Using a Combination of Effective Feature Selection Methods and an Entropy-based Approach to Identify DDoS Anomalies

Basheer Husham Ali [1], Khaled Mansour Al-Rawe [2], Mohammed A. Ahmed [3], Ali J. Askar Al-Khafaji [4], Nasri Sulaiman [5]

[1] Dept. of Electrical and Electronic Engineering, Faculty of Engineering, Universiti Putra Malaysia, 43400 Serdang, Malaysia
Email: gs58547@student.upm.edu.my

[2] College of Administration and Economics, Al-Iraqia University, Baghdad, Iraq
Email: khaled.mansour@aliraqia.edu.iq

[3]Institute of IR 4.0, Universiti Kebangsaan Malaysia, Bangi, Malaysia
Email: p103761@siswa.ukm.edu.my

[4]Razak Faculty of Technology and Informatics Universiti Teknologi Malaysia (UTM) Kuala Lumpur, Malaysia
Email: jal-khafaji@graduate.utm.my

[5]Dept. of Electrical and Electronic Engineering, Faculty of Engineering, Universiti Putra Malaysia, 43400 Serdang, Malaysia
Email: nasri_sulaiman@upm.edu.my

Article History

Received: May 27, 2025

Revised: Aug. 12, 2025

Accepted: Aug. 22, 2025

Abstract

Distributed Denial of Service (DDoS) attacks are among the most dangerous types of attacks. These kinds of attacks bring targeted servers down and make their services unavailable to legal users. The first objective of this study is to identify infected Ethernet and detect various kinds of up-to-date DDoS attacks using a dynamic threshold by implementing multiple features of entropy and the Sequential Probabilities Ratio Test approach (E-SPRT). The second is to select relevant features to improve the performance of detection by implementing a new combination of machine learning techniques, which are ANOVA, Extra Trees Classifier, Random Forest, and Correlation Matrix with Pearson Correlation approaches. Canadian Institute for Cybersecurity (CIC-DDoS2019) databases were utilised to evaluate the implementation. ESPRT using a feature selection approach with five features achieved an accuracy of over 97% with an average False Positive Rate (FPR) close to 0 in identifying most different kinds of DDoS attacks.

Keywords- DDoS Attack, Entropy, Feature Selection, SPRT.

I. INTRODUCTION

DDoS attacks refer to Distributed Denial of Service attacks. These attacks pose a significant cybersecurity threat. These attacks are designed to disrupt the access of legitimate users to services provided by a targeted server or controller. To achieve this, cyber attackers exploit vulnerabilities in network systems, commandeering compromised machines to generate a massive volume of network packets or flows directed at a specific victim, which could be a server, controller, or device. Consequently, the targeted victim experiences disruption, rendering its services inaccessible to authorised users [1].

DDoS is one of the most dangerous attacks targeted at SDN networks and IoT devices [2]. The intensity, complexity, rates, and frequency of DDoS anomalies have increased and reached new records. Attackers are driven to send a very large number of useless packets toward their victims to disrupt their services. These volumetric attacks can be categorised using three metrics as shown in Figure 1. The first metric is bits per second (bps) that targets network connections. The second metric is packets per second (pps) that target network devices and DNS servers. The third metric is requests of HTTPs per second (rps), which targets application servers [3], [4].

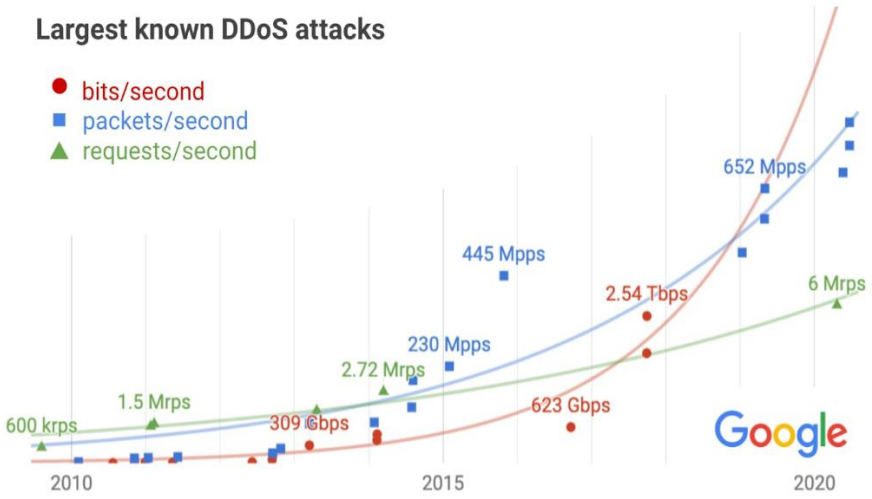

Figure 1: Trends in DDoS Attack Volumes based on Different Metrics [3], [4]

Feature selection is the process of choosing the most relevant feature among a set of them to build an optimal intrusion model. Feature selection involves selecting the fewest number of features in order to increase the accuracy of the model. Adding more irrelevant features to the dataset may lead to an increase in the complexity of the model. The model may take a long time to process a very large number of features in the dataset compared with models that use useful features. Feature selection also involves removing repetitive, unimportant, and meaningless features from the dataset to increase the detection rate and reduce errors in the detection of a certain model.

In addition, statistical models are techniques that rely on analysing and investigating network traffic to obtain a deeper understanding of these traffics. These approaches are able to identify variables to be used as inputs and find a common connection among these inputs to predict output behavior. Statistical approaches can detect attacks at an early stage. Sequential Probability Ratio Test (SPRT) and entropy are some examples of these effective techniques that can be used to detect DDoS attacks.

Likewise, an entropy-based approach is used to measure randomness. When many flows target a specific host, the destination IP address appears frequently, reducing randomness and entropy. This can provide some indication about the attack availability. SPRT analyses incoming features that were extracted from traffic and monitors switch ethernets that let these traffic pass through. SPRT is a statistical technique introduced by Wald and based on mathematical calculation. It uses two hypothesis values, H1 and H2, to differentiate between normal and compromised flows and to locate compromised interfaces. H1 is used to determine interfaces that have been injected with normal flows, while H2 is used to determine those injected with compromised flows based on feature values of flows [5]. SPRT can provide quick feedback about the status of incoming traffics in the early stage. Thus, combining both entropy and SPRT approaches is an effective way to detect DDoS attacks. It eliminates the dependence on using a static threshold of entropy by combining the entropy approach with the SPRT method in order to make decisions.

Finally, incoming traffic contains features that can be used as input for the detection stage. The CICFlowMeter tool can be used to extract these features. Some of these features are irrelevant and may increase execution time or cost. Thus, feature selection is an important step before the detection stage in order to remove meaningless, irrelevant, and repetitive features, which in turn reduces detection errors and increases the accuracy of the identification process. A combination of four feature selection methods was used which are ANOVA [6], Extra Tree [7], Random Forest [8], and Pearson Correlation Coefficient [9] to determine the important features. Then, the best group of features with the highest occurrences was chosen as inputs for the detection stage. The lower the number of features selected, the shorter the time required to get the results. Thus, 5 features were better than others in terms of execution time. However, the number of selected features may be better when the accuracy of detection is better.

II. RELATED WORKS

Several methods have been developed to detect and identify DDoS attacks in their early stages. This section will present a few kinds of approaches developed by many researchers to detect DDoS attacks. Several techniques of DDoS attacks have been established such as statistical-based techniques, machine learning-based techniques, and deep learning-based techniques. First of all, Statistical models are techniques that depend on investigating and analysing data to understand the data more strategically. These techniques can recognise variables as inputs and find common connections among these inputs to predict results. Many studies have been done by using statistical-based techniques in order to identify DDoS attacks. Statistical approaches provide a good in-depth analysis of incoming traffic in computer networks. These methods can detect anomalies in the early stage [10]. Statistical-based detection used different techniques to do DDoS detection, such as chi-squared goodness-of-fit test, hypothesis testing, standard deviation, entropy, covariance, and/or correlation matrix.

For example, Kousar et al. [11] implemented a DDoS detection method by using Apache Spark, which is based on distributed processing. Spark was designed to handle real-time data, and it is better than Hadoop which handles batch processing offline. Spark can handle data at a fast speed because it stores intermediate data in RAM during the read/write process, compared with Hadoop which uses diskette to read and write processes, which slows down processing speed. The implementation has two main phases. The first phase is the preprocessing of traffics header in real time by using Spark RDD. Features selection can also be done at this stage by using a correlation matrix, and results can be stored in Spark RDD. They used an NSL-KDD dataset to evaluate their method. This dataset contains 42 features, and they selected 7 features out of 42 features during feature selection. The second phase is training and detection by using Spark MLlib such as support vector machine, random forest, naïve base, and decision tree. The design detection system reduced time needed to identify DDoS attacks and increased the efficiency of detection due to using Spark technology.

Another example, St-Hilaire and Mousavi [12] presented an approach that used entropy calculation to detect DDoS attacks in SDN networks. They divided incoming flows into fixed sizes which are called window sizes, that were measured based on number of flows or time slot. They calculated entropies for the number of occurrences of each unique destination IP address in order to measure randomness. When very large flows are targeted at a specific host, the destination IP address will appear frequently. This leads to a decrease in randomness, and the entropy value is decreased. The authors used a threshold to make decision regarding availability of DDoS threats. When entropy values are less than threshold, there are a higher probability that a controller of SDN is under DDoS attacks. However, when entropy values are higher than threshold, there is a higher probability that incoming traffics are normal. Finally, they were able to detect DDoS attacks after five hundred infected traffics.

Another instance, Hoque et al. [13] introduced a method based on statistical calculation to detect threats. They used Feature Feature Score (FFSc) technique to determine malicious traffic from normal traffics. They first extract three features from incoming traffic. These features are variation of IP source address, entropy of IP source address, and packet rate. Variation of IP source address feature is the rate of change of source IP addresses per time. When sources of IP addresses are changed regularly, the variation of IP source addresses will increase as well. Entropy of IP source address feature can be computed by identifying IP source per time and then calculated entropy value for each set of sample traffic using Shannon entropy calculation. The packet rate is the number of traffic that is transmitted in one second. These three features are used to compute FFS during normal network traffic. Finally, their method produced 99.55% of accuracy on MIT dataset when the threshold was 0.05 and 96.6% when the threshold was 1

Finally, Özçelik and Brooks [14] suggested a detection system based on using combination of cumulative sum (CUSUM), wavelet, and entropy. First, they computed the entropy of packet header field in order to measure randomness of header fields such source IP address. The second stage was using wavelet transform to remove long term variations of entropy values to eliminate fall-out or false positive rate. Third stage is using the CUSUM detection method in order to identify sudden hidden growth in the background data. Finally, this approach produced high performance and efficiently in identifying DDoS attacks.

Moreover, machine learning-based techniques were also used to identify DDoS attacks. For example, Sharafaldin et al. [15] proposed a model that includes four machine learning based techniques. These techniques are logistic regressing, ID3, Naïve bayes, and Random Forest. CICFlowMeter was used to generate 80 features from incoming traffic. Random Forest Regressor was used to select useful features based on the importance of each feature. They generated a new dataset which is CICDDoS2019 and used it to test their design. Finally, ID3 had 78%, 65%, and 69% of precision, accuracy, and F1 score, respectively, and it was the highest result compared with the rest of the models that were used.

Finally, the combination of deep learning and machine learning was used in this regard. For instance, Gaur and Kumar [16] introduced a method to detect DDoS anomalies in IoT devices. They tested four machine learning algorithms which are XGBoost, Random Forest, Decision tree, and K-Nearest Neighbors. They also applied three different feature selection techniques with each classifier to select relevant features. These techniques are ANOVA, chi-square, and Extra tree. They presented confusion matrix results for each classifier with each feature selection techniques in the environment of cloud and using CICDDoS2019 dataset. This hybrid technique can quickly identify DDoS anomalies in IoT environment. However, parameters can be optimised in order to reduce overfitting and underfitting and increase accuracy.

III. DETECTION APPROACH

First of all, a combination of four selection methods was used to identify effective features. The results of the feature selections were collected. Then, the best 5 or 10 features with the highest occurrences were chosen as inputs for the next step. These methods include ANOVA, Extra Tree, Random Forest, and Pearson Correlation Coefficient.

The features selected using the previous techniques are fed into this stage. Incoming flows and the interfaces through which these flows pass toward the targeted devices are gathered into groups. Each group has a fixed size known as the window size. The window size can be determined based on the number of flows or a certain time interval. In the implementation, a fixed number of flows was used to determine the window size, which varies for each data trace or dataset. It can be determined through experimentation based on detection accuracy. For example, the range of window sizes for the DARPA dataset that generated high accuracy falls within 5 to 120 flows. However, the best number of flows for CICDDoS2019 was above 300 flows. Therefore, the best window size can be determined experimentally. For each group of flows, Shannon Entropy (E) will be calculated for the selected features only as shown below [12]:

$$E = -\sum_{0}^{n} \text{prob}(n) \ln \text{prob}\ (n) \qquad (1)$$

Where (n) is the number of unique feature values.

The results of the Entropy calculation will serve as input for the next stage, which is the SPRT technique. Finally, the SPRT takes a decision and determines whether the flows and their associated switch interfaces are normal or infected. The SPRT detection ($D\_SPRT_i^s$) monitors incoming flows based on their features values (FL1, FL2, …, FLi). It also identifies the switch interface (s) that allow these flows to cross over to the targeted machine. The detection ($D\_SPRT_i^s$) is the likelihood ratio between these observations, whether they are compromised or normal flows injected into the compromised interface (H2), and these observations injected into a normal interface (H1). Therefore, the detection formula can be formulated as follows [5]:

$$D\_SPRT_i^s = \ln \frac{prob\ (FL_1^s, \ldots.,\ FL_i^s\ |H_2\ )}{prob\ (\ FL_1^s, \ldots.,\ FL_i^s\ |H_1)} \quad (2)$$

Where (i) is the total number of flow observations. Let us assume that these observations (FL_i^s) are identically independent and distributed. Thus, the detection formula can be as follows [5]:

$$D\_SPRT_i^s = \sum_{v=1}^{i} \ln \frac{prob\ (FL_v^s|H_2\ )}{prob\ (FL_v^s\ |H_1\ )} \quad (3)$$

Where (v) is each value in a group of flow observations. Because (FL_i^s) can be as Bernoulli random variables, the detection will be as follow [5]:

$$prob\ (\ 0 \le FL_i^s \le 0.5|H_1\ ) = 1\text{- }prob\ (\ FL_i^s > 0.5|H_1) = \mu_1 \quad (4)$$

$$prob\ (\ 0 \le FL_i^e \le 0.5|H_2) = 1\text{- }prob\ (\ FL_i^s > 0.5|H_2) = \mu_2 \quad (5)$$

Where value of μ_2 is larger than the value of μ_1 since compromised interfaces are more likely to be injected with compromised flows to flood the targeted system with DDoS attacks. Switch interfaces are more likely to be injected with infected flows when FL_i^s is between 0 and 0.5. On the other hand, switch interfaces are more likely to have normal traffics when the value of FL_i^s is above 0.5. Therefore, the detection of SPRT can be shown in (6) [5].

$$D\_SPRT_i^s = \begin{cases} D\_SPRT_{i-1}^s + \ln \dfrac{prob\ (FL_i^s|H_2\ )}{prob\ (\ FL_i^s\ |H_1)}, & 0 \le FL_i^s \le 0.5 \\ D\_SPRT_{i-1}^s + \ln \dfrac{prob\ (FL_i^s\ |H_2\ )}{prob\ (FL_i^s|H_1)}, & FL_i^s > 0.5 \end{cases} \quad (6)$$

By substituting (4) and (5) in (6), the detection equation can be rewritten as shown in (7) [5].

$$D\_SPRT_i^s = \begin{cases} D\_SPRT_{i-1}^s + \ln \dfrac{\mu_2}{\mu_1}, & 0 \le FL_i^s \le 0.5 \\ D\_SPRT_{i-1}^s + \ln \dfrac{1-\mu_2}{1-\mu_1}, & FL_i^s > 0.5 \end{cases} \quad (7)$$

where $D\_SPRT_0^s = 0$. The detection technique of SPRT generates two types of errors that affect the accuracy of detection. These mistakes are the false positive errors λ_1 and a false negative error λ_2. The False positive error λ_1 occurs when the detection technique mistakenly considers normal interface H1 as a malicious interface H2. On the other hand, the false negative mistake λ_2 occurs when an infected Ethernet H1 is wrongly identified as a benign Ethernet. Lower bound (U) and upper bound (L) thresholds were calculated as shown in (8) to deal with these two errors [5].

$$\begin{cases} U = \log_2 \dfrac{\lambda_2}{(1-\lambda_1)} \\ L = \log_2 \dfrac{(1-\lambda_2)}{\lambda_1} \end{cases} \quad (8)$$

Finally, the detection result ($D\_SPRT_i^s$) for each observed flow that passes through a certain interface, checked with the upper and lower bound thresholds dynamically to make a decision. If the value of ($D\_SPRT_i^s$) is larger than or equal to U, then a monitored interface with their flows is marked benign, and the test will stop. However, if the value of ($D\_SPRT_i^s$) is less than or equal to L, then a monitored interface with their flows is marked compromised, and the test will stop. Finally, the detection test will continue by checking another flow observation when the above two conditions are not applied.

IV. FEATURE SELECTION RESULTS

This section will depict the results of feature selection methods used to select the best 20 features among 82 features that were extracted from January 12th datasets that are part of the CICDDoS2019 dataset [15]. The primary purpose behind employing these feature selection methods was to reduce execution time and enhance the performance of the detection method.

CICDDoS2019 stands for Canadian Institute for Cybersecurity Distributed Denial of Service attacks. It was designed by researchers from this institute and encompasses the latest DDoS intrusions, encompassing UDP, SYN, DNS, MSSQL, SSDP, UDP-lag, TFTP, SNMP, NETBIOS, and LDAP attacks. The dataset contains benign as well as actual DDoS malicious data captured and stored in Pcap

format, alongside a set of datasets stored in CSV file format. These CSV files were generated from Pcap files using CICFlowMeter, and they encompass labelled flows, each with multiple features produced by CICFlowMeter. Each flow was constructed from packets sharing common specifications, such as source/destination IP, source/destination port, and protocol type. Table I below shows all attack names in this dataset along with the duration of each attack.

TABLE I. DETAILS OF ATTACK TYPES FOR CICDDOS2019 DATASET

Days	Attack database	Attack times	Days	Attack database	Attack times
(January 12th)	NTP	10:35 - 10:45	(March 11th)	PortScan	9:43 - 9:51
	DNS	10:52 - 11:05		NetBIOS	10:00 - 10:09
	LDAP	11:22 - 11:32		LDAP	10:21 - 10:30
	MSSQL	11:36 - 11:45		MSSQL	10:33 - 10:42
	NetBIOS	11:50 - 12:00		UDP	10:53 - 11:03
	SSDP	12:27 - 12:37		UDP-Lag	11:14 - 11:24
	UDP	12:45 - 13:09		SYN	11:28 - 17:35
	UDP-Lag	13:11 - 13:15			
	SYN	13:29 - 13:34			
	TFTP	13:35 - 17:15			

The basic of choosing the techniques of ANOVA [6], Random Forest [7], Extra Tree [8], and Pearson correlation [9] among others was because these techniques generated higher value of sensitivity, specificity, accuracy, and F1-score comparing with other feature selection methods. These techniques also generated lower values of probability of false alarm and miss-rate compared with others. An experiment was conducted on six features selection techniques. ANOVA, Random Forest, Extra Tree, Pearson correlation, Chi-square, and Mutual Information approaches were trained and tested for that purpose. A subset of CICDDoS2019 dataset was chosen to run the experiment, and this subset was NETBIOS dataset. The train set of NETBIOS dataset contains 30,533 benign and 40,500 malicious instances. However, the test set contains 3,963,446 malicious instances and 3,100,100 benign instances.

The value of TPR, TNR, accuracy, and F1-score were close to 0.99 for ANOVA, Random Forest, Extra Tree, and Pearson correlation while values for these metrics for chi-square and mutual information are close to 0.8 in the training dataset as shown in Table II or even in the testing dataset as shown in Table III. On the other hand, values of FPR and FNR were close to 0.001 for ANOVA, Random Forest, Extra Tree, and Pearson correlation which is better than value of FPR and FNR which was close to 0.1 as shown in Table II and Table III below.

TABLE II. TRAINING CONFUSION MATRIX FOR DIFFERENT FEATURE SELECTION APPROACHES

Confusion Metrics	Feature Selection Techniques					
	ANOVA	*Random Forest*	*Extra Tree*	*Pearson correlation*	*Chi-square*	*Mutual Information*
TPR	0.9996	0.9994	0.9995	0.9998	0.8634	0.8714
FPR	0.0032	0.0035	0.0029	0.0016	0.1803	0.1
TNR	0.9967	0.9964	0.9970	0.9983	0.8196	0.9
FNR	0.0003	0.0005	0.0004	0.0001	0.1365	0.1285
Accuracy	0.9984	0.9981	0.9984	0.9991	0.8446	0.8831
F1-Score	0.9986	0.9983	0.9986	0.9992	0.8638	0.8978

TABLE III. TESTING CONFUSION MATRIX FOR DIFFERENT FEATURE SELECTION APPROACHES

Confusion Metrics	Feature Selection Techniques					
	ANOVA	*Random Forest*	*Extra Tree*	*Pearson correlation*	*Chi-square*	*Mutual Information*
TPR	0.9949	0.9899	0.9845	0.9830	0.8350	0.8011
FPR	0.0033	0.0098	0.02013	0.0081	0.0544	0.06715
TNR	0.9966	0.99016	0.9798	0.99186	0.9455	0.9328
FNR	0.0050	0.0100	0.0154	0.0169	0.1649	0.1988
Accuracy	0.9956	0.9900	0.9825	0.9874	0.8738	0.8442
F1-Score	0.9961	0.9911	0.9844	0.9874	0.8957	0.8737

A. ANOVA Results

An ANOVA was utilised to pinpoint the top 20 features among the 82 extracted via CICFlowMeter for each flow. These best features were then selected and ranked based on their respective weights. The optimal features for various attack datasets on the First Day of CICDDoS2019 are visualised in Figure 2. For instance, the primary feature for NETBIOS attack datasets was the protocol (f5), as indicated in Figure 2. As for the second and third top features, they were the URG flag (f52) and up/down ratio (f55), respectively.

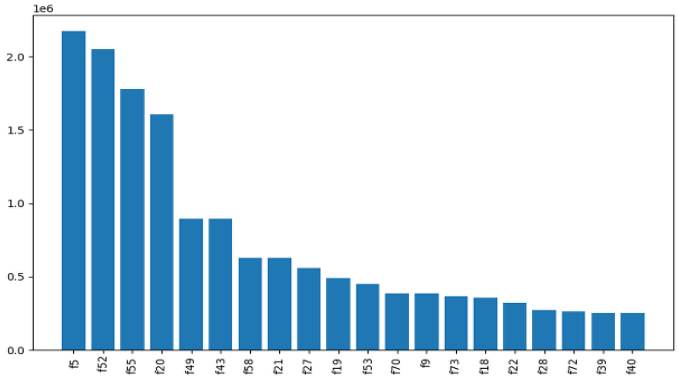

Figure 2: Best (20) Features of Different Attack Datasets for the First Day of CICDDoS2019 Using ANOVA Technique based on NETBIOS Attack Dataset

B. Extra Trees Classifier Results

The Extra Trees Classifier stands as another feature selection technique that was employed to identify the top twenty features, aiming to enhance execution speed and system performance. The protocol feature (f5) emerged as one of the three features with higher weights, selected through this method. Moreover, Min_seg_size_forward (f59) was consistently recognised as the fourth-best feature. Finally, the URG flag (f52) and up/down ratio (f55) were acknowledged as one of the top two selected features. Yet, its inclusion was limited to the best twenty features in the case of other datasets, as presented in the figure below. Bwd Packet Length Min (f20) secured a spot among the top five features for using the Extra Tree feature selection method.

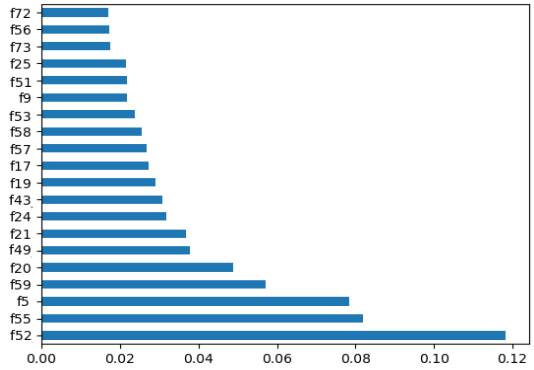

Figure 3: Best (20) Features of Different Attack Dataset for First Day of CICDDoS2019 Using Extra Tree Technique based on NETBIOS Attack Dataset

C. Random Forest Classifier Results

The Random Forest technique emerges as another feature selection method utilised in tandem with other approaches to extract impactful features. Min Packet Length (f24) and Fwd Packet Length Min (f16) secured positions among the first two selected features, carrying higher weights. Finally, Average Packet Size (f56) and Avg Fwd Segment Size (f57) also found their way into the selection for specific attack datasets NETBIOS, as shown in the figure below.

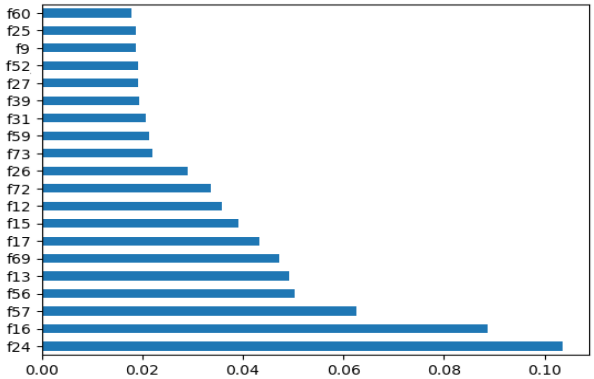

Figure 4: Best (20) Features of Different Attack Datasets for the First Day of CICDDoS2019 Using Random Forest Technique based on NETBIOS Attack Datasets

D. Correlation Matrix with Pearson Correlation Heatmap Results

The Pearson Correlation serves as a feature selection technique that ranks features based on their weights, derived from their correlation with a target label. To implement this method, a Correlation Matrix Heatmap was computed for various attack datasets on the first day of CICDDoS2019. The correlation between features and the label for the NETBIOS attack dataset was initially computed. Illustrated in Figure 5, this figure showcases the correlation among the top twenty selected features and each feature individually. Weight values of the features most correlated with the label are displayed in the first row. For instance, f5, with weights of 0.6, exhibits a strong correlation with the target label. Following suit, the second and third most correlated features, f24 and f16, hold weights of 0.095, respectively. The remaining features and their respective weights are detailed in Figure 5.

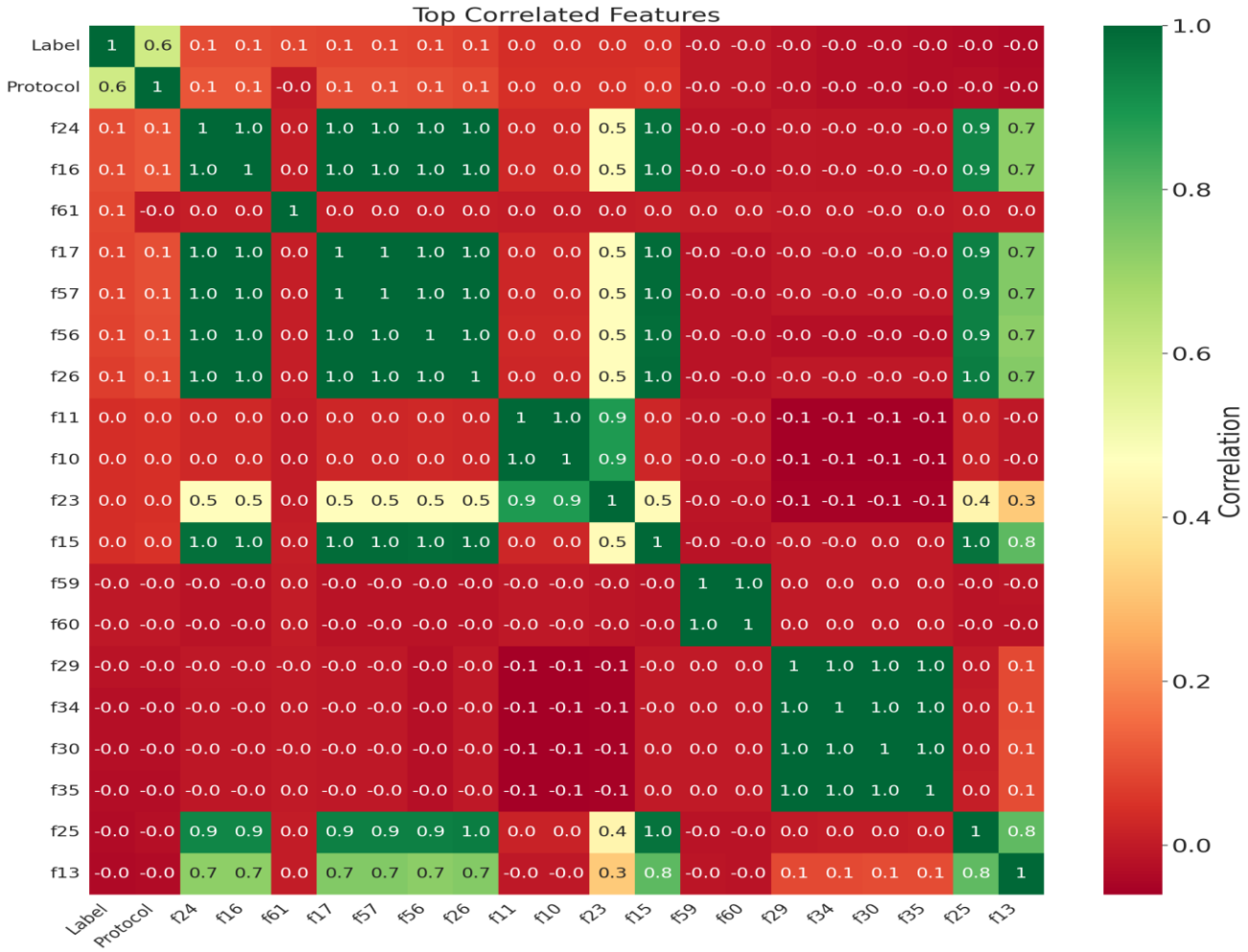

Figure 5: Best (20) Features of NETBIOS Dataset for First Day of CICDDoS2019 Using Pearson Correlation Heatmap

Finally, as shown above from figure 2 to figure 5, the best 20 features were selected from the pool of 82 features extracted from the January 12th datasets within the CICDDoS2019 dataset. The primary purpose behind employing these feature selection methods was to reduce execution time and enhance the performance of the detection method. Then, Features selected from each attack dataset employing techniques including ANOVA, Extra Tree, Random Forest, and Pearson Correlation are amalgamated, and the frequency of each unique feature is computed. Subsequently, the top ten features with the highest frequency are earmarked as inputs for the subsequent stage—the detection method.

V. CONFUSION MATRIX METRICS

Confusion Matrix [17], also known as an error matrix, is a tabular representation that illustrates the performance of a detection system. It constitutes a distinctive form of table, termed a contingency table, possessing two dimensions: "actual" and "predicted". Both dimensions encompass the same set of "classes". Each entry in the contingency table signifies a combination of a class and a dimension. This table is constructed based on four fundamental metrics: True Positive (T_P), True Negative (T_N), False Positive (F_P), and False Negative (F_N). All other metrics within the confusion matrix are derived from these core metrics. Primarily, the True Positive Rate (TPR), also referred to as sensitivity, recall, or hit rate, denotes the proportion of correctly identified malicious flow

traffic among all actual malicious flow traffic. Sensitivity (S) can be computed using the following:

$$S = \frac{T\_P}{T\_P+F\_N} \quad (9)$$

Moreover, the True Negative Rate (TNR) or specificity is another metric that can be calculated based on four main metrics. It is the average of normal flow traffic that was predicted correctly as normal flows. It is also called specificity or selectivity. It can be computed based on (10). The False Positive Rate (FPR) is the average of normal flow traffic that was incorrectly predicted by the detection algorithm as compromised traffic. It is also called the probability of a false alarm or fall-out. It can be calculated by Equation (11). Similarly, the False Negative Rate (FNR) or Miss Rate is the rate of compromised traffics that was incorrectly predicted as normal traffics. It is also called Miss Rate. It can be computed using Equation (12) below. Finally, the higher values of sensitivity and specificity and lower values of miss rate and probability of false alarm indicate a higher accuracy of the detection system.

.

$$\text{Specificity} = \frac{T\_N}{T\_N+F\_P} \quad (10)$$

$$\text{Probability of False Alarm} = \frac{F\_P}{F\_P+T\_N} \quad (11)$$

$$\text{Miss Rate} = \frac{F\_N}{T\_P+F\_N} \quad (12)$$

Furthermore, the positive predictive value (PPV) or precision is the rate of normal flows resulting in detection algorithms that are truly identified as true positive. While negative predictive value (NPV) is the rate of infected flows results in identification algorithm that are truly identified as true negative. The higher values of precision and NPV are indications of higher accuracy of the detection system. These two metrics can be computed as shown in Equation (13) and Equation (14).

$$\text{precision} = \frac{T\_P}{T\_P+F\_P} \quad (13)$$

$$\text{NPV} = \frac{T\_N}{F\_N+T\_N} \quad (14)$$

In addition, the False discovery rate (FDR) is the proportion of normal flows that are falsely identified as compromised flows by the detection algorithm, and it can be calculated using Equation (15). False omission rate (FOR) is the proportion of compromised flows that are incorrectly identified as normal flows by the detection algorithm, and it can be calculated using Equation (16 The lower the values of FDR and FOR, the better the accuracy of the detection system.

$$\text{FDR} = \frac{F\_P}{T\_P+F\_P} \quad (15)$$

$$\text{FOR} = \frac{F\_N}{F\_N+T\_N} \quad (16)$$

Finally, Accuracy is a commonly used metric to measure the performance of a detection system, but it may not always provide a complete picture of the system's effectiveness, especially when dealing with imbalanced datasets. In such cases, F1-score can be a better metric as it takes into account both precision and recall. A higher F1-score indicates a better balance between precision and recall and hence a better performance of the detection system. The following two equations are used to calculate Accuracy and F1-score, respectively:

$$\text{Accuracy} = \frac{T\_P+T\_N}{T\_P+T\_N+F\_N+F\_P} \quad (17)$$

$$\text{F1\_score} = \frac{2 * T\_P}{2 * T\_P+F\_P+F\_N} \quad (18)$$

A. Sensitivity vs. Probability of False Alarm

The relationship between sensitivity and the probability of false alarm metrics for all features of the ESPRT detection technique across the NETBIOS attack in CICDDoS2019 is presented in Figure 6. For the NETBIOS attack dataset, most of the features used in ESPRT generated a lower FPR and higher sensitivity. However, features such as f5, f9, f55, f59, f70, and f74 generated false alarm errors above 0.9 as shown in Figure 6.

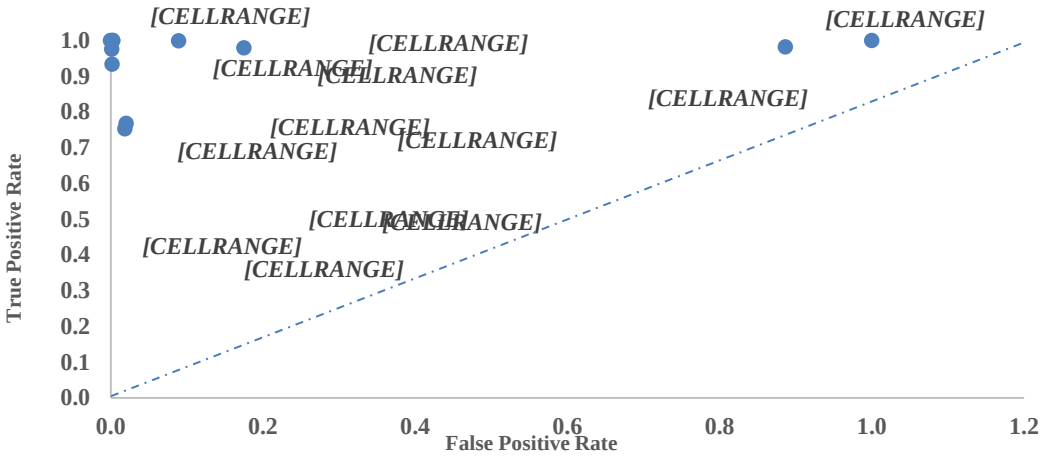

Figure 6: Sensitivity (TPR) vs Probability of False Alarm (FPR) for All Features of ESPRT Using Different DDoS Attacks of CICDDoS2019 Dataset based on NETBIOS Dataset

B. Specificity vs. Miss Rate

The connection between specificity and Miss Rate metrics of all features in the ESPRT identification method for different types of attacks in CICDDoS2019 is presented in Figure 7.

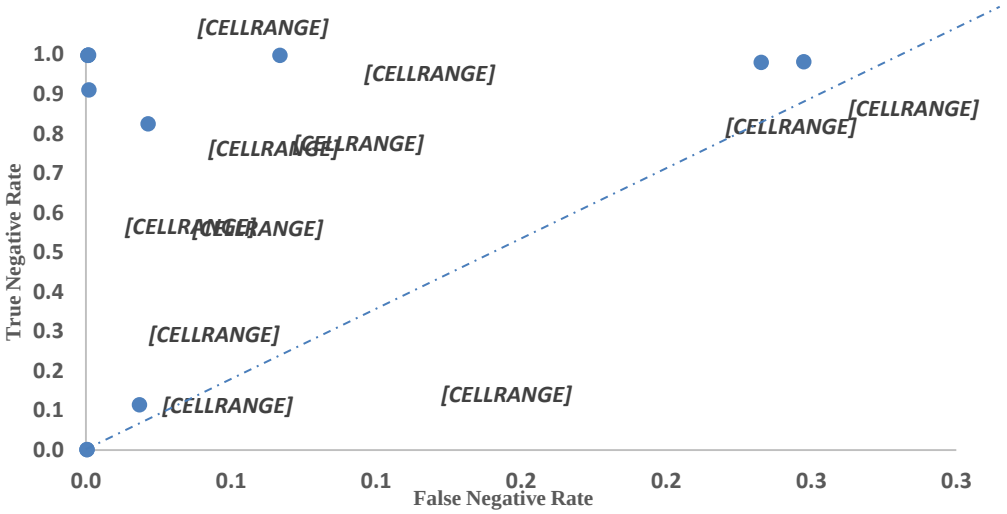

Figure 7: Specificity (TNR) vs. Miss Rate (FNR) for all Features of ESPRT Using Different DDoS Attacks of CICDDoS2019 Dataset based on NETBIOS Dataset

Tables IV and V show a clearer view of the relationship between TPR vs. FPR and TNR vs. FNR as represented in Figures 6 and 7. Some individual features generate high accuracy, as measured by high TPR and low FPR. This is why selecting a group of features is better in the detection system than picking individual features. For example, the individual features including (f18, f22, f40, f75, f76, f77, f78, f79, f80, f81, f82) achieved perfect classification with FPR = 0.000 and TPR = 1.000. On the other hand, individual features like (f5, f9, f55, f70, f74) showed lower performance with FPR = 1.000 and TPR = 1.000, as shown in Table IV. Another example is that individual features like f59 and f60 had higher FPR values (0.887 and 0.175, respectively), indicating lower performance. A similar pattern of high accuracy, as measured by high TNR and low FNR, is shown in Table V. For example, the individual features including (f7, f8, f10, f13, f14, f15, f17, f19, f20, f21, f29, f30, f31, f32, f33, f34, f36) achieved high detection results with FPR = 0.001 and TPR = 0.997. On the other hand, a feature like f24 showed lower performance with FPR = 0.247 and TPR = 0.981, as shown in Table V. As a result, the data suggests that certain combinations of features provide better detection results, while some individual features may contribute more to misclassifications, as shown in Table IV and Table V.

TABLE IV. SENSITIVITY (TPR) VS PROBABILITY OF FALSE ALARM (FPR) FOR ALL FEATURES OF ESPRT USING DIFFERENT DDOS ATTACKS OF CICDDOS2019 DATASET BASED ON NETBIOS DATASET

Feature number	FPR	TPR
f53	0.002	0.975
f7,f8,f10,f13,f14,f15,f17,f19,f20,f21,f29,f30,f31,f32,f33,f34,f36,f37,f38,f39,f41,f42,f43,f8	0.003	0.999
f13	0.002	0.998
f23	0.089	0.999
f59	0.887	0.982
f18,f22,f40,f75,f76,f77,f78,f79,f80,f81,f82	0.000	1.000
f35	0.020	0.767
f60	0.175	0.979
f11,f12,f25,f26,f27,f28,f49,f51,f52,f56,f57,f58,f61,f68,f69,f71,f72,f73	0.003	0.999
f24	0.019	0.753
f5,f9,f55,f70,f74	1.000	1.000

TABLE V. SPECIFICITY (TNR) VS. MISS RATE (FNR) FOR ALL FEATURES OF ESPRT USING DIFFERENT DDOS ATTACKS OF CICDDOS2019 DATASET BASED ON NETBIOS DATASET

Feature number	FNR	TNR
f7,f8,f10,f13,f14,f15,f17,f19,f20,f21,f29,f30,f31,f32,f33,f34,f36	0.001	0.997
f16	0.067	0.998
f23	0.001	0.911
f11,f12,f25,f26,f27,f28,f37,f38,f39,f41,f42,f43,f61	0.001	0.998
f59	0.018	0.113
f60	0.021	0.825
f24	0.247	0.981
f49,f51,f52,f53,f56,f57,f58,f68,f69,f71,f72,f73	0.001	0.998
f70,f74,f75,f76,f77,f78,f79,f80,f81,f82, f5,f9,f18,f22,f40,f55	0.000	0.000

C. Confusion Metrics Based on Set of Selected Features of CICDDoS2019

Some ESPRT's confusion matrix metrics were calculated using the CICDDoS2019 dataset, which contains various types of DDoS attacks. A combination of feature selection techniques was used to select the best features as mentioned earlier, resulting in feature sets of 5, 10, 15, and 20. The calculated metrics included accuracy, F-score, and probability of false alarm (FPR). The accuracy of ESPRT was first evaluated using the 5 best-selected features, resulting in an accuracy rate of 93.2% for NTP. However, the accuracy of the 10, 15, and 20 feature sets was lower, at 92.5, 92.2, and 92.9, respectively.

The accuracy of ESPRT using a set of 5 features for the DNS, MSSQL, SSDP, UDP-Lag, SYN, and SNMP attack datasets was better than for other feature sets, with accuracy rates of 95.9%, 99.6%, 97.5%, 97.4%, 99.6%, and 99.6%, respectively. For LDAP and TFTP attacks, the highest accuracy was achieved using 10 features, while using 15 features for NETBIOS resulted in the highest accuracy rate of 96.7%. Finally, the 20-feature set was found to be the best for UDP attacks, with an accurate rate of 97.3% compared to other feature sets.

VI. CONCLUSION

DDoS attacks refer to Distributed Denial of Service attacks. These attacks pose a significant cybersecurity threat. These attacks are designed to disrupt the access of legitimate users to services provided by a targeted server or controller. A combination of feature selection techniques was used to select the best features. The accuracy of ESPRT using a set of 5 features for the DNS, MSSQL, SSDP, UDP-Lag, SYN, and SNMP attack datasets was better than for other feature sets, with accuracy rates of 95.9%, 99.6%, 97.5%, 97.4%, 99.6%, and 99.6%, respectively. For LDAP and TFTP attacks, the highest accuracy was achieved using 10 features, while using 15 features for NETBIOS resulted in the highest accuracy rate of 96.7%. The 20-feature set was found to be the best for UDP attacks, with an accuracy rate of 97.3% compared to other feature sets. For the future task, another significant target for DDoS attacks is Software-Defined Networks (SDN), which represents a new infrastructure that enables network administrators to program network devices and servers, thereby enhancing network performance. Finally, this infrastructure has also become a goal for DDoS threats, making its implementation a potential focus for future work.

ACKNOWLEDGMENT

This work is supported by the University Putra Malaysia -College of Engineering- Department of Electrical and Electronic Engineering, Faculty of Engineering, and Al-Iraqia University- College of Engineering- Department of Computer and Network Engineering.

REFERENCES

[1] B. H. Ali, "Study the Effectiveness of Sequential Probability Ratio Test in detection DDoS Attacks against SDN, " *Al-Iraqia Journal for Scientific Engineering Research*, vol. 0, no. 0, pp. 35–41, 2022, doi: https://doi.org/10.33193/IJSER.0.00.2021.21.

[2] H. M. Belachew, M. Y. Beyene, A. B. Desta, B. T. Alemu, S. S. Musa and A. J. Muhammed, "Design a Robust DDoS Attack Detection and Mitigation Scheme in SDN-Edge-IoT by Leveraging Machine Learning," in *IEEE Access*, vol. 13, pp. 10194-10214, 2025, doi: 10.1109/ACCESS.2025.3526692.

[3] B.H. Ali, N. B. Sulaiman, S. A. R. Al-Haddad, R. B. Atan, and S. L. Mohd Hassan, "DDoS Detection Using Active and Idle Features of Revised CICFlowMeter and Statistical Approaches," *IEEE conference*, Zakho, Iraq, 2022, doi: https://doi.org/10.1109/ICOASE56293.2022.10075591.

[4] B. H. Ali, N. B. Sulaiman, S. A. R. Al-Haddad, R. B. Atan, S. L. Mohd Hassan, and M. K. Alghrairi, "Identification of distributed denial of services anomalies by using combination of entropy and sequential probabilities ratio test methods," *Sensors*, vol. 21, no. 19, 2021, doi: https://doi.org/10.3390/s21196453.

[5] B. H. Ali, N. B. Sulaiman, S. A. R. Al-Haddad, R. B. Atan, and S. L. Mohd Hassan, "Detection of different Types of Distributed Denial of Service Attacks using Multiple Features of Entropy and Sequential Probabilities Ratio Test," *Journal of Engineering Science and Technology*, vol. 18, no. 2, pp. 844 – 861, 2023.

[6] H. Ding, P. M. Feng, W. Chen, and H. Lin, "Identification of Bacteriophage Virion Proteins by the ANOVA Feature Selection and Analysis," *Mol Biosyst*, vol. 10, no. 8, pp. 2229–2235, 2014, doi: 10.1039/C4MB00316K.

[7] N. Ghalia, M. Nassereddine, and O. Al-Khatib, "Ensemble Learning for Network Intrusion Detection Based on Correlation and Embedded Feature Selection Techniques" *Computers*, vol.14, no. 3, 2025, doi: https://doi.org/10.3390/computers14030082.

[8] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely Randomised Trees," *Mach Learn*, vol. 63, no. 1, pp. 3–42, Apr. 2006, doi: 10.1007/S10994-006-6226-1/METRICS.

[9] S. Jany Shabu et al., "Research on Intrusion Detection Method Based on Pearson Correlation Coefficient Feature Selection Algorithm," *in Journal of Physics: Conference Series*, *Volume. 1757, International Conference on Computer Big Data and Artificial Intelligence (ICCBDAI)*, Changsha, China: IOP Publishing, pp. 1–10, Jan. 2021, doi: 10.1088/1742-6596/1757/1/012054.

[10] M. Nooribakhsh and M. Mollamotalebi, "A Review on Statistical Approaches for Anomaly Detection in DDoS Attacks," *Information Security Journal: A Global Perspective*, vol. 29, no. 3, pp. 118–133, Feb. 2020, doi: 10.1080/19393555.2020.1717019.

[11] H. Kousar, M. M. Mulla, P. Shettar, and D. G. Narayan, "DDoS Attack Detection System using Apache Spark*," in International Conference on Computer Communication and Informatics (ICCCI*), Coimbatore, India: IEEE, Jun. 2021, doi: 10.1109/ICCCI50826.2021.9457012.

[12] S. M. Mousavi and M. St-Hilaire, "Early Detection of DDoS Attacks against SDN Controllers," in 2015 International Conference on Computing, Networking and Communications, ICNC, Garden Grove, CA, USA: IEEE, Mar. 2015, pp. 77–81, doi: 10.1109/ICCNC.2015.7069319.

[13] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "A Novel Measure for Low-Rate and High-Rate DDoS Attack Detection using Multivariate Data Analysis," *in 8th International Conference on Communication Systems and Networks (COMSNETS)*, Bangalore, India: IEEE, Mar. 2016, doi: 10.1109/COMSNETS.2016.7439939.

[14] I. Özçelik and R. R. Brooks, "Cusum - Entropy: An Efficient Method for DDoS Attack Detection," *in 4th International Istanbul Smart Grid Congress and Fair (ICSG)*, Istanbul, Turkey: IEEE, Jun. 2016, doi: 10.1109/SGCF.2016.7492429.

[15] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy," *in International Carnahan Conference on Security Technology*, Chennai, India: IEEE, Oct. 2019, pp. 1–8, doi: 10.1109/CCST.2019.8888419.

[16] V. Gaur and R. Kumar, "Analysis of Machine Learning Classifiers for Early Detection of DDoS Attacks on IoT Devices," *Arab J Sci Eng*, vol. 47, no. 2, pp. 1353–1374, Feb. 2022, doi: 10.1007/S13369-021-05947-3/METRICS.

[17] Farhan, M., Waheed ud din, H., Ullah, S. *et al.* Network-based intrusion detection using deep learning technique. *Sci Rep* 15, 25550 (2025). https://doi.org/10.1038/s41598-025-08770-0