments. However, exploring the impact of real-world noisy data with different traffic and attack types would actually test the model's performance which is lacking in this study. Collectively these researches highlight the efficiency of SIEM and advanced ML models especially XGBoost in improving the DDoS detections.

2.4 Summary

This section explored various existing researches conducted on SIEM, IDS, advanced ML techniques for detecting different attacks including phishing and DDoS in various environment especially cloud environments. The following TABLE-1 shows a concise summary of other findings while proposing the solutions which this research has to offer-

Table 1

Aspects	Findings	Gaps Identified	Research Contribution
SIEM and IDS	SIEM is considered as an essential component for event management while IDS improves the detection rate.	Insufficient integration of ML models with SIEM and IDS to create new hybrid systems.	Combined the ML models with SIEM and IDS in order to enhance the overall detection rates.
Phishing detection	Results obtained from ML models (RF and XGBoost) have a higher accuracy in identifying phishing and DDoS attacks.	Models not tested with real-world dynamic data thus questioning the scalability.	Framework designed for real-time assessment of ML models for phishing and DDoS detection in a cloud environment.
Recurring gaps	Lack of real-time experiments and inadequate combination of rule-based prediction and ML-based prediction approaches.	Inadequate evaluation of results in hybrid systems under real-time conditions.	Proposes a system that integrates rule-based and the ML-based predictions in a real-time cloud setup evaluating its effectiveness and scalability.

3 Research Methodology

The research is designed with an experimental and quantitative approach methodology, wherein the controlled simulation of attacks, its monitoring and analysis using SIEM and IDS reflects the experimental aspect, while the numerical nature of the collected logs and their analysis using ML models and evaluation metrics resembles the quantitative side. The system designed uses both rule-based and ML-based detection techniques in order to identify and analyse DDoS and phishing attacks. The methodology can be structured and divided into the following standard research categories-

3.1 Research Design

An experimental approach consisting of three main components: simulated attack, monitoring network activity and detection mechanism was designed to answer the stated research question effectively.

The first component includes the simulation of two different types of attack that is DDoS and phishing attack. These attacks were launched through an attacking machine setup on the AWS elastic compute cloud (EC2) instance and the simulations were made sure were designed in a way which reflects a real-world threat scenario. The DDoS attack was specifically targeted on the file server hosted on the EC2 instance using the hping3 command from a docker service on the attacking machine which generated high volume of traffic. While the phishing attack was conducted by cloning a website http://example.com/ using the social engineering toolkit. This website was then triggered and accessed by a click on the employee machine which imitates that a user has potentially accessed a malicious link. These

attack simulation steps were crafted and implemented in order to generate and capture enormous network activity logs on the IDS.

As mentioned previously the second component was monitoring all the activities in the cloud environment on a real-time basis. For this Suricata which is an open-source (IDS) was deployed on the EC2 instance in the same virtual private cloud (VPC) as that of the file server and employee machine and through traffic mirroring on AWS it was made sure that the IDS is able to capture all the logs depicting the network traffic in the design. The IDS was encompassed with all the Emerging threat rules which made sure that the IDS is up to date and would identify different anomalous behaviour which is possibly associated with the launched attacks. Moreover, importing the emerging threat rules also made sure that the system also detects other attacks through a detailed inspection of logs. Filebeat was also installed and configured in the same instance as suricata in order to send the logs to ELK stack. This made sure that the logs received by the SIEM are transferred in real-time without any data loss for the further processing and analysis.

The final and the most crucial component was detection mechanism. The evaluation of the detection mechanisms can be summarized in two parts: rule-based detections and ML-based detections. The rules were encompassed in the Kibana to identify the specific patterns in the logs to identify DDoS and phishing activities. Additionally, ML models (RF and XGBoost) were trained on the IDS logs to classify the attacks. This dual approach in the detection mechanism ensured a comprehensive evaluation of the logs in real-time making sure of a secure network infrastructure deployed completely on the cloud. By combining these components, the research design ensured a robust framework for simulating, analysing and monitoring security threats in a cloud-based environment. This design also made sure that the testing is adhered to standard practices such as the National Institute of Standards and Technology (NIST) guidelines by following a proper data collection and analysation from relevant sources.

3.2 Data Collection

The simulated real-world attacks generated datasets in the form of logs (network traffic activity) which was the primary source for the data collection in the entire setup. The main objective here was to collect network activity logs that would resemble the DDoS and phishing attack which would then further be used for rule-based and ML based detections. To simulate the DDoS attacks hping3 command was used from a docker container on the attacking machine hosted on the EC2 instance. This command was targeted on the file server ensuring that a high-volume traffic is logged on the file server imitating a typical DDoS like scenario with traffic spikes and abnormal connection requests/attempts. The hping3 command was specifically used due to its flexible approach in sending network packets helping effective log generation.

For the phishing attack simulation, the social engineering toolkit (SET) was used to clone a website and then this website was accessed through a script on employee machine which replicated an interactive simulation between a real user engaging with a malicious link. Through this the logs having the traffic flows were generated indicating phishing attempts. All these testing was done in a controlled an isolated environment adhering to the standard ethical testing practices. All these logs were as stated earlier captured on an open-source IDS-Suricata. It was configured in a way to detect all the network traffic in the infrastructure and record the network metadata including the source and destination Ips, ports, protocols, timestamps and other relevant information associated in a genuine network traffic. These logs were structured and stored in the JavaScript object notation (JSON) format on the Suricata which made it convenient for further data processing by the upcoming tools and technologies.

Specific Characteristics of the Logs:
The logs include critical metadata which is explained in the TABLE-2.

Table 2

Fields Captured	Reason
Source IP (src_ip)	Origin of network traffic
Destination IP (dest_ip)	Target system or service
Source Port (src_port)	Port linked to the originating application
Destination Port (dest_port)	Service which is being targeted such as HTTP (port 80)
Protocol	The network protocol which is used (e.g., TCP, UDP).
Timestamp	Precise time of the recorded activity
Event Type (event_type)	Classifies the nature of network activity whether it is an alert, flow or normal

Characteristics:

- High traffic volume spikes: Reflecting DDoS attack patterns.
- Repeated connection attempts: Possible characteristic of SYN flood attacks.
- Suspicious HTTP requests: Access attempt through automated tools can be indicated through User-agent strings.

The logs had special features like high consistency as the suricata uses ET rulesets which makes sure that the events are correctly tagged as well as depicted realistic simulation since the logs contained both benign and malicious traffic. Data quality was a key aspect to be taken care of during the collection phase as the entire detection process was relying on the data in IDS and hence filebeat was introduced so that it transfers the data ahead without dropping the essential metadata fields ensuring seamless log transmission.

3.3 Data Preprocessing

The data preprocessing step mainly focused on the preparation of the network activity logs collected by the IDS for ML-based threat detection. The data preparation involved cleaning and feature engineering from the raw JSON logs stored in the Elasticsearch to make sure that the data on which the models are to be trained is accurate. The process started with the extraction of important fields from the raw JSON logs consisting of nested metadata fields source and destination IPs, ports and event types. Through the use of python programming language this JSON structure was parsed to extract the fields like src_ip, dest_ip, src_port, dest_port, event_type, user_agent and hostname. Extracting these fields was critically important as they possess necessary information such as origin of network traffic, target of the traffic, originating service, service which is being targeted, nature of the activity, software or tool which was used to access a resource and the name of the host involved in the activity. This data combined can be used for an effective threat detection. This was then organized into a tabular format by using pandas. Through this process it was made sure that the data is suitable for training and further analysis. Further the numerical encoding was applied to convert the categorical features such as IP addresses into numerical representations. This was done using a custom function which made sure the data is consistent and compatible with the ML models. Features such as src_port and dest_port were retained as it is, due to its integer structure. The missing values were addressed either by assigning them with default values or excluding out the incomplete records.

The data was then labelled to its corresponding simulated attack scenario. The data was mainly tagged in three categories: DDoS, phishing and other. For DDoS, the logs were tagged by detecting high amount of traffic flows targeting the same destination IP and port. For phishing logs, the conditions included HTTP requests to specific ports (80 and 443) with specific user-agent strings which can possibly indicate the access of cloned website through automated tools. The logs which did not fall under the above criterions were tagged as other. Once the data was completely ready for analysis it was split into training and testing making sure there was no imbalance. Standardization was also done on the features in order to put the data into correct format for enhancing the model's performance. Thus, this cleaned and pre-processed data was then used to train the RF and XGBoost models ensuring effective evaluation.

3.4 Experimental Setup

The setup was designed and executed completely on AWS making sure that the environment is completely controlled and isolated as it involved simulation of attacks, real-time log collection followed by its evaluation. The main components of the setup are explained below-

- File server- It is hosted on an EC2 instance using an Apache web server. The main objective of this was to act as a central repository for any organization which would be accessed by its employees and hence was connected in the same way. As a crucial asset to the company if the service is unavailable for its legitimate users, it could severely impact on the organization. Hence this server was selected as the primary target for the DDoS attack.
- Employee machine- This is hosted on another EC2 instance which would imitate the behaviour of an end-user for the company. This was primarily developed to interact with the phishing website so that the associated logs would be generated.
- Suricata IDS- It is deployed on a dedicated EC2 instance in order to monitor the real-time traffic associated within the network. This not just captures and analyses the malicious and normal traffic but also generates the logs and stores them in JSON format for further processing.
- Traffic Mirroring- While the file server, employee machine and IDS all instances were deployed within the same VPC additional configurations were required to setup a clear connection between these in order to ensure that the IDS captures the network traffic from both the sources and thus traffic mirroring was configured to establish and maintain this connection.
- Log Pipeline- For a smooth and seamless transfer of data, filebeat was deployed and configured on the Suricata instance. It is used to ship the logs from IDS to Logstash instance directly in real-time. The Logstash then processes and structures the data before sending and storing it in the Elasticsearch. This flow makes sure an efficient way to handle the data transmission.
- ELK stack SIEM- This is again deployed on a separate instance which is responsible for functioning of Elasticsearch, Logstash and Kibana. The main functionality of the SIEM here is to maintain the logs centrally which are received from the IDS, visualising the alerts triggered through the rule's setup in the Kibana and display it on the dashboards. It not just inputs the data from IDS but also from the ML model instance further displaying its predictions.

- Trained models (EC2 instance)- A separate and dedicated instance was deployed in order to store the trained models wherein it can fetch the logs from Elasticsearch, classify them according to the trained models and index the predictions back to the Elasticsearch.
- Attacking Machine- While the above setup displays the detection capabilities, this instance was deployed in order to generate the network traffic through the simulation attacks. This is deployed on an isolated instance wherein a docker container is installed and used to simulate the DDoS attack using the hping3 command. While the deployment of SET is used for creating a cloned website (http://example.com/) which helps in simulating the phishing attack. This cloned website was then linked with a dynamic Domain Name Service (DNS) through No-IP which would make it accessible for public usage increasing the authenticity of the simulated attack.

3.5 Detection Mechanism

The detection mechanism used in this research uses a dual strategy wherein the detection relies on both rule-based detection using the SIEM capabilities as well as on the predictions made through the ML models. This combines approach complements each other and strongly puts together a robust framework for identifying security threats in real-time.

Rule based Detection- The ELK stack SIEM has a vast dictionary of pre-configured rules to detect wide range of known threats and attacks. These rules were then imported in our system from SIEM's default library. This helped in identifying the attack patterns such as high-volume traffic, unusual port usage, suspicious HTTP requests, etc. In order to enhance the system's detection capabilities custom rules were also created in Kibana to identify and flag the attack patterns observed in the logs through the simulated scenarios. Mainly two rules were created focusing on DDoS and Phishing wherein the former was customised to trigger the alerts after identifying high traffic anomalies, such as SYN flood attack targeting the file server while the latter was customised in identifying HTTP requests to a website which was hosted on a dynamic DNS service. These rules helped in generating real-time alerts on Kibana which was then helpful in visualizing the dashboards that enable efficient real-time monitoring of network activities.

ML-based Detection- The ML-models- RF and XGBoost were deployed and trained based on the preprocessing steps mentioned in the earlier section. They were integrated in order to adapt the threat detection by analysing the network traffic logs and classifying the events as phishing, DDoS and benign. Here a python-based pipeline was designed and structured in way that it fetches the logs from Elasticsearch and passes them to the model for classification and these predictions made by the models were then indexed back to Elasticsearch ensuring an absolute integration with the SIEM system. This integration of ML based detection thus strengthened the system's detection capabilities as it detects the attack patterns which get evaded from the static rules enhancing the overall detection capabilities.

This combined approach can be viewed together at the same time on Kibana dashboard making it effective to address the challenges posed by the attackers through the attacks such as DDoS and phishing in real-time.

4 Design Specification

4.1 Introduction to System Design

The main objective of this research was to structure a robust cloud-based real-time monitoring structure which would be able to capture, detect and classify the DDoS and

phishing attacks. The architecture of the entire setup was precisely crafted to imitate a realistic corporate environment including multiple components that would seamlessly interact with each other in order to monitor and analyse the network traffic. The primary components integrated here are the file server acting as sensitive organisational resource, an IDS which would monitor the network traffic, a Security Information and Event Management (SIEM) system which would centrally log and visualise the traffic and finally the ML models which would enhance the threat detection capabilities of the system through its adaptiveness.

The design was built by prioritizing the scalability and adaptability of the solution, which would allow the system to handle real-time traffic flows and support multiple detection strategy. The setup was implemented completely on AWS due to its enhanced flexibility and great computational power resources which was necessary for simulating the real-world attack scenarios and analysing the network traffic activity in real-time without any delays.

4.2 Architectural Overview

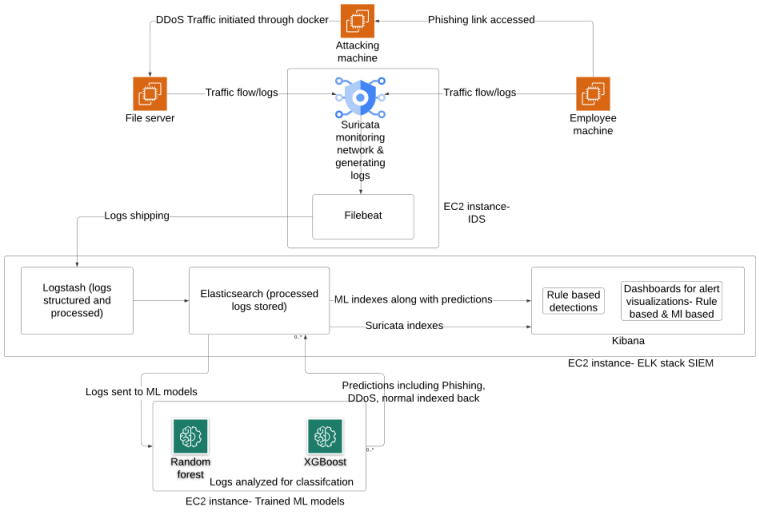

Figure 2: System Architecture

The Figure 2 above shows the architecture of designed system consisting mainly of six AWS EC2 instances which are responsible for handling the working of file server, employee machine, Suricata IDS, ELK stack SIEM, trained ML models (RF and XGBoost) and the attacking machine. All these instances had to be interconnected for a seamless interaction between them which would enable real-time threat detection. Executing the solution on cloud also enabled the system to be completely isolated and controlled for the experimental purpose. The file server which is hosted on the Apache web server acts as the primary target for the DDoS attack, as it represents to be an integral asset for any company, while the employee machine is the end user device which would be interacting with the phishing/malicious link. Traffic mirroring was done on AWS by setting up two mirror sessions mainly between the file server and IDS, employee machine and IDS. This approach ensured that all the network traffic would be routed towards the IDS enabling complete monitoring of the resources. The Suricata IDS was then responsible in analysing this captured traffic and generating detailed logs for all the network activity. These logs were then passed to the SIEM server with the help of filebeat, where the logs were analysed, indexed and presented in real-time dashboards. The ML models were deployed and run on an EC2 instance which would link them to the SIEM instance for detection. This connection helped in SIEM sending the logs to the instance where the models could classify the events as phishing, DDoS and normal (benign/other) and send the predictions back to the SIEM for its

visual representation along with the rule-based detections. The attacking machine instance was used to simulate the DDoS attack via a docker container which would generate the SYN flood and UDP flood attack using the hping3 command. On the other hand, the phishing attack simulation was isolated from the docker by directly hosting a cloned website http://example.com/ using the SET. This website was linked to a Dynamic DNS service through No-IP which would simulate real-world phishing scenario. By using the dynamic DNS, the setup tried to enhance the authenticity of the attack as many attackers around the globe are using such services/resources to avoid detection and block listing.

4.3 Justification for selected Tools and Technologies

All the tools and technologies selected were chosen to align with the current needs of the cybersecurity world making sure that the research's setup would imitate the practical use cases on one hand while maintaining the academic relevance as well.

- Suricata (IDS)- This IDS being an open-source and highly customizable system emerges to have prominent ability to perform a real-time network traffic analysis. It generates detailed logs of the captured network activity which is a primary need of the research. It also supports Emerging Threat rule sets allowing real-time detection of not just DDoS and phishing attack patterns but hundreds of other attacks. Due to all these benefits especially its ability to handle the high-volume traffic efficiently because of its native multi-threading techniques made it a better choice than any other open-source IDS (viz. Snort IDS) (Waleed, Jamali and Masood, 2022).
- ELK stack SIEM- This again being an open-source tool with Elasticsearch which can efficiently store and index large volumes of data/logs for real-time analysis, Logstash which can process and structure the logs ensuring that the data is suitable for advanced querying in the Elasticsearch and lastly Kibana which provides a user-friendly GUI for interpreting what is actually going on in the SIEM by visually representing data via dashboards. Due to all the necessary requirements being fulfilled by using stack and given its popularity in real-world wherein it is used by multinational giants this tool was highly relevant for the overall system.
- Apache web server- Being lightweight and used across the globe for hosting web applications, it became an ideal choice for simulating a corporate file server.
- Dynamic DNS (No-IP)- It was primarily selected to link the cloned phishing website which adds realism going parallel with an attacker mindset often using dynamic DNS to evade detection while phishing attempts.
- AWS EC2 instances- As the above-mentioned tools and technologies require heavy computational power and storage to process and function to the best of their requirements it was very important to provide them the base, they need for the setup. This was offered by EC2 instances with its immense scalability & flexibility for deploying multiple components in one go. It also helped in assigning the elastic IPs to our systems.
- ML models (RF and XGBoost)- Both being ensemble learning models have improved classification rate and can reduce overfitting issues notably. While RF is a robust ML model usually used to handle large datasets, XGBoost provides optimum performance with imbalanced datasets making both an ideal selection for adaptive detection in real-time environment considering their speed and effectiveness.
- Social Engineering Toolkit- It is an open-source tool which is powered by python mainly used in penetration testing activities. This was chosen to clone the website and simulate a phishing attack.

- hping3- This is a highly flexible command line tool which can generate custom packets with varying attack intensities to simulate flood attacks (Ariffin, et al. 2021).

4.4 Design Considerations and Constraints

The balance between realism, scalability and adaptability was the core while designing the entire system. Scalability was highly prioritised as the system had to deal with large volume of traffic which would be generated during the simulated DDoS attack, this traffic was also to be processed giving the results in real-time. Thus, the use of AWS EC2 instances became very crucial as it provided the flexibility to deploy components with different computational needs such as lightweight instances for file server and other machines and high performance and high storage instances for IDS and SIEM. As this setup was entirely run in an AWS environment, it ensured that the simulated attacks are not creating any issues for the external system. Docker container was deployed on the attacking machine to set apart the traffic sources of two distinguished attacks. This enabled control over traffic generation without any issues such as cross-contamination of logs. The next big thing to be constituted was realism as the entire setup had to be replicated as a real-world attack scenario. The usage of cloned website linked to a dynamic DNS resembles a modern-day attacker's mindset to dodge the detections during a phishing attack. Also, as the system encompasses the rule sets from emerging threat point on Suricata the detection capabilities of phishing and DDoS attacks were enhanced and well equipped with industry standards.

While all this was to be done several challenges aroused during the implementation which were to be addressed and overcome for making a resilient system. Initially, AWS lambda and AWS Sagemaker were implemented for the deployment of ML models. But due to configuration and connectivity issues and also problems raised by IAM constraints, this approach was not feasible and was replaced by the adoption of EC2 instance wherein these models are deployed. Moreover, with ML models another big issue faced in early stage was that the models were trained on sample datasets initially but the raw logs which were generated from the IDS were nothing similar to the sample datasets available in various resources, this created a massive feature mismatch causing the retraining of models from the datasets acquired from actual raw JSON logs from the experimental setup. Through this the alignment between the dataset and real-time predictions the results received were unparalleled. Connectivity issues between filebeat, Logstash and Elasticsearch was addressed through repeated trial and testing to ensure that the log transmission pipeline is functioning accurately. The resource limitations of EC2 instances in handling the heavy tools with huge volume of data was sorted out by promoting/upgrading the instance types and finding the accurate one which could balance the load for the system. These considerations and constraints overall framed the architecture of the entire system resulting in a robust and reliable design.

5 Implementation

5.1 Infrastructure Setup

The TABLE-3 shows the EC2 instances with their configured roles and resource allocations

Table 3

Component	Instance name	Elastic IP	Instance type	Role	Configuration details
File Server	File Server	98.83.82.139	t3.micro	Hosting Apache web server	Apache server with a login form. Traffic mirrored to IDS
Employee machine	Employee machine	52.204.25.98	t3.micro	For end user interaction	Script to auto click the phishing link Traffic mirrored to IDS
IDS server	IDS server	54.82.178.224	t2.micro	Monitoring the network traffic	Configured Suricata with the Emerging threat point rules. Deployed filebeat and connected with SIEM
SIEM server	SIEM-server	3.228.185.88	t3a.large	Hosting Elasticsearch, Logstash and Kibana	Logstash receiving data. Elasticsearch (port:9200), Kibana (port:5601)
Trained models server	ML models	54.81.145.37	t3.micro	Running trained models	Connected with SIEM instance to fetch logs and push predictions
Attacking machine	Attacking machine	Dynamic IP	t2.micro	Simulating the DDoS and phishing attack	DDoS via hping3 through docker. SET and No-IP dyn DNS for phishing

5.2 Detailed Configurations

- Suricata Configuration- Suricata as an IDS was deployed on a dedicated t2.micro instance running on ubuntu with a storage space of 17 GiB. All the primary configuration was modified in the customizable suricata.yml file. This instance was also set as the mirror target to have a clear mirror session between the file server and the employee machine on the IDS. The IPs of these two servers were also added in the Home Net section of Suricata configuration file so that it is able to capture the network traffic smoothly. The rule sets were then imported from Emerging Threat point in the IDS through suricata-update command to ensure that the detection capabilities of the IDS are well in place. Mainly, the *ET Info Dynamic_DNS query to .dyndns domain* was also included through this which potentially flags the phishing activity as it identifies the DNS queries are commonly associated with phishing websites being hosted on dynamic DNS services.

Other critical rules included several other attack patterns especially the one identifying DDoS attacks in case of SYN flood and UDP flood scenario. All these rules were imported and activated on the system.

- Filebeat configuration and integration- Filebeat was deployed and configured on the IDS server due to its lightweight approach wherein it forwards the eve.json logs to the Logstash in the SIEM server. This was done primarily by enabling the suricata module on filebeat through (sudo filebeat modules enable suricata). Through this module the suricata logs were automatically parsed and structured eliminating any need for custom input configurations in the filebeat.yml file. However, the output configuration had to be set in the filebeat.yml wherein it can target the logstash server on its IP- 3.228.185.88 which listens the data on port 5044. This allowed the filebeat to forward the structuted logs.
- Logstash Configuration and integration- Logstash plays an important role in the entire SIEM server as it listens to the raw logs sent from the filebeat and transmits it to the elasticsearch. All this communication is defined in the filebeat-suricata.conf file located in logstash. The configuration is mainly divided in three parts input, filter and output for handling the suricata logs effectively. The input section depicts the listening of Json logs on port 5044 aligining with the output section in filebeat conf. Next, in the filter section, it identifies and processes the events of alert type. The output section of the file is defined to send the logs to elasticsearch which is hosted on the same SIEM server on port 9200. These logs are indexed in a separate index pattern naming, suricata-logs-*. By specifying the formatting as index => "suricata-logs-%{+YYYY.MM.dd} the file also ensures that a separate index file is created on each day with different logs. This configuration is validated through immense testing of pipeline using testing commands and thorough inspection of whether the real-time logs are indexed in elasticsearch.
- Elasticsearch and Kibana configuration- Elasticsearch is deployed as a single node-cluster. It is configured to listen all the network interfaces (0.0.0.0) on port 9200 so that the connections beyond the localhost can also be accepted. This was mainly essential for forming a seamless connection with other SIEM components that is logstash and kibana. The xpack.security module was added to support the authentication and secure access through Application programming interface (API) keys. On the other hand Kibana has also been configured to bind all the network interfaces as the server.host is set to 0.0.0.0 on port 5601. This was done for accessibility from any system, within the network and interaction with elasticsearch. Although elasticsearch and kibana are configured to an initial localhost for development usage, these parameters were changed to fit the distributed cloud structure. Kibana accesses the elasticsearch through its own credentials which are set in its conf file (kibana_system user and its passwords). All the default elastic rules a total of 686 are updated in the kibana along with the custom rules. Rules for detecting DDoS and phishing were defined in kibana, which uses suricata-logs-* index and fields such as event_type, alert_signature and hostname. The phishing alerts are triggered with the rule identifying DYNAMIC_DNS HTTP Request and its associated patterns, while DDoS detections are relying on SYN flood signatures tagged as Attempted Denial of Service attack patterns. There were specific connectors put in place for the integration of this rules.
- ML integration- Here the two .pkl files for the selected ensemble learning models which are pre-processed as per the instructions stated in earlier section are hosted in a t3.micro server. These models are deployed to analyse the logs in real-time from suricata-logs-* index located at elasticsearch. A connection to elasticsearch is established here using the REST API, using the SIEM server IP which is 3.228.185.88. Also it has been given the authentication credentials for a secure data connection. As the system is designed to work continuously with real-time data the @timestamp field is used to fetch only the logs that

are new or updated since the last processed batch. This dynamic strategy helped the models to deal with fresh data every time. Next up is the preprocessing pipeline which is applied to extract the features such as src_ip, dest_ip, src_port, dest_port and event_type from the fetched logs to match the feature requirements of the model for effective classification. Logs with missing values are filtered out during this stage, to make sure that model works on complete data. Both these models generate probability scores for each log entry for marking the classification of attacks. These scores are then averaged so that the strengths of these models are combined and a classification threshold is put in place marking the entries with scores above 0.7 and 0.3 are tagged as phishing and DDoS respectively, while those in between are tagged as benign activities. These thresholds were chosen due to the nature of attack. As phishing attack has high-confidence indicators such as Dynamic DNS usage (*.dyndns) it leads to a higher score in the model prediction while in DDoS attack, it heavily relies on huge volume of traffic with repetitive patterns such as SYN flood attempt resulting in a generally lower score from the model thus logs with less than 0.3 score are tagged as DDoS. To minimize false positive rate the logs which lack the distinguished patterns for the attacks are classified as benign. The processed logs which now contain the prediction classifications are indexed back to elasticsearch in a new index pattern as ml-predictions-YYYY.MM.DD* using bulk API operations. The date-based naming system was introduced for a better organisation of logs & predictions. The system continuously processes the available logs in a continuous loop every 5 seconds and if the system encounters any connectivity issues it gracefully handles it through retry mechanism. Thus, this approach offers optimized data ingestion and scalability without affecting real-time predictions efficiency.

- Attack simulation setup- The phishing attack was done using a webpage hosted by a cloned website through SET's credential harvester attack method and site cloner functionalities. Once the inputs of POST-back IP (attackingmachine.ddns.net) and target website URL (http://example.com/) were provided it saved the cloned content in /var/www/social-engineer-toolkit/ directory. However, this cloned website can only be accessed when the hosting server (SET) is active, hence once the SET is terminated the http server used by it also stops. Thus, Apache was used to copy the cloned content from SET by sudo cp -r /var/www/social-engineer-toolkit/* /var/www/html ,as it provides stable and long-term hosting capabilities which was a crucial requirement for testing purposes. This was linked to a dynamic DNS hostname on NO-IP and was made accessible at http://attackingmachine.ddns.net/phishing-page.html. For automating the interaction between the end user (employee machine) and malicious link a script was launched in the end user machine as currently there is no GUI access of the employee machine available. The script is programmed in the click_phishing_link.sh which ensures that the phishing link is accessed at 30 second interval time. This script is run with the nohup command. While for DDoS attack, the file server's port 80 is targeted using the hping3 -S -p 80 –flood 98.83.82.139. This command is executed from the docker container 59f67d529812 to isolate the attack traffic. This command generates huge number of SYN packets at a very high rate. Thus, suricata then receives all these logs as the traffic is mirrored from the employee machine and the whole detection process is initiated.

6 Evaluation

This section would evaluate the performance and functionality of the implemented cloud-based real-time security system through mainly two aspects which is system performance (depicting the efficiency and stability of the detection mechanism) and detection accuracy

(showing the effectiveness of the rule-based and ML-based in identifying the DDoS and phishing attacks). The results are mainly analysed against metrics such as detection times, system reliability, classification performance aligning with the research question's objectives.

6.1 System Performance

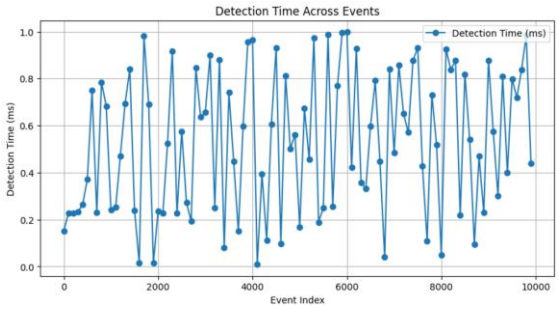

Figure 3: Detection Time across events

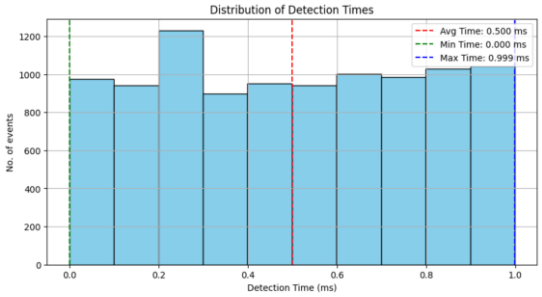

Figure 4: Distribution of Detection Times

The graph above in Figure 3, illustrates the detection times for individual events showing fluctuations in detection time mostly between 0.2 ms and 1.0 ms. The detection mechanism remains consistent throughout the events without major spikes and delays. The overall detection time remains within the in a suitable sub-millisecond range which depicts that the system is able to rapidly process the events. Also, even after the variability in detection times the upper limit of a maximum 1.0 ms is reflecting the real-time responsiveness. The system's ability to handle large volume of data consistently maintaining minimal detection timeline aligns well with the research's goal of real-time monitoring. This consistent performance depicted in the result increases the reliability of the system.

The graph in Figure 4, illustrates the frequency distribution of detection times which ranges between min. time of 0.0 ms up to max. time of 0.999 ms while the average detection being 0.5 ms. The significant finding from this the efficiency of the monitoring system with the average detection time being 0.5 ms, it highlights the system's ability to detect events instantly. The low detection scores for a dataset of 10000 events also depicts the scalability of the system reflecting that it can handle larger workloads efficiently. This evaluation depicts the robustness of the system's detection mechanism even when the events also contain the high-density traffic flows such as during the DDoS simulation, the system can still handle this traffic without any performance degradation.

6.2 Confusion Matrix Analysis

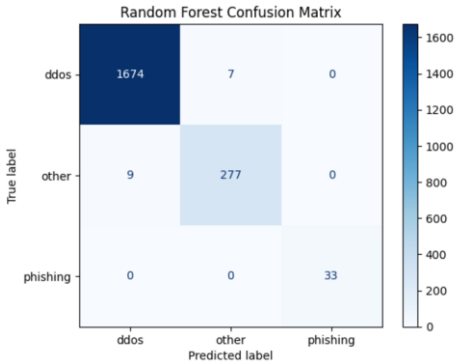

Figure 5: Confusion matrix- RF

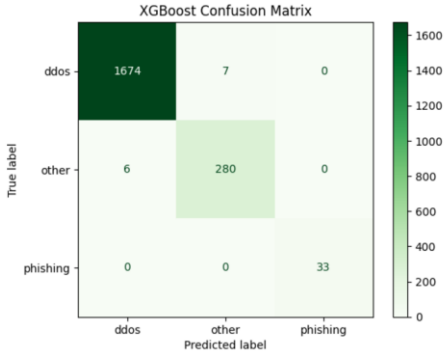

Figure 6: Confusion matrix- XGBoost

RF- The confusion matrix in the Figure 5 shows the performance of the RF model in predicting the two attacks. The true DDoS predictions here indicate that there are 1674

entries correctly identified as DDoS, the true Phishing predictions indicate that a total of 33 predictions are available as phishing and lastly a total of 277 entries in are labelled as other in true other predictions. While there are a few minimal entries which are misclassified with 7 instances of DDoS misclassified as Other and 9 instances of Other misclassified as DDoS. The matrix shows a very high accuracy rate of detections by the model wherein no false positives are present for phishing, while only 7 samples from DDoS entries were misclassified. The matrix proves the model's robust performance in detecting and tagging the phishing and DDoS attacks which aligns perfectly with the research's aim of improving real-time monitoring system and thus the integration of this model's predictions with SIEM ensures a robust detection mechanism. However, the matrix also defines the future scope of refinement in feature engineering section with certain misclassifications being done between DDoS and other category.

XGBoost- The confusion matrix in the Figure 6 depicts the XGBoost's performance in detecting the three categories: DDoS, Phishing and other. With very little difference from the other model even this model has 33 true predictions correctly identified as phishing, 1674 instances correctly identified as DDoS and 280 entries classified correctly as Other. With a reduced misclassification number as 6 instances are wrongly tagged as DDoS when they are Other and 7 instances wherein DDoS is wrongly tagged as Other. Similar to RF, phishing detection is completely accurate as all the instances are correctly marked as phishing and a near-perfect score for identifying the DDoS instances indicating that the XGBoost model is also performing well. The results in the matrix show great reliability for the model in terms of detection capabilities. High precision and recall with lower false positive rates would drastically minimize the workload for an analyst using the system. However further enrichments are required in the features section so that minor misclassifications can be addressed and overcome to increase the overall efficiency.

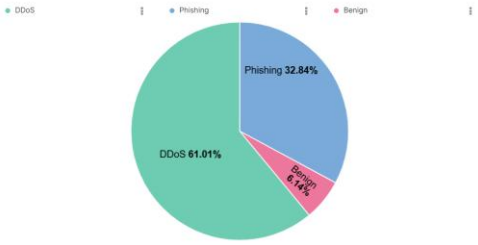

Figure 7: Tag Distribution: DDoS, Phishing & Benign

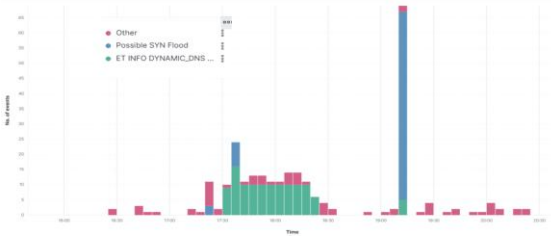

Figure 8: Rule-based detection

6.3 Distribution of Detected events (ML predictions- Kibana Visualization)

The pie chart in figure 7 describes the distribution of detected events on the basis of ML-based predictions. These predictions are derived from separate dataset in Kibana from a different timeline making sure the accuracy is maintained even with dynamic input data reflecting real-time log analysis differing with the static dataset used for the confusion matrix. The graph shows the accurate findings of the ML models wherein the DDoS accounts for 61.01% of the total logs analysed which makes sense given the huge traffic DDoS attacks create. Next notable finding was that the phishing has accounted for 32.84% classifications depicting the attack simulated through the automated script is correctly captured by the models. While only 6.14% traffic classified as benign confirms that the logs are analysed in a precise way as the maximum number of traffic captured by IDS is related to security events only due to the launched simulated attacks. The graph highlights the adaptable capabilities of

the models across different datasets (real-time logs) which was the core fundamental of the research question. The integration of ML-based predictions with the SIEM demonstrates real-world application of the selected models hence proving the research objective of creating a real-world solution for security monitoring. This constantly updating visualization dashboard on Kibana would help the security teams to efficiently handle the different attack types and maintain the security of the company infrastructure.

6.4 Detection over Time (Rule-based- Kibana Visualization)

The bar chart in figure 8 is describing the distribution of detections done over time by the rule-based detection mechanism in the SIEM. It has categorized events in mainly 3 parts wherein ET_INFO DYNAMIC_DNS refers to phishing, Possible SYN Flood is classified as DDoS and non-classified traffic is categorized as Other. There is a significant spike in the time duration of 17:30 to 19:30 depicting concentrated malicious activity being carried out during the period. While phishing events are consistently appearing due to the continuous access to malicious link, there can be seen a sharp rise in the Possible SYN flood events indicating DDoS attacks due to the nature of this attack. The clear differentiation in tagging the attack types shows the system's ability to flag the different threat patterns in real-time effectively. This validates the ability of rule-based system in the SIEM to identify and classify events helping to strengthen the defence mechanism for the security teams. However the system is performing really well with these two attacks, there is a huge scope of expanding the rules set to identify and flag emerging threats which can practically enhance the detection coverage.

6.5 Discussion

The above findings are critical in understanding the valuable insights which the hybrid detection system has to offer with the integration of SIEM, IDS and ML models in identifying the phishing and DDoS attacks. This discussion will critically evaluate these results, highlighting the system's strengths and weaknesses comparing it with existing research and providing the suggestions for future growth.

The system's performance which is measured through detection time analysis has an average detection time of 0.5 milliseconds and a maximum of 0.999 milliseconds indicating its suitability in real-world applications. This shows the efficiency of the system in handling and processing the events as swiftly as required in a dynamic cloud-based environment. However, there are certain spikes in the detection time across some events which possibly indicate in occasional computational delays which might be caused due to resource contention or overheads in data preprocessing. For maintaining the consistent real-time performance there is a space for further optimization in terms of resource allocation and multi-threading capabilities. Both the confusion matrices for the ensemble learning models are demonstrating high accuracy in terms of detection, with nearly perfect detection classification for the DDoS category. But it has been noticed that XGBoost slightly outperforms RF in predicting the 'Other' events as XGBoost is known to work well with imbalanced datasets. Although the phishing detection class exhibits limited recall, which can possibly be due to the smaller dataset size, these findings are consistent with existing research, highlighting the challenges in detecting the phishing events due to their vast evolving nature. However further enrichments can be done here by working on the feature sets with additional parameters to improve the model's performance to be updated with the ever-evolving threats landscape. The kibana visualizations of event distributions shows that the DDoS attacks account for 61.01% of all the total traffic, followed by phishing activity at around 32.84% leaving 6.14% traffic tagged as benign. This predominance of DDoS attack aligns perfectly with the high-frequency network traffic generated during the simulated attack

but also leaves a room for improvement in the detection coverage of phishing simulations. Moreover, the rule-based system effectively identifies the spikes in activities showcasing the system's real-time ability in capturing the flooding of SYN events validating the systems robustness in dealing with high volumetric attacks.

With the context of prior research conducted in this field, this research demonstrates a practical implementation of a hybrid detection mechanism in a cloud environment. Unlike the prior studies which mostly rely on conceptual frameworks this research works under real-time conditions showcasing the use of XGBoost and RF effectively in security applications such as SIEM and IDS and thus providing hands-on evidence of the system's applicability in addressing the gaps like scalability, adaptability and real-time performance.

7 Conclusion and Future Work

The research initially addressed critical challenge of developing a real-time hybrid security solution designed in cloud-based environment focusing primarily on DDoS and phishing attacks. The research question was stated to discover how the SIEM, IDS and ML models can be integrated effectively to achieve a scalable and adaptive threat detection system. The research's objective was to design and evaluate the framework under simulated attacks replicating the real-world scenarios. Through the developments done in this research the implemented framework has successfully demonstrated its ability in detecting threats in real-time by combining the strengths of rule-based and data-driven (ML-models) approach. Key matrix in the evaluation section including detection time and classification accuracies validate the system' efficiency in addressing the real-time threats.

This research work proves that the integration of the SIEM, IDS as well as ML model technologies can be vital in developing a stronger and enhanced framework in order to combat the current threats posed in a dynamic cloud environment. The hybrid detection system scored an average detection time of 0.5 milliseconds depicting the robust real-time detection abilities. The system also underlines the potential strengths of combining rule-based and ML-based detections rather than being used as isolated solutions in a security architecture by visualizing the detections made by both rules and ML-predictions on kibana. The system's real-time detection highlights its applicability in real-world environments. Through this the system has achieved its objectives of designing, implementing and evaluating a scalable hybrid detection system. The findings of this research are important to both academic researchers and real-world practitioners as they provide a link between theoretical and practical applications. Academically, it provides substantial evidence in integrating ML techniques with SIEM and IDS focusing on the practical approach and for industries, the framework puts forward a robust system by enhancing real-time monitoring in cloud-based environment by an adaptive approach in threat detection. While the research exhibits the potential of the hybrid system it also displays several limitations associated with the research. Attack scenarios were performed in a controlled environment which at times fails to capture all the possible variations of an attack which can occur in the actual environment. Furthermore, the system faced little challenges in detecting the phishing attacks possible due to dataset imbalance and feature set used in the classification. These limitations focus the need for increasing the complexity in the approach and exporting richer datasets from the system for more enhanced feature engineering to understand the evolving nature of phishing attacks.

Future works:

- Real-world integration: Although our system is designed to work under the real-world conditions, it is only made possible through simulations and thus it is critical in validating the system with real-world network traffic logs to capture the unpredictability and

diversity in the data. This can be possible through partnering with organisations so that the system could get high volume of real-world logs.

- Adaptive Learning models: Future developments of the system can be achieved by integrating self-adaptive models which will dynamically update based on different network activities or attack patterns. This will make the system to be more reliable in identifying the zero-day attacks.
- Commercialisation potential: The proposed framework is robust enough to be deployed in a small and medium sized enterprise (SMEs) by working on user-friendly interface and packaging the system for commercial use as it can turn out to be a reliable, scalable as well as cost-effective detection strategy for them.

References

Ahmadi, S. (2024) 'Network Intrusion Detection in Cloud Environments: A Comparative Analysis of Approaches', *International Journal of Advanced Computer Science and Applications(IJACSA)*, 15(3), 2024. http://dx.doi.org/10.14569/IJACSA.2024.0150301

Allure Security (2024) *Fraudsters Abuse Dynamic DNS Subdomains For Phishing*. Available at: https://alluresecurity.com/fraudsters-abuse-dynamic-dns-subdomains/ [Accessed 20 October 2024].

Ariffin, S. H. S., Chong, J. L. and Wahab, N. H. A. (2021) 'Configuring local rule of intrusion detection system in software defined IoT testbed', in *2021 26th IEEE Asia-Pacific Conference on Communications (APCC)*. Kuala Lumpur, Malaysia, 1-13 October 2021, 298-303. doi: http://dx.doi.org/10.1109/APCC49754.2021.9609824.

Ayu, M.A., Erlangga, D., Mantoro, T. and Handayani, D. (2024) 'Enhancing Security Information and Event Management (SIEM) by Incorporating Machine Learning for Cyber Attack Detection', in *2023 IEEE 9th International Conference on Computing, Engineering and Design (ICCED)*. Kuala Lumpur, Malaysia, 07-08 November 2023, IEEE Xplore. doi: 10.1109/ICCED60214.2023.10425288

Çakmakçı, S. D., Hutschenreuter, H., Maeder, C. and Kemmerich, T. (2021) 'A Framework for Intelligent DDoS Attack Detection and Response using SIEM and Ontology', in *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*. Montreal, QC, Canada, 14-23 June 2021, pp. 1-6, doi: 10.1109/ICCWorkshops50388.2021.9473869.

Chen, Z., Jiang, F., Cheng, Y., Gu, X., Liu, W. and Peng, J. (2018) 'XGBoost Classifier for DDoS Attack Detection and Analysis in SDN-Based Cloud', in *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*. Shanghai, China, 15-17 January 2018, pp. 251-256, doi: 10.1109/BigComp.2018.00044.

Deacon, A. (2023) *DNS as a vector for phishing attacks, different victims, different methodologies, different results*. Available at: https://dnsrf.org/blog/dns-as-a-vector-for-phishing-attacks--different-victims--different-methodologies--different-results/index.html [Accessed 20 October 2024].

Dhahir, Z. S. (2024) 'A Hybrid Approach for Efficient DDoS Detection in Network Traffic Using CBLOF-Based Feature Engineering and XGBoost', *Journal of Future Artificial Intelligence and Technologies*. ID, 1(2), pp. 174–190. doi: 10.62411/faith.2024-33.

Dutta, A. K. (2021) 'Detecting phishing websites using machine learning technique', *PLoS ONE*, 16(10): e0258361. doi: https://doi.org/10.1371/journal.pone.0258361.

González-Granadillo, G., González-Zarzosa, S., & Diaz, R. (2021) 'Security Information and Event Management (SIEM): Analysis, Trends, and Usage in Critical Infrastructures', *Sensors*, *21*(14), 4759. https://doi.org/10.3390/s21144759.

Hariawan, F. R. and Sunaringtyas, S. U (2022) 'Design an Intrusion Detection System, Multiple Honeypot and Packet Analyzer Using Raspberry Pi 4 for Home Network', in *2021 17th International Conference on Quality in Research (QIR): International Symposium on Electrical and Computer Engineering*. Depok, Indonesia, 13-15 October 2021, pp. 43-48, doi: 10.1109/QIR54354.2021.9716189.

Innab, N., Atoum, I., Alghayadh, F., Abu-Zanona, M., Alrubayyi, N. and Basudan, F. (2024) 'Intrusion Detection System Mechanisms in Cloud Computing: Techniques and Opportunities, in *2024 2nd International Conference on Cyber Resilience (ICCR)*. Dubai, United Arab Emirates, 26-28 February 2024, pp. 1-5, doi: 10.1109/ICCR61006.2024.10532903.

Lee, J-H., Kim, Y. S., Kim, J. H. and Kim, I. K. (2017) 'Toward the SIEM architecture for cloud-based security services', in *2017 IEEE Conference on Communications and Network Security (CNS)*. Las Vegas, NV, USA, 09-11 October 2017, pp. 398-399, doi: 10.1109/CNS.2017.8228696.

Morkos, R. (2023) *Powering the Growth of Cloud Computing: Infrastructure Challenges and Solutions*. Available at: https://www.forbes.com/councils/forbestechcouncil/2023/07/24/powering-the-growth-of-cloud-computing-infrastructure-challenges-and-solutions/ [Accessed 25 November 2024].

NCCS (2024) *Traffic Generator/DDoS Tool*. Available at: https://nccs.gov.in/public/events/DDoS%20Presentation%2017092024.pdf [Accessed 15 October 2024].

Oracle (2023) *Design Guidance for SIEM Integration*. Available at: https://docs.oracle.com/en-us/iaas/Content/cloud-adoption-framework/siem-integration.htm?u [Accessed 27 November 2024].

Proofpoint (2024) *Community Alert: Ongoing Malicious Campaign Impacting Microsoft Azure Cloud Environments*. Available at: https://www.proofpoint.com/us/blog/cloud-security/community-alert-ongoing-malicious-campaign-impacting-azure-cloud-environments [Accessed 25 November 2024].

Rochim, A. F., Aziz, M. A. and Fauzi, A. (2020) 'Design Log Management System of Computer Network Devices Infrastructures Based on ELK Stack', in *2019 International Conference on Electrical Engineering and Computer Science (ICECOS)*. Batam, Indonesia, 02-03 October 2019, pp. 338-342, doi: 10.1109/ICECOS47637.2019.8984494.

Saeed, M. S., Saurabh, R., Bhasme, S. R. and Nazarov, A. N. (2022) 'Machine Learning Based Intrusion Detection System in Cloud Environment', in *2022 VIII International*

Conference on Information Technology and Nanotechnology (ITNT). Samara, Russian Federation, 23-27 May 2022, pp. 1-6. doi: 10.1109/ITNT55410.2022.9848611.

Salahdine, F., Mrabet, Z. E. and Kaabouch, N. (2022) 'Phishing Attacks Detection A Machine Learning-Based Approach', in *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. New York, NY, USA, 01-04 December 2021, pp. 0250-0255, doi: 10.1109/UEMCON53757.2021.9666627.

STAMVS Networks (2024) *Suricata vs Snort*. Available at: https://www.stamus-networks.com/suricata-vs-snort#:~:text=Suricata's%20ability%20to%20handle%20high,networks%20with%20heavy%20traffic%20loads [Accessed 10 October 2024].

Tuyishime, E., Balan, T. C., Cotfas, P. A., Cotfas, D. T., and Rekeraho, A. (2023) 'Enhancing Cloud Security—Proactive Threat Monitoring and Detection Using a SIEM-Based Approach', *Applied Sciences*, *13*(22), 12359. https://doi.org/10.3390/app132212359.

Waldman, A. (2024) 'Microsoft confirms DDoS attack disrupted cloud services', *TechTarget*, 31 July. Available at: https://www.techtarget.com/searchsecurity/news/366599523/Microsoft-confirms-DDoS-attack-disrupted-cloud-services [Accessed 25 November 2024].

Waleed, A., Jamali, A.F. and Masood, A. (2022) 'Which open-source IDS? Snort, Suricata or Zeek', *Computer Networks*, 213, p. 109116. doi: https://doi.org/10.1016/j.comnet.2022.109116.

Yasar, K. (2022) *Elastic Stack (ELK Stack)*. Available at: https://www.techtarget.com/searchitoperations/definition/Elastic-Stack [Accessed 20 October 2024].