A Model for the Detection of Denial-of-Service Attack Patterns

Bennett, E. O[1] & Nwala, Q. N[2]
bennett.okoni@ust.edu.ng[1] & queen.nwala@ust.edu.ng[2]
DOI: 10.56201/ijcsmt.v11.no2.2025.pg26.39

Abstract
Denial of Service (DoS) attacks is a critical cybersecurity threat that can disrupt network services by overwhelming it with illegitimate traffic. This paper presents a hybrid detection and mitigation system to address DoS attacks. The system leverages Python as the primary programming language, incorporating its robust ecosystem of libraries such as Scikit-learn, TensorFlow, and Tkinter for machine learning, feature extraction, and GUI development. The system integrates a hybrid model combining K-means clustering with a Random Forest classifier. Initially, the K-means algorithm groups data into clusters, which are then fed into the Random Forest for classification. The system was trained and evaluated using an undersampled DoS dataset to ensure balanced class representation. Results show that the model performed quite excellently, achieving a 98.97% accuracy in detecting DoS attacks, compared to other models such as XGBoost (98.48%) and multidimensional feature approaches (98.96%). The model's deployment on a web-based platform demonstrated its ability to filter and classify incoming network traffic into normal or attack types like SIDDOS and UDP-Flood in real-time, effectively mitigating the threats posed by DoS attacks.

Keywords: *Denial of Service (DoS), Random Forest Classifier, k-means, Cybersecurity.*

1. Introduction

Denial of service (DoS) is a malicious activity aimed at rendering a computer, operating system, or server inaccessible to its intended users by disrupting the host services connected to the Internet. This can be achieved through intensive attacks by internal or external perpetrators, leading to the depletion of network and service resources, resulting in the unavailability of the targeted nodes. Furthermore, Denial of Service (DoS) attacks is prevalent in specific networks such as Vehicular Ad Hoc Networks (VANETs), occurring through internal or external vehicles, thereby impacting the network's functionality [1].

This study aims to bridge these gaps by designing a system to quickly identify and mitigate malicious traffic and detect emerging threats and vulnerabilities in network logs.

2. Related Works

Mitigating the effects of DoS attacks is an intense area of research, with a focus on exploring potential approaches to counter these threats One such approach involves the use of Software-Defined Networking (SDN) technologies, particularly in IoT scenarios, to mitigate DDoS attacks. Furthermore, the integration of knowledge-driven Software-Defined Networking technology has been suggested to enhance the security of IoT networks and mitigate the impact of DDoS attacks.

Machine learning (ML) is a discipline of artificial intelligence that focuses on creating algorithms and statistical models to increase the performance of computer systems in a particular activity by utilizing experience [2]. In recent years, machine learning has emerged as a promising option for resolving classification difficulties and has demonstrated its promise in solving binary classification problems [3]. The importance of machine learning is apparent in its capacity to manage large-scale data environments and adjust to intricate jobs [4]. The following are the types of ML algorithm:

Supervised Learning: In supervised learning, a collection of examples or training samples is given along with their corresponding correct outputs. The algorithm then learns to improve its accuracy by comparing its output with the provided inputs. Supervised learning, alternatively referred to as learning from examples or learning from exemplars, involves the use of labelled data to train a model.

Unsupervised Learning: Unsupervised learning involves identifying unknown patterns in data to extract rules. This strategy is suitable when the data categories are uncertain. In this case, the training data lacks labels. Unsupervised learning is a statistical approach to learning that involves discovering hidden patterns in data without labels.

Semi-Supervised Learning: These algorithms adopt a method that combines both supervised and unsupervised learning capabilities. In specific scenarios, a few observations may get labels, but the majority remain unlabelled due to the expensive nature of labelling and a shortage of trained human expertise. Semi-supervised methods are most appropriate for constructing models in such circumstances. Semi-supervised learning applies to regression, classification, and prediction [5]. It can also be classified as Generative Models, Transductive SVM and Self-Training.

Hybrid Learning: Despite initially being seen as a solution to the challenges of computational complexity, overfitting, and local minima in classification algorithms, researchers have discovered issues with ensemble learning. The intricate combination of many classifiers poses challenges in implementation and result analysis. Ensembles, rather than enhancing the model's accuracy, may amplify errors at the level of each base learner. Ensembles can lead to decreased accuracy due to the inclusion of subpar classifiers in the combination. A recent method to address these issues is hybridisation, which involves establishing an ensemble of diverse models. This involves integrating multiple methods, such as combining clustering with a decision tree or clustering with association mining.

Application K-means on Denial of Service Attacks

Utilising K-means clustering can be a beneficial method for tackling the problem of DoS attacks. DoS attacks, characterised by the utilisation of botnets to generate substantial volumes of traffic aimed at a specific target, provide considerable obstacles to network security [6]. Within the realm of cybersecurity, pertinent data can be employed to construct automated systems that rely on data analysis. In particular, K-means clustering can be utilised on cybersecurity datasets to enhance the process of analysing and identifying potential threats [7]. Moreover, the growing occurrence of DoS attacks on IoT networks emphasises the necessity for strong security measures. Implementing DoS attack mitigation strategies, such as those provided by SDN technologies, can play a crucial role in tackling these challenges [8]. Furthermore, researchers have investigated the application of machine learning algorithms, such as K-means clustering, to identify potential attacks in communications involving Unmanned Aerial Vehicles (UAVs). This highlights the importance of utilising these approaches in the field of cybersecurity [9].

K-means clustering is used in cybersecurity to analyse several types of data, including multimodal clinical data and single-cell RNA-sequencing data in the healthcare domain [10]. Furthermore, the importance of K-means clustering in cybersecurity is emphasised by its utilisation in threat modelling methodologies for APT-style attacks, emphasising its value in detecting and mitigating potential security risks [11]. The importance of utilising clustering techniques such as K-means for data encryption and secure routing in wireless sensor networks is underscored by the necessity for strong security measures to defend against both passive and active attacks [12]. Furthermore, the susceptibility of medical devices to cyber-attacks requires the investigation of strong cybersecurity measures, in which K-means clustering and other machine learning approaches might be pivotal in identifying threats and managing risks [13].

Ultimately, the utilisation of K-means clustering to tackle DoS attacks is substantiated by its applicability in cybersecurity datasets, threat modelling, healthcare systems, and wireless sensor networks. Organisations may improve their cybersecurity position and successfully reduce the impact of DoS attacks by utilising K-means clustering and machine learning methods.

Application of Random Forest Classifier on Distributed Denial of Service Attacks

Distributed Denial of Service (DDoS) attacks pose a great risk to computer networks, including Internet of Things (IoT) and industrial systems [14]. These attacks employ botnets to create a substantial amount of traffic, which overwhelms the target and renders it incapable of functioning. To tackle this DDoS attacks on network logs, there has been a growing interest in utilising machine learning methods, specifically the random forest classifier. The random forest technique is renowned for its capacity to effectively manage extensive and intricate datasets, rendering it well-suited for the identification and alleviation of DDoS attacks [15]. The technique utilised involves the selection of a random subset of features while performing the splitting procedure. This helps to decrease the correlation between the trees in a bagging sample [16]. This methodology has been effectively utilised in several scientific fields, such as the categorization of neuroimaging data in Alzheimer's disease [17].

The reviewed literature proves that the use of the random forest classifier has the capacity to efficiently identify and alleviate Distributed Denial of Service (DDoS) attacks in computer networks, IoT devices, and industrial contexts. The random forest algorithm provides a viable method to improve the security and resilience of network infrastructures against DDoS threats by utilising its ability to handle complicated datasets and its shown performance in several fields.

3. Methodology

The adoption of Object-Oriented Analysis and Design (OOAD) stems from the following justifications:

Modularity and Reusability:

OOAD promotes breaking down a system into modular components or objects. In the context of predictive fault maintenance, this means that different aspects of the system (e.g., data processing, predictive algorithms, user interface) can be developed and maintained separately. This promotes reusability of code, which can save time and effort in the long run.

Maintainability and Scalability:

OOAD encourages good design practices, which makes the codebase easier to maintain over time. This is particularly important for a system like predictive fault maintenance where updates and improvements are likely to be ongoing. Additionally, a well-designed system can be more easily scaled to accommodate additional equipment or features.

Flexibility and Adaptability:

OOAD allows for flexibility in design, making it easier to adapt the system to changing requirements. In an industrial setting, requirements may evolve due to changes in equipment, processes, or regulations. OOAD provides a framework that allows for these changes to be incorporated more smoothly.

3.1 System Design

System design is the process of designing the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through the system. Figure 1 shows the use case diagram of the system.

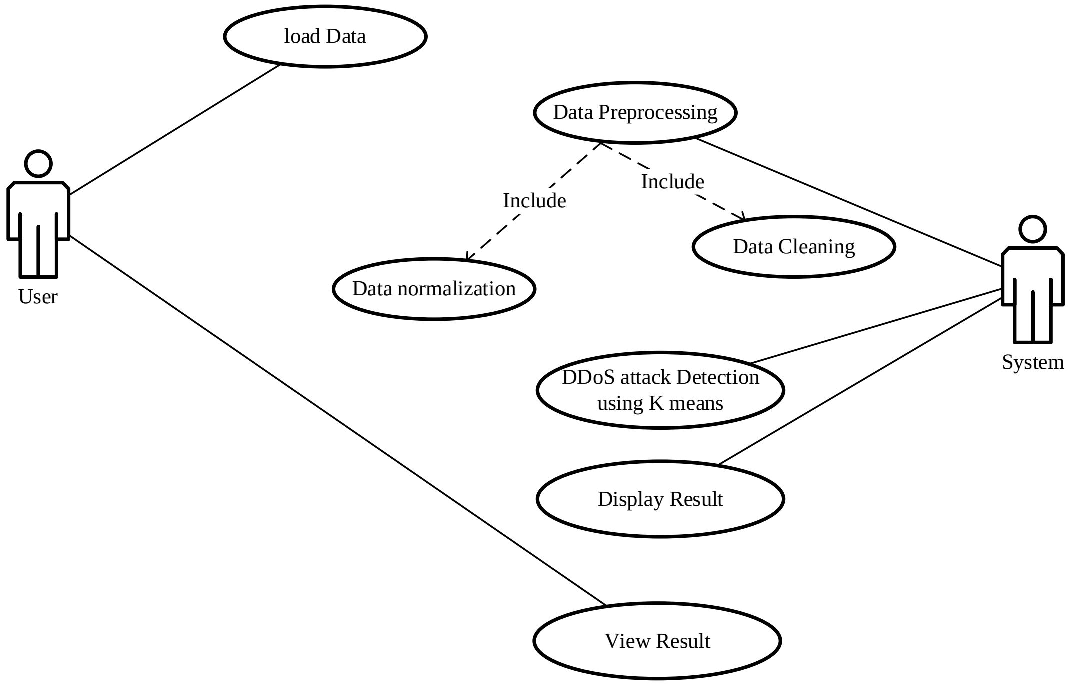
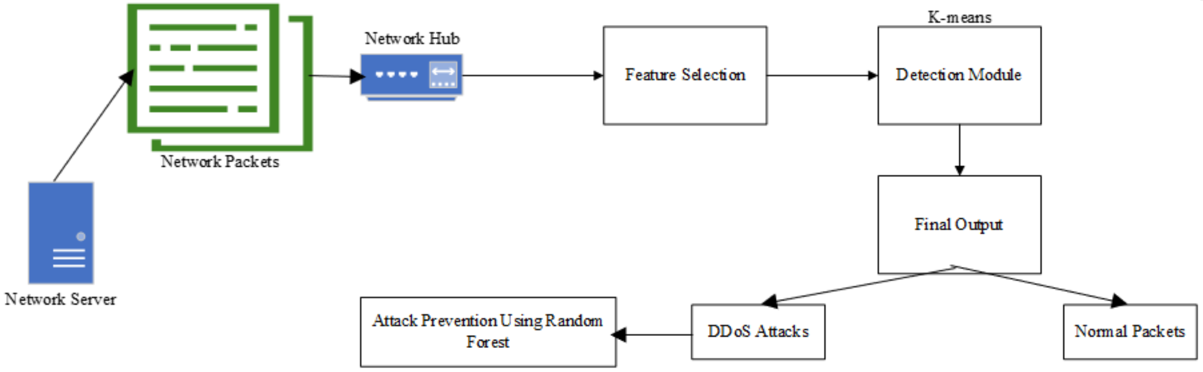


Fig. 1: system use-case diagram

3.2 Architectural Design of the System

The architectural design shown in figure 2 encompasses the structural framework and components necessary for effectively identifying and mitigating Distributed Denial of Service attacks within a network environment. This design entails the arrangement of modules, algorithms, and data flows to facilitate real-time monitoring, analysis, and response to potential threats. Key considerations in the architectural design include scalability to handle varying traffic loads, integration with existing network infrastructure, and optimization for high-performance detection algorithms.



The system adopts a hybrid model, combining k-means clustering with random forest classification; two primary stages are involved. Firstly, the k-means algorithm partitions network traffic data into distinct clusters based on similarity in features, effectively grouping together normal and anomalous traffic patterns. Subsequently, these clusters serve as input features for the random forest classifier, which leverages ensemble learning to accurately classify each cluster as either benign or indicative of a DDoS attack. Through this hybrid approach, the model benefits from both the clustering capabilities of k-means for unsupervised learning and the predictive power of random forest for robust classification, thereby enhancing the system's ability to detect and mitigate DDoS attacks with improved accuracy and efficiency.

Random Forest Classifier: This is used for training the model for further detection of DDoS attacks.

Algorithm 3.1: Algorithm of the real time data stream processing

```
Initialize DDoS data.
Loop:
 a. Read incoming data from network traffic.
b. Preprocess data:
          i. Clean and normalize data.
          ii. Perform basic statistical analysis:
- Calculate mean (μ): $\mu = (1/N) * \Sigma(x_i)$
- Calculate variance (σ^2): $\sigma^2 = (1/N) * \Sigma(x_i - \mu)^2$
c. Extract relevant features from the preprocessed data.
d. Model and analyze the features:
          i. Apply k-means clustering algorithm to identify attack patterns.
          ii. Utilize Random Forest classifier for continuous detection.
e. Make decisions based on the analysis:
```

i. Set threshold for predicted probability of anomalous patterns
ii. Trigger alert or mitigation response if threshold is exceeded.
f. Output the results:
i. Display detected DDoS attacks and insights.

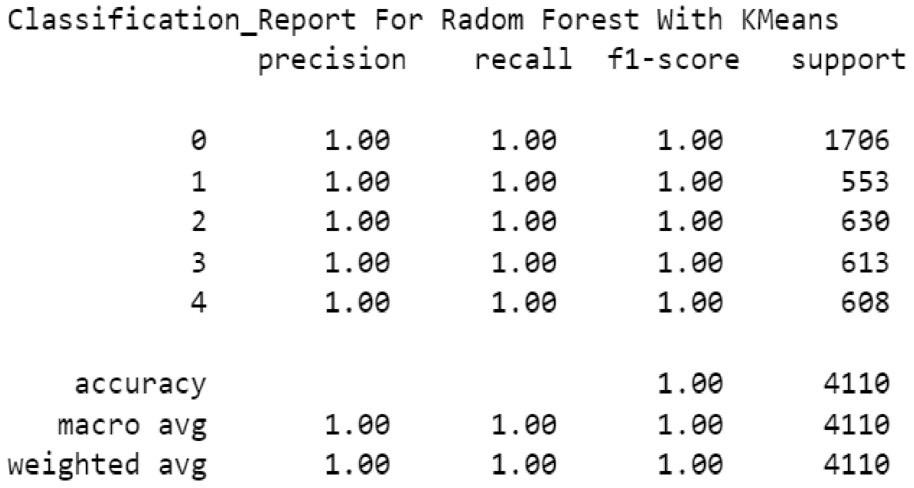
4. Results and Discussion

The K-means model was trained using sampled data, with a predetermined number of clusters set to 5. Significantly, the training dataset deliberately omitted the class column during this process. After the completion of the training process, the K-means model was utilised to forecast labels for subsequent data points. The recently obtained cluster labels were added to the original dataset, resulting in an expanded dataset that included the clustering information.

4.1 System Setup

The minimum hardware specification of the system includes: a system with processor speed of 2.5GHz or higher, 8GB of RAM, Hard disk of about 500GB

The software requirement include: Microsoft Windows, Anaconda (Python Distribution), Jupyter Notebook Development Environment



```
Classification_Report For Radom Forest With KMeans
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      1706
           1       1.00      1.00      1.00       553
           2       1.00      1.00      1.00       630
           3       1.00      1.00      1.00       613
           4       1.00      1.00      1.00       608

    accuracy                           1.00      4110
   macro avg       1.00      1.00      1.00      4110
weighted avg       1.00      1.00      1.00      4110
```

Figure 3: Classification Report of the Hybrid Model

Figure 3 provides the classification report for the Random Forest classifier, which was used in conjunction with K-Means clustering. The report evaluates the model's performance by presenting precision, recall, F1-score, and accuracy metrics for each class (ranging from class 0 to class 4). The perfect scores (1.00) across all classes indicate that the classifier has achieved excellent performance in identifying DDoS attack categories. The high precision demonstrates the model's accuracy in predicting positive instances, while the recall shows its ability to detect all true positives. The F1-score reflects the balance between precision and recall, further confirming the model's robustness.

Figure 4 shows the confusion matrix, which gives a detailed breakdown of the model's prediction results. The matrix shows how many correct and incorrect predictions were made for each class. It provides a visual understanding of where the model may have made errors and which classes were most accurately predicted

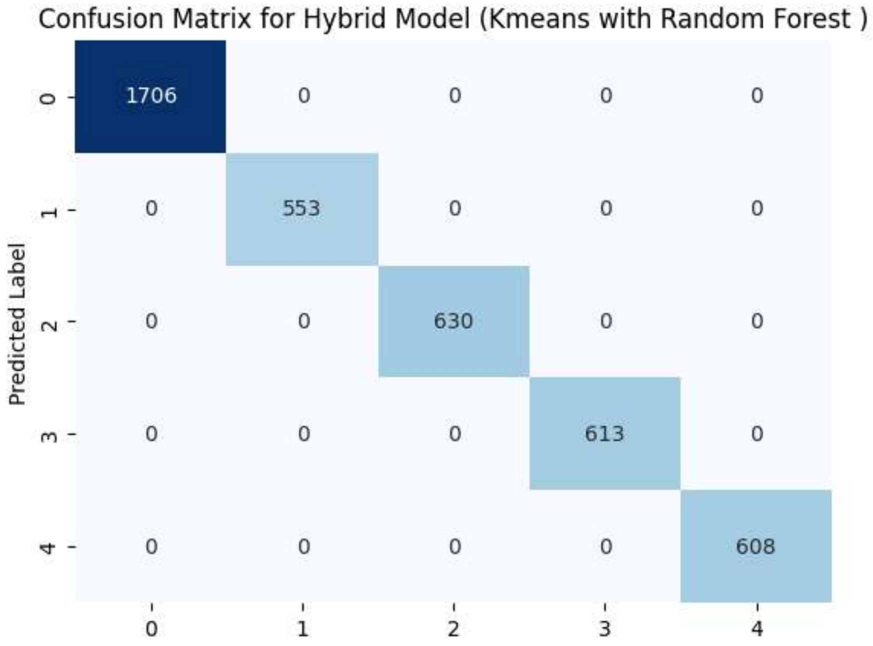
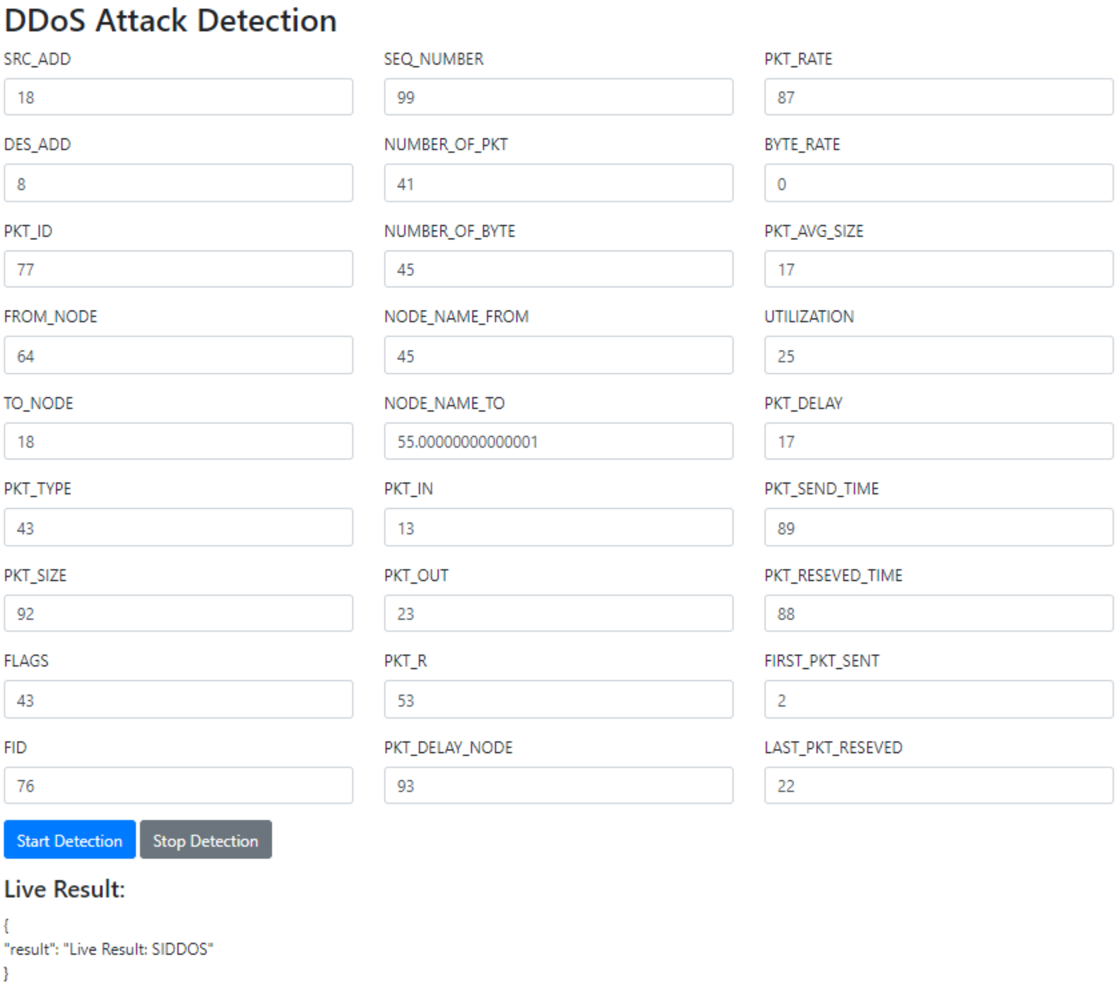


Figure 4: Confusion Matrix of the Hybrid Model

Figure 5 displays the initial deployment of the DDoS detection model on the web application interface. This figure focuses on the user interface, showing how network administrators can interact with the system. It provides a clean dashboard that displays essential statistics, such as network traffic data, and allows users to initiate traffic analysis.



DDoS Attack Detection

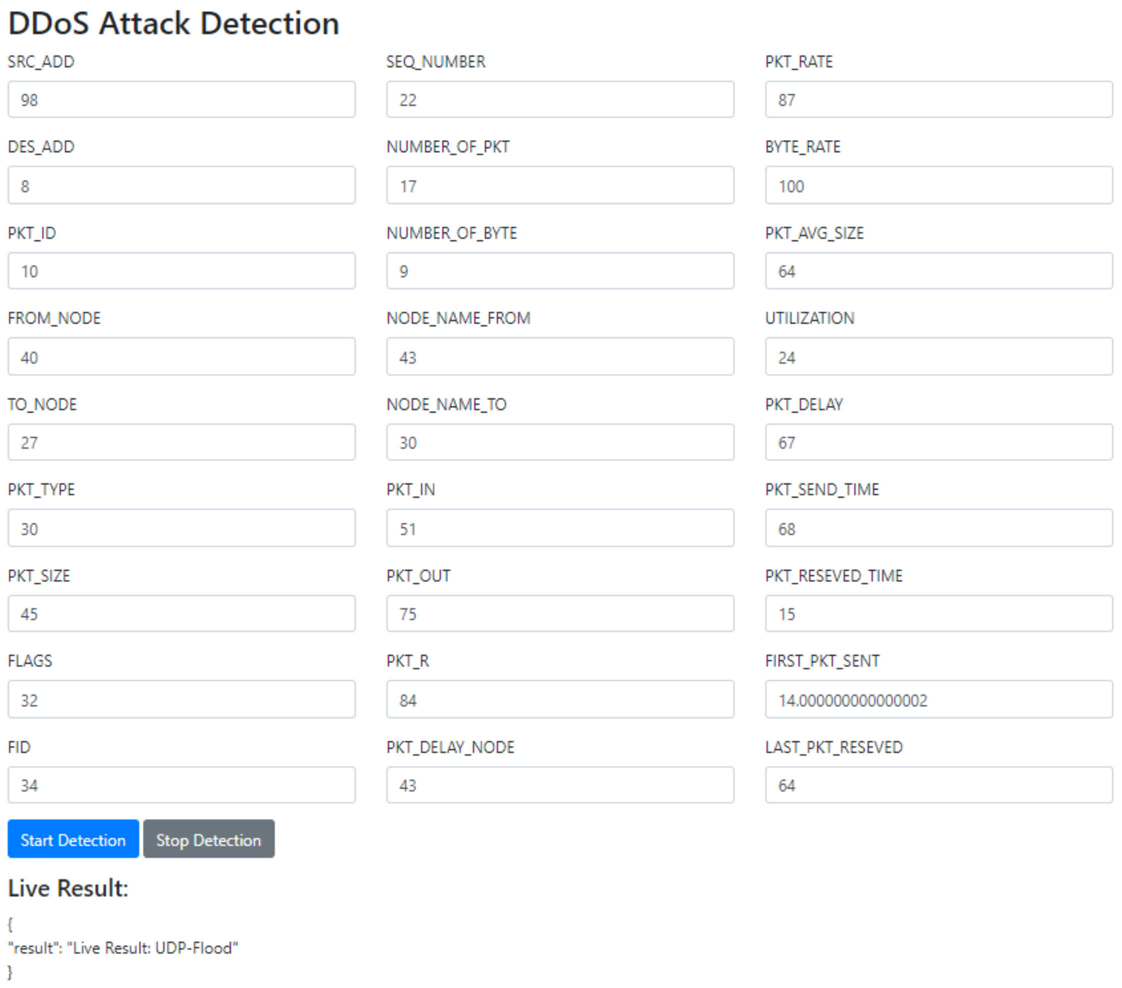
SRC_ADD	18	SEQ_NUMBER	99	PKT_RATE	87
DES_ADD	8	NUMBER_OF_PKT	41	BYTE_RATE	0
PKT_ID	77	NUMBER_OF_BYTE	45	PKT_AVG_SIZE	17
FROM_NODE	64	NODE_NAME_FROM	45	UTILIZATION	25
TO_NODE	18	NODE_NAME_TO	55.00000000000001	PKT_DELAY	17
PKT_TYPE	43	PKT_IN	13	PKT_SEND_TIME	89
PKT_SIZE	92	PKT_OUT	23	PKT_RESEVED_TIME	88
FLAGS	43	PKT_R	53	FIRST_PKT_SENT	2
FID	76	PKT_DELAY_NODE	93	LAST_PKT_RESEVED	22

Live Result:

```
{
"result": "Live Result: SIDDOS"
}
```

Figure 5: SIDDOS attack detected.

Figure 6 illustrates the real-time traffic monitoring capabilities of the web application. This figure shows how the system continuously tracks incoming network traffic, visualizing traffic flow and identifying potential anomalies. The dynamic nature of the traffic data is presented, offering administrators real-time updates on network activity.



DDoS Attack Detection

SRC_ADD	98	SEQ_NUMBER	22	PKT_RATE	87
DES_ADD	8	NUMBER_OF_PKT	17	BYTE_RATE	100
PKT_ID	10	NUMBER_OF_BYTE	9	PKT_AVG_SIZE	64
FROM_NODE	40	NODE_NAME_FROM	43	UTILIZATION	24
TO_NODE	27	NODE_NAME_TO	30	PKT_DELAY	67
PKT_TYPE	30	PKT_IN	51	PKT_SEND_TIME	68
PKT_SIZE	45	PKT_OUT	75	PKT_RESEVED_TIME	15
FLAGS	32	PKT_R	84	FIRST_PKT_SENT	14.000000000000002
FID	34	PKT_DELAY_NODE	43	LAST_PKT_RESEVED	64

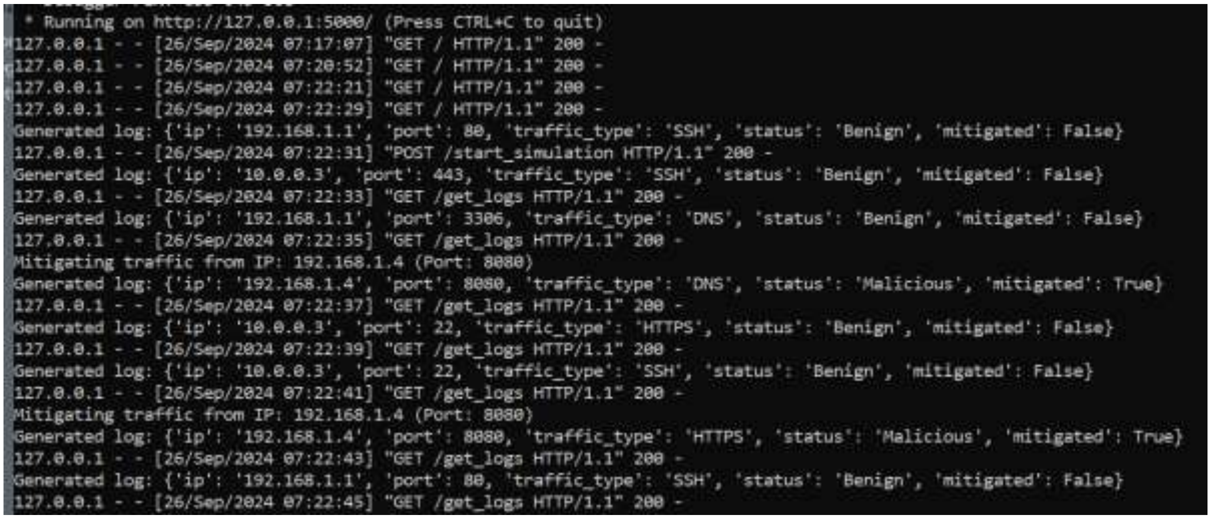
Live Result:

```
{
"result": "Live Result: UDP-Flood"
}
```

Figure 6: UDP-Flood DDoS attack detected.

Simulated environment to Identify and Mitigate Malicious Traffic

This section shows the simulated environment and command prompt that was shown to identify and mitigate malicious activities can be seen in Figures 7.

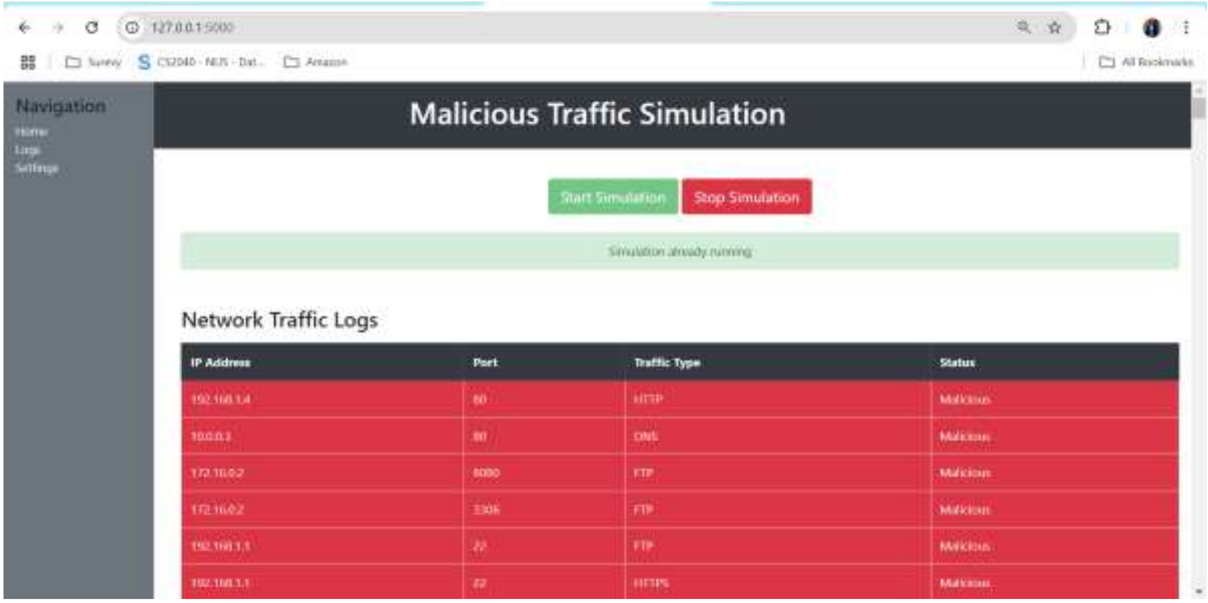


```
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [26/Sep/2024 07:17:07] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [26/Sep/2024 07:20:52] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [26/Sep/2024 07:22:21] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [26/Sep/2024 07:22:29] "GET / HTTP/1.1" 200 -
Generated log: {'ip': '192.168.1.1', 'port': 80, 'traffic_type': 'SSH', 'status': 'Benign', 'mitigated': False}
127.0.0.1 - - [26/Sep/2024 07:22:31] "POST /start_simulation HTTP/1.1" 200 -
Generated log: {'ip': '10.0.0.3', 'port': 443, 'traffic_type': 'SSH', 'status': 'Benign', 'mitigated': False}
127.0.0.1 - - [26/Sep/2024 07:22:33] "GET /get_logs HTTP/1.1" 200 -
Generated log: {'ip': '192.168.1.1', 'port': 3306, 'traffic_type': 'DNS', 'status': 'Benign', 'mitigated': False}
127.0.0.1 - - [26/Sep/2024 07:22:35] "GET /get_logs HTTP/1.1" 200 -
Mitigating traffic from IP: 192.168.1.4 (Port: 8080)
Generated log: {'ip': '192.168.1.4', 'port': 8080, 'traffic_type': 'DNS', 'status': 'Malicious', 'mitigated': True}
127.0.0.1 - - [26/Sep/2024 07:22:37] "GET /get_logs HTTP/1.1" 200 -
Generated log: {'ip': '10.0.0.3', 'port': 22, 'traffic_type': 'HTTPS', 'status': 'Benign', 'mitigated': False}
127.0.0.1 - - [26/Sep/2024 07:22:39] "GET /get_logs HTTP/1.1" 200 -
Generated log: {'ip': '10.0.0.3', 'port': 22, 'traffic_type': 'SSH', 'status': 'Benign', 'mitigated': False}
127.0.0.1 - - [26/Sep/2024 07:22:41] "GET /get_logs HTTP/1.1" 200 -
Mitigating traffic from IP: 192.168.1.4 (Port: 8080)
Generated log: {'ip': '192.168.1.4', 'port': 8080, 'traffic_type': 'HTTPS', 'status': 'Malicious', 'mitigated': True}
127.0.0.1 - - [26/Sep/2024 07:22:43] "GET /get_logs HTTP/1.1" 200 -
Generated log: {'ip': '192.168.1.1', 'port': 80, 'traffic_type': 'SSH', 'status': 'Benign', 'mitigated': False}
127.0.0.1 - - [26/Sep/2024 07:22:45] "GET /get_logs HTTP/1.1" 200 -
```

Figure 7: Command line that identifies and mitigate malicious traffic in network logs

The final model was deployed to web for continuous detection of DDoS attacks. A simulation was carried out to replicate the packets that flows into the network system. The model filters the packets and categorizes the packets into normal packets and further types of DDoS attacks.

Figure 8 highlights the detection alert system integrated into the web application. When the deployed model identifies suspicious traffic, the system generates an alert, notifying administrators of potential DDoS attacks. This figure shows how the alert is displayed on the interface, offering details about the type of attack detected and its severity.



Malicious Traffic Simulation

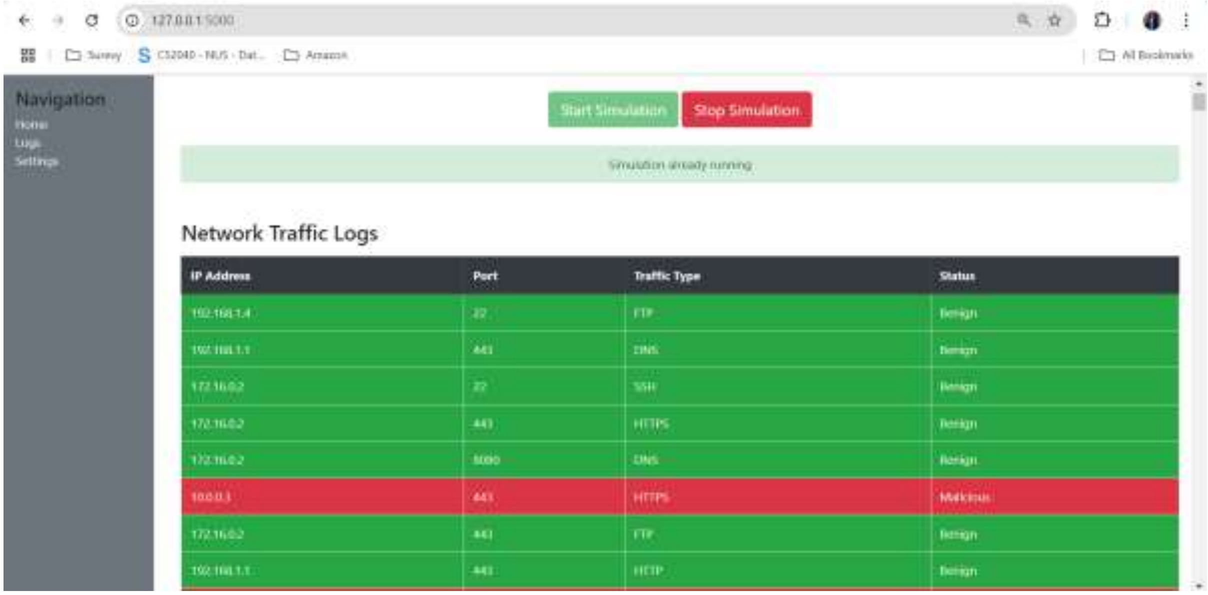
Simulation already running

Network Traffic Logs

IP Address	Port	Traffic Type	Status
192.168.1.4	80	HTTP	Malicious
10.0.0.3	80	DNS	Malicious
172.16.0.2	8080	FTP	Malicious
172.16.0.2	3306	FTP	Malicious
192.168.1.1	22	FTP	Malicious
192.168.1.1	22	HTTPS	Malicious

Figure 8: Simulated environment identifying malicious traffic in network logs

Figure 4.6 showcases the system’s ability to identify malicious traffic in network logs. It presents how the system processes network logs to detect irregularities associated with DDoS attacks.



Simulation already running

Network Traffic Logs

IP Address	Port	Traffic Type	Status
192.168.1.4	22	FTP	Benign
192.168.1.1	443	DNS	Benign
172.16.0.2	22	SSH	Benign
172.16.0.2	443	HTTPS	Benign
172.16.0.2	8080	DNS	Benign
10.0.0.3	443	HTTPS	Malicious
172.16.0.2	443	FTP	Benign
192.168.1.1	443	HTTP	Benign

Figure 9: Simulated environment identifying some normal and malicious traffic

The figure emphasizes the system's accuracy in pinpointing traffic patterns that deviate from normal behaviour, providing insights into when and where malicious activities occur. It zooms in on specific instances of identified DDoS attacks, offering a comprehensive analysis of the traffic anomalies. The figure highlights key parameters such as the source IP address, the volume of traffic, and the specific attack type. Figure 4.8 demonstrates the system's ability to activate mitigation responses when malicious traffic is detected. Once a DDoS attack is identified, the system triggers an automated response to neutralize the threat, such as blocking malicious IPs or limiting traffic flow.

5 Conclusion

This paper developed a system that leverages a real-time monitoring mechanism to detect unusual traffic patterns and respond automatically. K-Means clustering was applied to group similar traffic patterns and quickly identify anomalies that may indicate a DDoS attack. When suspicious traffic is detected, the system mitigates it by blocking or rate-limiting malicious requests, ensuring minimal network disruption.

REFERENCES

1. Sheikh, M., Liang, J., & Wang, W. (2019). A survey of security services, attacks, and applications for vehicular ad hoc networks (vanets). *Sensors*, 19(16), 3589.
2. Htun, T. &Tun, Z., (2018). Algorithm tuning from comparative analysis of classification algorithms. *International Journal of Research and Studies Publishing*, 8(5).
3. Smeureanu *et al*., 2013. Customer segmentation in private banking sector using machine learning techniques. *International journal of business economics and management, 14(5), pp.923-939.*
4. Cil, A.E., Yildiz, K. & Buldu, A., (2021). Detection of DDoS attacks with feed forward based deep neural network model. *Expert Systems with Applications*, 169, 114520
5. Sandhya, N. & Charanjeet, K.R., (2016). A review of machine learning techniques. *International Journal on Recent and Innovation Trends in Computing and Communication*, 4(3), 451-458.
6. Pedreira, V., Barros, D. & Pinto, P., (2021). A review of attacks, vulnerabilities, and defenses in industry 4.0 with new challenges on data sovereignty ahead. *Sensors*, 21(15), p.5189.
7. Sarker, I. (2021). Data science and analytics: an overview from data-driven smart computing, decision-making and applications perspective. *Sn Computer Science*, 2(5).
8. Mrabet, H., Belguith, S., Alhomoud, A. & Jemai, A., (2020). A survey of IoT security based on a layered architecture of sensing and data analysis. *Sensors*, 20(13), p.3625.
9. Bithas, P., Michailidis, E., Νομικός, N., Vouyioukas, D., & Kanatas, A. (2019). A survey on machine-learning techniques for uav-based communications. *Sensors*, 19(23), 5170.
10. Horne, E., Tibble, H., Sheikh, A., & Tsanas, A. (2020). Challenges of clustering multimodal clinical data: review of applications in asthma subtyping. *Jmir Medical Informatics*, 8(5), e16452.

11. Tatam, M., Shanmugam, B., Azam, S., & Kannoorpatti, K. (2021). A review of threat modelling approaches for apt-style attacks. *Heliyon*, 7(1), e05969.
12. Djedouboum, A., Ari, A., Gueroui, A., Mohamadou, A., & Aliouat, Z. (2018). Big data collection in large-scale wireless sensor networks. *Sensors*, 18(12), 4474.
13. Kruse, C., Frederick, B., Jacobson, T., & Monticone, D. (2017). Cybersecurity in healthcare: a systematic review of modern threats and trends. *Technology and Health Care*, 25(1), 1-10.
14. Pedreira, V., Barros, D., & Pinto, P. (2021). A review of attacks, vulnerabilities, and defenses in industry 4.0 with new challenges on data sovereignty ahead. *Sensors*, 21(15), 5189.
15. Sufriyana, H., Husnayain, A., Chen, Y., Kuo, C., Singh, O., Yeh, T., … & Su, E. (2020). Comparison of multivariable logistic regression and other machine learning algorithms for prognostic prediction studies in pregnancy care: systematic review and meta-analysis. *Jmir Medical Informatics*, 8(11), e16503.
16. Seo, H., Khuzani, M., Vasudevan, V., Huang, C., Ren, H., Xiao, R., … & Li, X. (2020). Machine learning techniques for biomedical image segmentation: an overview of technical aspects and introduction to state-of-art applications. *Medical Physics*, 47(5).
17. Sarica, A., Cerasa, A., & Quattrone, A. (2017). Random forest algorithm for the classification of neuroimaging data in alzheimer's disease: a systematic review. *Frontiers in Aging Neuroscience*, 9.