I. J. Computer Network and Information Security, **2022, 3, 33-46**
Published Online June 2022 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2022.03.03

DDoS Attacks Detection in the Application Layer Using Three Level Machine Learning Classification Architecture

Bassam M. Kanber
School of Computer Science and Technology, Xidian University
E-mail: aflatoon353@gmail.com

Naglaa F. Noaman
School of Artificial Intelligence, Xidian University
E-mail: noga2012fa@gmail.com

Amr M. H. Saeed
School of Computer Science and Technology Engineering, Northwestern Polytechnical University
E-mail: amr.alkamel.m1@gmail.com

Mansoor Malas
School of Communication, Xidian University
E-mail: mansoormalas5@gmail.com

Received: 02 January 2022; Accepted: 07 April 2022; Published: 08 June 2022

Abstract: Distributed Denial of Service (DDoS) is an ever-changing type of attack in cybersecurity, especially with the growing demand for cloud and web services raising a never-ending challenge in the lucrative business. DDoS attacks disrupt users' access to the targeted online services leading to significant business loss. This article presents a three-level architecture for detecting DDoS attacks at the application layer. The first level is responsible for selecting the best features of the samples and classifying the traffic into either benign or malicious, then the second level consists of a hard voting classifier to identify the type of the DDoS source: UDP, TCP, or Mixed-based. Finally, the last level aligns the attack to the appropriate DDoS type. This approach is validated using the CIC-DDoS2019 dataset, and the time, accuracy score, and precision are used as the model performance metrics. Compared to the existing machine learning (ML) approaches, the proposed architecture reveals substantial improvements in both binary and multiclass classification of application-layer DDoS attacks.

Index Terms: CatBoost, CICDDoS2019, DDoS, LGBM, XGBoost.

1. Introduction

Globally, in different fields, the number of Internet devices continues to rise and is approaching new peaks, powered in particular by the popularization of ubiquitous computing, materialized by the concept of the Internet of Things (IoT), and defined by the notion of connecting anything, anywhere, anytime[1]. Various industries hope to share resources on the network, and the increase in internet devices tends to correspond with an increase in cyber-crimes, which represents a threat to the information security of the organization. The importance of maintaining network security is axiomatic. In general, DDoS attacks present a very serious risk to network security.

China's Internet is in a condition of rapid development, according to the "Report on the Development and Security of Internet Sites in China (2018)," which is causing severe concerns [2]. Web page fraud, tampering, vulnerabilities, and denial-of-service attacks have all increased in recent years.

According to Cloudflare's study, ransom DDoS assaults climbed by about a third between 2020 and 2021 and by 75% in Q4 2021 [3]. Moreover, the manufacturing industry had the greatest application-layer DDoS attacks, up 641% quarter over quarter. Attackers tried to block individuals from accessing online resources of various commercial and non-profit organizations, according to Yandex and Qrator Labs [4]. Attackers are adopting increasingly complex strategies, such as multi-vector assaults, in their DDoS attacks. Generally, the majority of DDoS attackers use multi-vector DDoS attacks, which combine a variety of DDoS attacks into one [5].

As recent surveys on network applications, wireless networks, cloud computing, and big data illustrate, DDoS attacks are a particular form of network interference that has attracted academia's attention [1, 6-8]. The attack on seven South Korean banks to extort money instead of returning their function to normalcy is a recent example of cybercrime's evolution. On September 6 and 7, 2019, In Germany and some parts of Europe, Wikipedia was shot down by a DDoS attack [9]. DDoS attacks became more widespread during the following few years, and Cisco expects that the overall number of DDoS attacks would double from 7.9 million in 2018 to over 15 million by 2023. Given that IT service downtime costs firms anywhere from $300,000 to over $1,000,000 per hour [10, 11]. You can see how even a minor DDoS attack might have a significant financial impact. Various network layers, such as physical, transport, and application layers, can be submerged by DDoS attacks. Since one layer has a single failure point, if it is submerged by some DDoS attack, the entire network will go down simultaneously. The application Layer attack is more complicated and targets individual apps or utilities and exhausts network resources slowly [12, 13]. Computer networks are currently undergoing a large-scale and complex evolution. DDoS attacks have continued to develop and grow over the years, with the presence of botnets, and the damage has gotten increasingly significant. DDoS attacks are primarily directed against network bandwidth and network resources, and they are easy to set up, direct, and launch. The least missing resource for an attacker in today's general environment is network resources. As a result, as long as devices can connect to the internet, they can be used as malicious actors. With the advancement of defense functions and attacker technology, new DDoS attack tactics appear one after the other. To successfully drain network bandwidth, the attacker just needs to transmit a high quantity of network data packets to the server at first. Network layer protocols such as UDP and ICMP are utilized for this. Networks and servers have improved their ability to detect DDoS attacks at the network layer over time. Servers have the ability to give better and more bandwidth. Despite this, large queries will use network capacity and cause server unavailability.

However, when it comes to attacks, it appears that launching an attack has grown more difficult. As a result, the attacker shifts his focus to the transport layer, but the server begins to defend itself against these attacks over time. The patterns of these attacks can be discovered and properly distinguished across mediums. As network layer and transport layer defenses become stronger, attackers are moving to the application layer, resulting in a new wave of application-layer DDoS attacks.

This work provides a machine-learning-based detection approach for DDoS attacks and conducts experimental verification and analysis. This paper's contribution contains the following:

1. An application-layer DDoS attack detection approach based on Extreme Gradient Boosting (XGBoost) and Light Gradient Boosted Machine (LGBM) XGBoost-LGBM is proposed. This approach can reduce the dimensionality and redundancy of traffic attribute features, resulting in increased detection efficiency and effectiveness. The following findings are derived after experimental verification: In terms of accuracy, precision, and model training duration, the application layer DDoS attack detection approach based on XGBoost-LGBM outperforms techniques based on traditional machine learning (ML) algorithms.
2. A three-level classification framework based on ML is presented for DDoS detection in the application layer, which improves on the drawbacks of existing ML approaches, and to identify the type of DDoS attack and choose the right plan of action. At the most basic level, our approach classifies incoming network traffic as either regular or DDoS attacks. A multiclass algorithm classifies DDoS attacks as TCP, UDP, or Mix-based DDoS attacks at the second level. In the third level, multiclass algorithms characterize the anomaly discovered in the second phase to identify the type of attack and choose the right plan of action.
3. We applied many ML algorithms that are well known to observe the differences and record the variations for detecting DDoS attack on the application-layer and test them in terms of time, accuracy and precision.
4. The proposed architecture differs from others by identifying of a variety of DDoS application-layer attacks, such as NetBIOS, LDAP, MSSQL, Syn, UDP, UDP-lag, DNS, SNMP, SSDP, NTP, Portmap, and TFTP.

The remaining portion of this paper is structured as follows: In this research area, Section 2 provides the history and related work. The research methodology is listed in section 3. Section 4 illustrates in detail the results. Section 5 contains the research discussion. Finally, the paper is concluded by Section 6 and proposes some future work.

2. Related Work

In network security research, application-layer DDoS attack detection is still a hot research area [1, 6]. This section covers the recent research on the subject.

Kumar, A et al. [14] examined the application of the MeanShift algorithm to identify an attack on a network using the KDD 99 dataset resulting in two clusters. Clustering and performance evaluation are the two processes that make up the experiment, where the detection rate of the proposed method was limited to 81%.

Aamir, M et al. [11] used unlabeled traffic from the CICIDS2017 dataset with two clustering algorithms: agglomerative clustering to watch the process and identify the maximum number of clusters, and PCA in K-means for feature extraction, with a voting mechanism for labeling in the middle. After labeling, k-Nearest Neighbors (KNN),

Support Vector Machine (SVM), and Random Forest (RF) were used for creating trained models for future classification. This study achieved a good accuracy but unstable precision.

A machine-learning matrix with a bio-inspired bat algorithm was proposed by Sreeram, I., and Vuppala, V. P. K [15] to allow HTTP DDoS attack detection. Instead of using user sessions, the researchers used time intervals and packet patterns to create a detection system. The time interval employs a machine-learning matrix by setting a value of maximum sessions for one-time intervals and computing the number of sessions in the one-time interval. To track user behavior, the frequency with which users access a web page and the time difference between the first and the second-page requests are calculated. The tests were carried out on the CAIDA dataset against ARTP and FCAAIS. Behal, S. et al. [16] proposed D-FACE, an anomaly-based distributed approach for early detection of DDoS attacks and Flash Events at the ISP level using MIT Lincoln, FIFA98, DDoSTB, and CAIDA datasets without using sampling techniques. A detection metric is calculated and delivered to a central coordinator on the victim's network's premises at each of ISP's entry points.

Hoque, N. et al. [17] designed a FPGA hardware for real-time DDoS detection at the victim's end application layer. The suggested solution utilized to detect legitimate traffic from fake traffic. CAIDA DDoS 2007, MIT DARPA, and TUIDS datasets were used to test the technique. They analyzed source IPs, source IPs index variation, and packet rate to detect the attack. The FPGA implementation used in this study obtained a high-speed rate.

Hameed, S. and Ali, U. [18] described A framework named HADEC for detecting live high-rate DDoS attacks at network and application layers, such as TCP-SYN, HTTP GET, UDP, and ICMP. The detection server and the capture server are the two primary components of the framework. The server responsible for capturing live network traffic and transferring the log to the detection server for processing is the first step. If the source connection exceeds the set threshold, the detection calculates incoming packets for UDP, ICMP, and HTTP. On the cluster nodes, the detection server runs a Hadoop cluster and starts MapReduce-based DDoS detection operations. Financial institutions, small and medium businesses, can benefit from the proposed detection because it is low-cost. They used MapReduce to create a counter-based DDoS detection system for four different types of flooding attacks.

Jazi, H. H. et al. [19] suggested a Hypertext Transfer Protocol HTTP-based technique for detecting and preventing flood attacks on web servers via sampling technique. The authors used a nonparametric CUSUM algorithm with real traffic gathered from a web server on an academic network and prepared with a combination of application layer DoS attacks and actual traffic to determine whether the traffic observed was benign or a DDoS attack, focusing on two features: the number of application-layer requests and the number of packets with a payload size of zero. This technique was previously utilized in [20] to identify large DDoS attacks at the network layer.

Singh, K. J. et al. [21] employed a multi-layer perceptron paired with a genetic algorithm (MLP.GA) to identify DDoS attacks at the application-layer. The study examined the characteristics of incoming packets leveraging four parameters to generate detections. The time interval for normal users is set. For the reason those real users look for and consume content on web pages before moving on to other web pages, the system predicts that the web server receives within 20 seconds, the number of HTTP GET requests received, the number of IPs that target the server and the DDoS port number. According to the literature, DDoS attackers use a fixed protocol length. To test the system, they used EPA-HTTP DDoS, CAIDA 2007, and an experimentally DDoS data set.

When recognizing attack traffic, actual traffic, and Flash Crowd traffic, Singh, K. et al. [22] developed a method to detect HTTP DDoS attacks using ML algorithms and discriminate between botnets and legitimate users. By recognizing the source of the botnet and monitoring user behavior to detect malicious requests to the Web server, the system is deployed as a proxy and solely checks the user's activity. This study presented four unique behavioral characteristics for distinguishing GET flood attack sources from actual and flash traffic. To produce the traffic logs, the researchers employ a variety of attack techniques, offering a comprehensive solution for detecting 12 distinct GET flood attack methods.

Liu, Z. et al. [23] designed a three tiered defense architecture to fight against a wide spectrum of DDoS attacks. According to authors, the proposed system is capable of dealing with large-scale attacks. The system is composed of three layers: The flood throttling layer prevents DDoS attacks based on amplification. The congestion resolving layer avoids complex attacks that are difficult to screen. The user-specific layer enables DDoS victims to impose traffic policing rules that best meet their business needs. This approach is widely used in business, although it has proven ineffectual against truly massive DDoS attacks.

Wang, C. et al. [24] proposed the SkyShield system to detect and prevent DDoS attacks on the application layer. In the detection phase, SkyShield utilizes the difference between two hash tables to identify intrusions caused by attacker hosts. In an active attack, the anomalous drawing is used to help in the discovery of hostile hosts. Filtering, whitelisting, blacklisting, and CAPTCHA are used as protection techniques in the mitigation phase. Custom datasets were gathered from a large internet group. Because SkyShield concentrated on the application layer, particularly the HTTP protocol, the proposed system is vulnerable to floods at the network and transport layers.

Sokolov, M. and Herndon, N. [25] proposed an Ensemble of Light Gradient Boosted Machines to anticipate malware attacks. The authors performed three Tests. Test A is the control experiment, they employed the entire dataset. Test B, authors used LGBM feature extraction with a threshold of 0.5% and reduced the number of features from 114 to 84 features. Test C, authors used Random Forest feature extraction with a threshold of 0.5% and reduced the number of

features from 114 to 41 features. They showed that the suggested technique outperformed Automated Artificial Intelligence using a dataset given by Microsoft (Microsoft Kaggle's Malware Prediction dataset). The architecture proposed is intended to identify malware with a good precision and less processing power.

Elsayed, M. S. et al. [12] proposed DDoSNet, an intrusion detection system for DDoS attacks in SDN environments using Recurrent Neural Network (RNN) and an autoencoder for feature extraction and softmax regression at the output layer using null routing method. The authors used the CICDDoS2019 to test the model for binary classification on DDoS. Rai, M. and Mandoria, H. L. [26] used three kinds of classifiers: Linear, Gradient Boosting Decision Tree, and Deep Neural Network (DNN) classifiers to identify network intrusion and train a layered model consisting of all these classifiers, then they were put to the test with the NSL KDD dataset. There are 41 features in the dataset, including fundamental features, traffic features, and content features, as well as 21 attack categories. According to the authors' studies, Gradient Boosting Decision Tree ensembles LGBM, XG-Boost, and the layered model outperformed linear models and deep neural networks.

A lightweight Version Number Attacks VNA detection model called ML-LGBM is suggested by Osman, M. et al. [27]. The construction of a large VNA dataset, a feature extraction technique, an LGBM algorithm, and the highest parameter optimization are all part of the ML-LGBM model's effort. Extensive tests show that the ML-LGBM model has numerous benefits based on metrics, including accuracy, precision, F1-score, true negative rate, and false-positive rate. Furthermore, the ML-LGBM model has a shorter execution time and requires less memory, making it appropriate for IoT devices with limited resources. In the literature [28], Bakhareva, N. et al. suggested “Attack detection in enterprise networks using ML methods,” which employs CatBoost tree, Logistic Regression, Linear SVC, and LGBM. They used the CICIDS2017 dataset for binary and multiclass classification. The authors used the "Flow Bytes/s" or "Flow Packets/s" columns and rounding values to five digits after the floating-point. The GPU was used for CatBoost and LGBM, while the CPU for the remainder. According to the authors, CatBoost beats others in terms of cross-validation accuracy, F1 score, precision, recall, and AUC, but it comes at a cost in terms of execution times. As a consequence, new possibilities to apply CatBoost to other areas where Gradient Boosted Decision Trees (GBDTs) are beneficial in resolving cyber-security problems may emerge.

Table 1. Related Work Analysis and Comparison

Reference	Algorithm	Dataset	Performance	Strengths	Limitation	Year
[11]	Multiple Clustering (AC, Kmeans over PCA), k-NN, RF, SVM	CICIDS2017	(RF) Accuracy =0.9666, Precision = 0.88	Multiple classification and clustering, Voting method for final labeling	Low number of features, High Time complexity	2019
[12]	Recurrent Neural Network (RNN) with autoencoder	CICDDoS2019	Accuracy=0.99 Precision =0.99, F1-Score = 0.99	Good performance in binary, using the null routing method	Used only for binary classification, Complex and low speed DL model	2020
[14]	Mean Shift	KDD99	Accuracy=0.812, Detection rate = 0.791	Two-Fold, Unsupervised Learning based	Low detection rate, applied on old dataset, Meanshift usually merges overlapping clusters which impacts the performance.	2020
[15]	bat algorithm	CAIDA 2007	Accuracy = 0.91	Use HTTP flood.	Bat algorithm has no mathematical analysis so converges very quickly at the early stage and then convergence rate become very slow.	2019
[27]	LGBM	VNA dataset	Accuracy = 0.99, Precision = 0.99	Routing protocols of the IoT, develop Version Number Attacks dataset, less memory resource	Slower execution time, only on RPL-based network	2021
[28]	LightGBM, CatBoost, LinearSVC Logistic Regression	CICIDS2017	Accuracy=0.99 Precision =0.99, F1-Score = 0.99	binary & multi-class classification. Good performance.	High training time especially with multiclassification, small numbers of DDoS attacks, Need to be generalized with more sophisticated datasets.	2019
[29]	CatBoost	Kaggle open dataset	Accuracy=0.98 Recall =0.82, F1-Score = 0.86	Using flow rules to mitigate attacks.	Limited number of attributes. High training time.	2021

In [29], Sanjeetha, R. et al. suggested detection and mitigation of botnets in an SDN environment using ML. They used Mininet to simulate the Botnet and RYU controller and then used Python to create an auxiliary component of the CatBoost model that interacted with the controller through REST API. When discovering a DDoS attack in the network, the hosts responsible for the attack are recognized, then they add flow rules into switches to mitigate the attack. They made use of a Kaggle open dataset, including around 2.1 million items. Dataset is trained on several models to discover the least amount of training time. The author demonstrates that the proposed approach is 98% accurate.

The whole related work performance, strengths and limitations are analyzed and summarized in Table 1 To this extent, researchers created mathematical models and detection algorithms to identify DDoS attacks through ML, genetic algorithms, biological heuristic algorithms, feature extraction, and other theories, and produced specific detection findings. However, the following weaknesses in the detection of application-layer DDoS attacks remain:

1. The accuracy of attack detection must be improved, and the number of false alarms generated during the detection process must be reduced.
2. The majority of research focuses on detecting high-rate DDoS attacks. Only a tiny proportion of researchers are concerned about low-rate DDoS. Only a few research have presented strategies for detecting two types of DDoS attacks: high-speed and low-speed DDoS attacks. The detection range should be increased as well.
3. Parts of the data sets now in use are out of date and cannot be meaningfully compared or evaluated.

3. Methodology

3.1. CICDDoS2019 Dataset

The lack of datasets is one of the most rigorous problems for Machine Learning (ML) intrusion detection strategies. Privacy and illegal problems are the primary explanation for the lack of datasets in the intrusion detection domain. Network traffic holds very confidential information, where users and company secrets can be exposed by the availability of such information or even personal contact.

From the Canadian Institute of Cybersecurity, we chose CICDDoS2019 Dataset [30]. Since 1998, this Institute has given different benchmark datasets to analyze various network security solutions. The dataset includes 12 separated DDoS attacks that can be carried out in the application layer via transport layer protocols such as Transmission control protocol (TCP) and User datagram protocols (UDP). In terms of exploitation-based and reflection-based attacks, the taxonomy of attacks is carried out. The dataset was collected within two separate days. However, each attack type was in a separate PCAP and CSV format. The dataset comprises 87 flow characteristics and has been extracted using CICFlowMeter tools [31, 32]. The major notable feature of this dataset is that it's contemporary and represents the latest attacks and benign traffic as well.

3.2. Proposed Architecture

The intrusion detection model in the suggested methodology is based on three primary phases for the classification and identification of incoming network data, as illustrated in Fig.1. Transport layer protocols, TCP, UDP, or a mix of the two, can be used to carry out these attacks through the application layer. This Section describes the system architecture, as well as data pretreatment procedures and the classification algorithm.

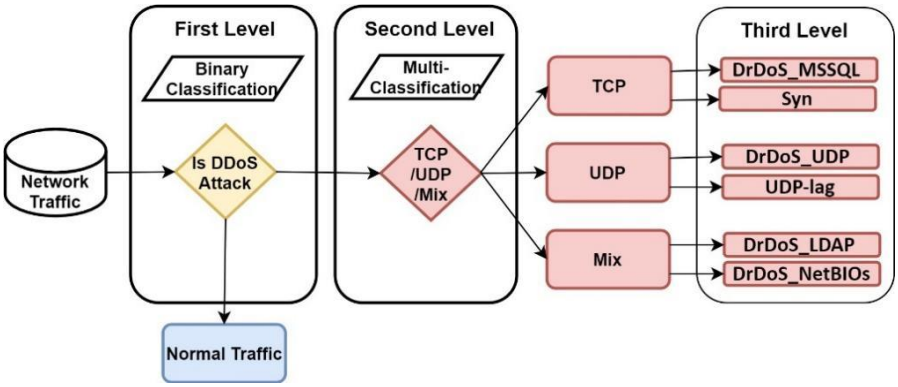

Fig.1. Overall Structure of the Model

A. Dataset Preprocessing

The following stage is standardizing the data set and making it standardized. First, removing socket characteristics: We deleted all socket characteristics such as source and destination IP, source and destination port, timestamp, and flow ID. These characteristics differ from one network to the next, so we'll have to train the model with packet characteristics.

So, we were able to acquire a range of 80 features for the model input in CIC-DDoS2019. Then, data cleaning: There are a lot of missing (nan) and infinity values in the original data, all of these values were extracted from the data. After that, the min-max normalization allows for more flexibility in classifier construction. The benefit of the normalization approach is that it properly preserves all of the data's associations, thus it won't create any divergence. As a result, when employing the min-max normalization approach, each feature is inside the classifier's suitable range, and the associated features' fundamental distribution remains unaffected. Discrete and continuous value characteristics are frequently seen in attack data sets. The range of feature values will be varied when discrete and continuous values are mixed. This solves the problem by fitting the computation with the minimum-maximum normalization approach, allowing the model to be properly trained. Many ML algorithm training procedures only accept particular types of input and normalize all characteristics using min-max normalization. The following is the conversion formula:

$$x' = \frac{x - \min}{\max - \min} \tag{1}$$

Adding two numerical columns: we trained our model for binary classification and multi-class classification. Therefore, apart from usual traffic, we consider all DDoS groups as attack categories. So, in a single column, we encode the string value for the normal and attack mark to a binary value of 0 and 1, respectively. We also encode the string value for the attack form to numbers from 1 to 3 as TCP, UDP, or Mixed based attacks.

B. First Level

The XGBoost-LGBM paradigm for application-layer DDoS attacks in the first level as depicted in Fig.2. The suggested approach contains two portions after data preprocessing in the early stages of creating the CatBoost classifier model: Feature selection and training and testing classifier. The feature selection portion employs a selection method based on the feature significance score generated by XGBoost and LGBM. The second-portion processed data is used to train and test the CatBoost classifier (CBC). We provide a technique for selecting features (Feature Selection Based on LGBM and Feature Selection Based on XGBoost) FSBLGXG to create feature rankings based on XGBoost and LGBM algorithms that can rapidly identify the optimal feature combination by generating a new list of features using union selection from both. Algorithm in Algorithm 1 contains an explanation of the selection procedure. This technique employs two algorithms to identify important feature combinations and uses classification accuracy as an assessment measure. First, remove the features in the feature set to be picked in order of importance from lower to higher for each separate algorithm. Then, using union selection from both algorithms, an empty target feature set is created from the best features of both methods. The feature with the highest priority is chosen from the LGBM and XGBoost feature sets to be added to the empty feature set.

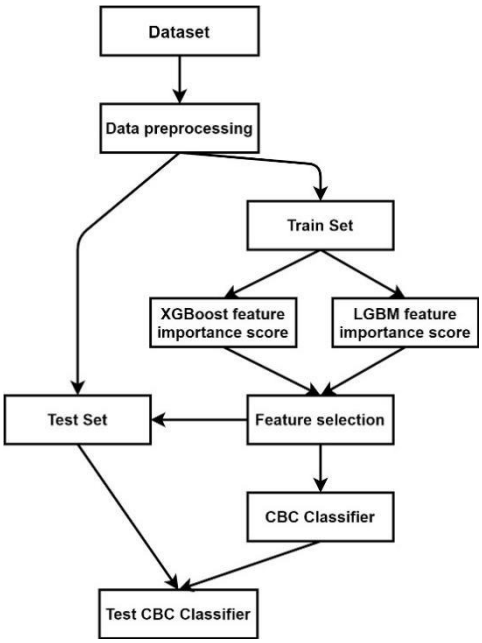

Fig.2. Model Diagram based on XGBoost-LGBM (First Level)

Algorithm 1 FSBLGXG(S)

Input: S: a data set containing all features;
Cla1: The first classifier
Cla2: The second classifier
Acc1: The accuracy of the first classifier
Acc2: The accuracy of the second classifier
n: the dimension of the feature
List1: list of important features in classifier one
List2: list of important features in classifier
Output: Frequent item sets

Begin:

Initialization: S ={F}$O = \phi$
for each $\{f_i\} \epsilon \{F\}$ **do**
for i = 1,⋯, n **do**
Calculate Cla1.feature_importance(fi)
Add(f_i, List 1)
end for
Sort (List1)
for each $\{f_k\} \epsilon \{F\}$ **do**
for i = 1,⋯, n **do**
Calculate Cla2.feature_importance(f_k)
Add (f_k, List2)
end for
Sort (List2)
for each $\{f_i\} \in \{List1\}$ **do**
for i = 1,⋯, n **do**
Remove (f_i)
Calculate Acc1 (List1)
List1.GetBestFeaturesOfBestAcc (Cal1)
end for
Sort (List1)
for each $\{f_i\} \in \{List2\}$ **do**
for i = 1,⋯, n **do**
Remove (f_i)
Calculate Acc2 (List2)
List2.GetBestFeaturesOfBestAcc (Cal2)
end for
Sort (List2)
O = List 1+ List2
Call the CBC classifier on the data set containing the target feature O
End

C. Second Level

After detecting DDoS attack in the first level, then in the second level, the DDoS attack is classified into three subsets: TCP, UDP, and Mix, by using a voting classifier consisting of two algorithms, XGBoost and LGBM, the structure of the second level is demonstrated in Fig.3.

D. Third Level

In the third level, pre-trained classifiers will classify the TCP, UDP, or Mixed based DDoS attack traffic into a specific type whenever an attack was identified by the first level and classified as a DDoS attack and choose the right plan of action for that type. According to [30], Table 2. represents a list of the best-selected features and related weights for each attack type.

At this level, the most influential features must be determined by accuracy and speed. We picked two features to discriminate between MSSQL and Syn attacks, and 2 features to discriminate between UDP and UDP-lag attacks, and two features to discriminate between LDAP and NetBIOS attacks.

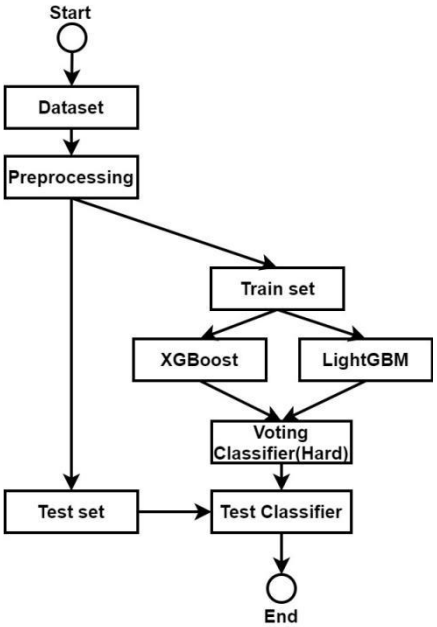

Fig.3. Model Diagram based on XGBoost-LGBM (Second Level)

Table 2. Best-selected Features & Weights for Each Attack Type

Name	Feature	Importance
UDP	Destination Port	0.000699
	Fwd Packet Length Std	0.000615
	Packet Length Std	0.000239
	min_seg_size_forward	9.80E-05
	Protocol	4.50E-05
UDP-lag	ACK Flag Count	0.125438
	Init_Win_bytes_forward	0.002093
	min_seg_size_forward	0.000795
	Fwd IAT Mean	0.000612
	Fwd IAT Max	0.000471
MSSQL	Fwd Packets/s	0.000204
	Protocol	4.60E-05
Syn	ACK Flag Count	0.145834
	Init_Win_bytes_forward	0.002432
	min_seg_size_forward	0.000872
	Fwd IAT Total	0.000571
	Flow Duration	0.000409
LDAP	Max Packet Length	1.278323
	Fwd Packet Length Max	0.143219
	Fwd Packet Length Min	0.008736
	Average Packet Size	0.006532
	Min Packet Length	0.003909
NetBIOS	Fwd Packets/s	0.000172
	min seg size forward	7.20E-05
	Protocol	4.60E-05
	Fwd Header Length	3.50E-05
	Fwd Header Length.1	3.20E-05

3.3. Evaluation Metrics

Metrics of classification algorithms: False Positive (FP): This is when an alarm is triggered by an intrusion detection device even though no attack has occurred in fact. False Negative (FN): occurs when an intrusion detection system fails to detect an actual attack. True Positive (TP): is known as the actual attack that an intrusion detection system detects. True Negative (TN): is a situation where the intrusion detection system does not cause an attack or raise

an alarm. Accuracy is defined as the sum of TP and TN divided by the sum of TP, TN, FP, and FN. Equation (2). Precision and Recall. Equation (3) and (4). F1-score (F1) is a measure that combines precision and recall is the harmonic mean of precision and recall. F1Score comprehensively considers the precision rate and the recall rate, and the value is often closer to the smaller of the two. Equation (5). ROC curve and AUC: A receiver operating characteristic curve (ROC curve) is a graph that shows how well a classification model performs across all categorization levels. Two parameters are shown on this curve: True Positive Rate and False Positive Rate. AUC is the abbreviation for "Area under the ROC Curve." is a measurement of the complete two-dimensional area under the entire ROC curve.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

$$F1 = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

4. Results

The proposed three-level architecture is examined multiple times using the CICDDoS2019 benchmark dataset to assess its performance. To implement the designed technique and perform experiments, we used Python 3.6, Pycharm editor, Windows 10 x64, Intel Core i7-8750H 2.20GHz, NVIDIA GTX 1060 GPUs, and 16GB of RAM.

Before performing the feature selection process, the features with the highest feature importance score are calculated using the XGBoost and LGBM methods as depicted in Fig.4. In XGBoost “Total Length of Bwd Packets” feature has the highest importance among others, while “Ideal Max” feature is the least once. However, the features with the features importance score estimated by the LGBM technique, are shown in Fig.5. where the maximum importance feature is “Init_Win_bytes_forward”, and the lowest importance is “Subflow Bwd Bytes”.

The whole population features in the dataset are 87. The first level architecture, which consist of a combination of both XGBoost and LGBM algorithms, extracted 24 features, 4 features were removed because of duplication resulting in 20 distinct features out of 87 features. The final selected features are shown in Table 3. Where XGBoost features are from 1 to 10, and LGBM features are ranging from 11 to 24. The selected features are then fed into the first level classifier (CatBoost classifier). To evaluate the effectiveness of the selected features, we compared these features with the 20 middle and 20 bottom features as shown in Table 4.

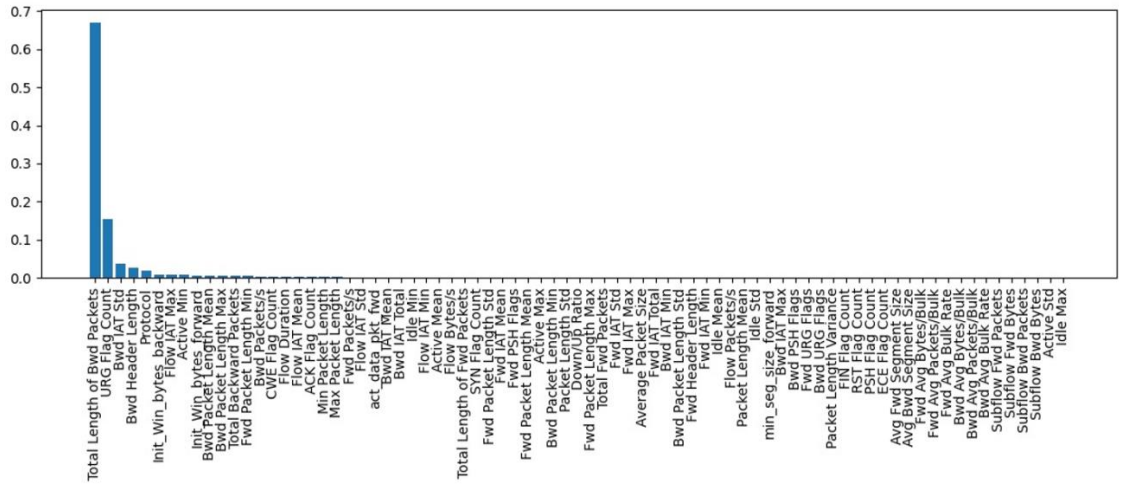

Fig.4. Feature importance score by XGBoost.

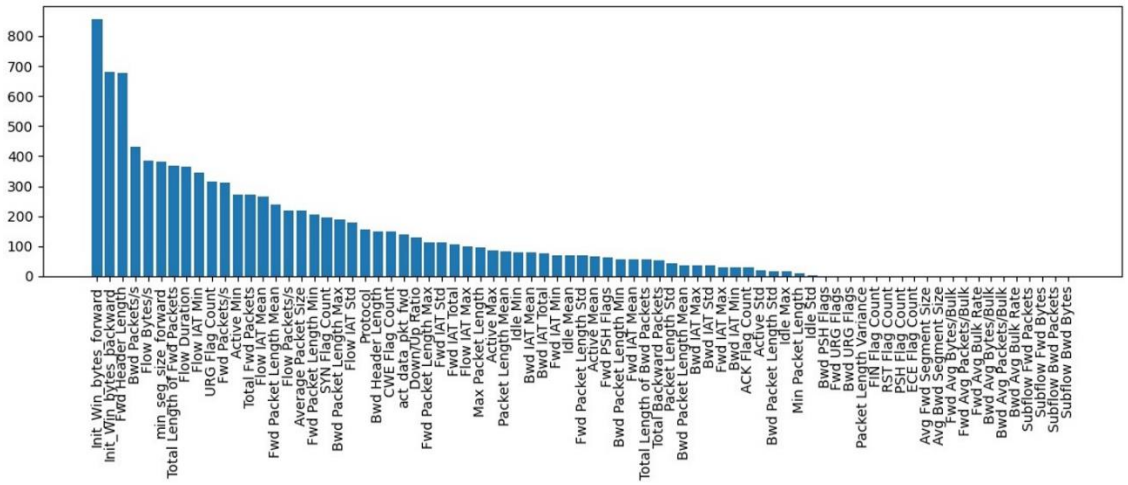

Fig.5. Feature importance score by LGBM.

Table 3. Top selected features by FSBLGXG.

No	Feature Name	No	Feature Name
1	Total Length of Bwd Packets	13	Fwd Header Length
2	URG Flag Count	14	Bwd Packets/s
3	Bwd IAT Std	15	Flow Bytes/s
4	Bwd Header Length	16	min_seg_size_forward
5	Protocol	17	Total Length of Fwd Packets
6	Init_Win_bytes_backward	18	Flow Duration
7	Flow IAT Max	19	Flow IAT Min
8	Active Min	20	URG Flag Count
9	Init_Win_bytes_forward	21	Fwd Packets/s
10	Bwd Packet Length Mean	22	Active Min
11	Init_Win_bytes_forward	23	Total Fwd Packets
12	Init_Win_bytes_backward	24	Flow IAT Mean

Table 4. Feature Selection Comparative Results.

	Top 20	Middle 20	Bottom 20
Acc	**0.9997**	0.9933	0.9330
Pre	**0.97**	0.81	0.55
Time(S)	**9.322350**	10.805213	10.506793

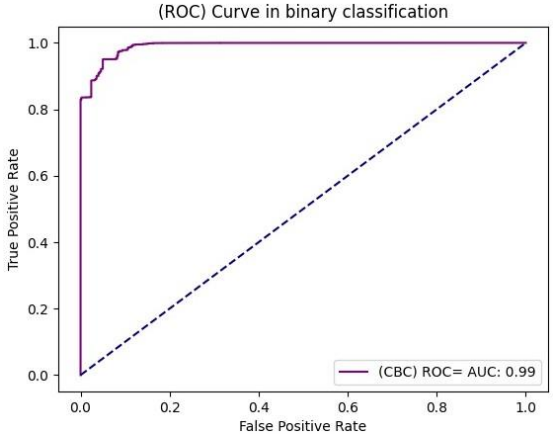

Fig.6. ROC Curve.

In the first level, the incoming traffic is firstly classified as benign or DDoS. The speed factor is taken into account to avert normal traffic delays and early isolating DDoS; thus, all types of DDoS attacks are combined into one category versus the normal traffic. Then performed a quick binary classification using the CatBoost classifier as presented in

Table 5., which illustrates the proposed method's performance against some well-known machine learning algorithms. The area under the curve (AUC) metric thoroughly evaluates the classification model performance, which proved the performance of our model as shown in Fig.6.

If an attack is detected in the first classifier, a multiclass classifier is applied to identify the DDoS source type as TCP, UDP, or Mix-based in the second level using a voting classifier to take advantage of the high accuracy obtained from the LGBM, as shown in Table 5. The UDP subset is reported to have the lowest accuracy as described in Table 6.

Table 5. Level 1 and Level 2 Comparative Results.

	Level 1			Level 2	
	Acc	**Pre**	**Time(s)**	**Acc**	**Pre**
Proposed Model	**0.9979**	**0.97**	9.322350	**0.876**	**0.87**
XGB	0.9656	0.60	30.771183	0.7695	0.77
LGBM	0.9604	0.59	22.675699	0.7632	0.81
RF	0.8928	0.54	77.265341	0.6085	0.55
NB	0.8936	0.54	**2.78547**	0.4831	0.61
CBC	0.9970	0.93	13.0335022	0.7448	0.83
MLP	0.9974	0.83	303.344402	0.29	0.53

Note: XGB = XGBoost, LGBM = Light Gradient Boosting Machine, RF = Random Forest, NB = Naïve Bayes, CBC = CatBoost, MLP = Multi-layer Perceptron.

Table 6. Proposed Model Level 2 detail.

	Precision	**Recall**	**F1-score**
TCP	0.90	0.83	0.86
UDP	0.85	0.77	0.81
Mixed	0.87	0.98	0.91

To ensure the removal of any irrelevant characteristics, feature selection is applied not only in the first level but also in the third level where the third level feature selection is held manually according the feature importance provided by the literature [30] as shown in Table 2. We picked two features to discriminate between MSSQL and Syn attacks, Fwd Packets/s and ACK Flag Count, following numerous trials with TCP attacks. Following extensive tests with UDP DDoS attacks, we chose two characteristics to distinguish between UDP and UDP-lag attacks: ACK Flag Count and Fwd Packet Length Std. In term of TCP/UDP Mixed-based DDoS attacks, we also picked two features to differentiate the LDAP and NetBIOS attacks: Max Packet Length and Fwd Packets/s as depicted in Table 7., Table 8. and Table 9. respectively. Because of having a huge dataset with relatively small independent features, our experiments confirmed that Naive Bayes algorithm provides the best accuracy as 99.94%, 85%, 99.54% in TCP, UDP and Mixed respectively, and less time as 0.1645s, 0.1562s, 0.1515s respectively. Thus, Naïve Bayes (NB) was selected as the final classifier in our architecture.

Table 7. Level Three TCP-based Attacks (MSSQL and Syn) Result.

	All Features			2 Features		
Algorithm	**Acc**	**Pre**	**Time(s)**	**Acc**	**Pre**	**Time(s)**
CBC	0.5534	0.77	9.331287	0.9994	1.00	7.988258
XGB	0.8468	0.88	3.304144	0.9993	1.00	1.329422
LGBM	0.5840	0.77	2.649924	0.9993	1.00	0.411286
RF	0.9997	1.00	4.227696	0.9994	1.00	2.645937
NB	0.6889	0.81	0.350103	**0.9994**	1.00	**0.164573**

Table 8. Level Three UDP-based Attacks (UDP and UDP-lag) Result.

	All Features			2 Features		
Algorithm	**Acc**	**Pre**	**Time(s)**	**Acc**	**Pre**	**Time(s)**
CBC	0.3784	0.53	9.324748	0.2649	0.58	7.471531
XGB	0.2781	0.57	6.347238	0.2670	0.59	1.419625
LGBM	0.3305	0.57	1.022268	0.5341	0.62	0.436601
RF	0.2788	0.59	15.552998	0.2649	0.58	2.101884
NB	0.1579	0.08	0.387639	**0.85**	**0.91**	**0.156209**

Table 9. Level Three TCP/UDP Mixed-based Attacks (LDAP and NetBIOS) Result.

	All Features			2 Features		
Algorithm	**Acc**	**Pre**	**Time(s)**	**Acc**	**Pre**	**Time(s)**
CBC	0.9966	1.00	9.039398	0.9910	0.99	7.836534
XGB	0.9969	1.00	5.054037	0.8776	0.90	1.625632
LGBM	0.9992	1.00	0.88380	0.9969	1.00	0.444812
RF	0.9987	1.00	6.373262	0.9938	0.99	3.723047
NB	0.9988	1.00	0.353289	**0.9954**	**1.00**	**0.151594**

5. Discussion

This paper introduced a three-level machine learning classifier for detecting DDoS attacks utilizing the CICDDoS2019 benchmark dataset. The experiment provided a new insight into the combination between the XGBoost and the LGBM (FSBLGXG) to identify the best features of a DDoS-based attack, which enhanced the model's overall performance in term of accuracy to 99.7% and precision to 97% before applying the final product classification level. Furthermore, analyzing the exact source of a DDoS attack as either UDP, TCP, or Mixed-Based using the hard voting classifier has improved the attack detection rate, allowing administrators to mitigate the potential assaults. In the second level, instead of selecting some features randomly, the whole features were utilized to better understand their correlations and provide the optimum performance of the model in the multi-classification mode.

In line with the hypothesis, the proposed architecture improved the training and validation time by about 30% compared to the recently introduced approaches, boosting the speed of early detection of attacks. Due to the high similarity characteristics of UDP-based traffic, the experiments reported the lowest accuracy compared to TCP and Mixed-based. Therefore, it is recommended to investigate the correlation of the UDP traffic features in any future studies. Also, it is recommended to test this architecture in the real time environment of a heterogeneous network.

6. Conclusion and Future Work

The use of internet devices and cloud computing continues to grow, leading to substantial growth in DDoS threats. Therefore, in this work, we built a three-level model architecture to anticipate DDoS attacks carried out by TCP, UDP effectively or mixed through the application layer. We used the CIC-DDoS2019 benchmark dataset to evaluate the model by conducting several verification experiments and analyses. The experiments stated that the proposed method outperformed the state-of-the-art ML-relevant approaches available in the literature in terms of accuracy, precision, and time. This method achieved a detection rate accuracy of 99.7% and a precision of 97% based on the results. Also, the training and validation time has been minimized by 30% compared to recently proposed methods.

For the future work, the following aspects might be consulted:

1. In the three-level classifier method, the voting classifier used in the second level has obtained ideal expectations through experimental verification. In future research, we can consider the improvement in time efficiency in the second layer by using parallelization in the voting classifiers.
2. The data used in the experiment was developed by the Canadian Security Institute for network anomaly detection. It simulates different network traffic and attack methods, with universal applicability and credibility. In order for the research results to be used for more reference, it is also necessary to test real network traffic data.
3. UDP-based DDoS attacks report the lowest accuracy so we must focus on improving this point in future works.

Acknowledgment

This paper and the research underlying it would not have been feasible without the tremendous help of Xidian University teachers in Xi'an, China. Their excitement, expertise, and demanding attention to detail have been an inspiration and kept my work on track from my first interaction until the final copy of my article.

References

[1] Salim, M. M., Rathore, S. and Park, J. H. Distributed denial of service attacks and its defenses in IoT: a survey. *The Journal of Supercomputing*, 76, 7 (2020), 5320-5363.
[2] Center, C. The 42nd Statistical Report on Internet Development in China. *Internet World*, 7 (2018).
[3] Palmer, D. *DDoS attacks that come combined with extortion demands are on the rise. ZDNet.* . City, 2022.
[4] Gutnikov, A. *DDoS attacks in Q3 2021. Yandex and Qrator Labs.* . City, 2021.
[5] Gutnikov, A. *DDoS attacks in Q3 2021. DDoS Attacks in Q3 2021.* . City, 2021.

[6] Thangavel, M., Nithya, S. and Sindhuja, R. Denial of Service (DoS) Attacks Over Cloud Environment: A Literature Survey. *Research Anthology on Combating Denial-of-Service Attacks* (2021), 491-521.
[7] Gopal, S., Poongodi, C., Nanthiya, D., Priya, R. S., Saran, G. and Priya, M. S. *Mitigating DoS attacks in IoT using Supervised and Unsupervised Algorithms–A Survey*. IOP Publishing, City, 2021.
[8] Cao, Y., Gao, Y., Tan, R., Han, Q. and Liu, Z. Understanding internet DDoS mitigation from academic and industrial perspectives. *IEEE Access*, 6 (2018), 66641-66648.
[9] contributors, W. *Denial-of-service attack*. City, 2021.
[10] Nicholson, P. *Five Most Famous DDoS Attacks and Then Some*. City, 2021.
[11] Aamir, M. and Zaidi, S. M. A. Clustering based semi-supervised machine learning for DDoS attack classification. *Journal of King Saud University-Computer and Information Sciences*, 33, 4 (2021), 436-446.
[12] Elsayed, M. S., Le-Khac, N.-A., Dev, S. and Jurcut, A. D. *Ddosnet: A deep-learning model for detecting network attacks*. IEEE, City, 2020.
[13] Kaushik Sekaran, G.Raja Vikram, B.V. Chowdary, "Design of Effective Security Architecture for Mobile Cloud Computing to Prevent DDoS Attacks ", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.9, No.1, pp. 43-51, 2019.DOI: 10.5815/ijwmt.2019.01.05
[14] Kumar, A., Glisson, W. and Cho, H. Network attack detection using an unsupervised machine learning algorithm (2020).
[15] Sreeram, I. and Vuppala, V. P. K. HTTP flood attack detection in application layer using machine learning metrics and bio inspired bat algorithm. *Applied computing and informatics*, 15, 1 (2019), 59-66.
[16] Behal, S., Kumar, K. and Sachdeva, M. D-FACE: An anomaly based distributed approach for early detection of DDoS attacks and flash events. *Journal of Network and Computer Applications*, 111 (2018), 49-63.
[17] Hoque, N., Kashyap, H. and Bhattacharyya, D. K. Real-time DDoS attack detection using FPGA. *Computer Communications*, 110 (2017), 48-58.
[18] Hameed, S. and Ali, U. HADEC: Hadoop-based live DDoS detection framework. *EURASIP Journal on Information Security*, 2018, 1 (2018), 1-19.
[19] Jazi, H. H., Gonzalez, H., Stakhanova, N. and Ghorbani, A. A. Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling. *Computer Networks*, 121 (2017), 25-36.
[20] Chen, Y., Hwang, K. and Ku, W.-S. Collaborative detection of DDoS attacks over multiple network domains. *IEEE Transactions on Parallel and Distributed Systems*, 18, 12 (2007), 1649-1662.
[21] Singh, K. J. and De, T. MLP-GA based algorithm to detect application layer DDoS attack. *Journal of information security and applications*, 36 (2017), 145-153.
[22] Singh, K., Singh, P. and Kumar, K. User behavior analytics-based classification of application layer HTTP-GET flood attacks. *Journal of Network and Computer Applications*, 112 (2018), 97-114.
[23] Liu, Z., Cao, Y., Zhu, M. and Ge, W. Umbrella: Enabling ISPs to offer readily deployable and privacy-preserving DDoS prevention services. *IEEE Transactions on Information Forensics and Security*, 14, 4 (2018), 1098-1108.
[24] Wang, C., Miu, T. T., Luo, X. and Wang, J. SkyShield: A sketch-based defense system against application layer DDoS attacks. *IEEE Transactions on Information Forensics and Security*, 13, 3 (2017), 559-573.
[25] Sokolov, M. and Herndon, N. *Predicting Malware Attacks using Machine Learning and AutoAI*. City, 2021.
[26] Rai, M. and Mandoria, H. L. *Network Intrusion Detection: A comparative study using state-of-the-art machine learning methods*. IEEE, City, 2019.
[27] Osman, M., He, J., Mokbal, F. M. M., Zhu, N. and Qureshi, S. ML-LGBM: A Machine Learning Model based on Light Gradient Boosting Machine for the Detection of Version Number Attacks in RPL-Based Networks. *IEEE Access*, 9 (2021), 83654-83665.
[28] Bakhareva, N., Shukhman, A., Matveev, A., Polezhaev, P., Ushakov, Y. and Legashev, L. *Attack detection in enterprise networks by machine learning methods*. IEEE, City, 2019.
[29] Sanjeetha, R., Raj, A., Saivenu, K., Ahmed, M. I., Sathvik, B. and Kanavalli, A. Detection and mitigation of botnet based DDoS attacks using catboost machine learning algorithm in SDN environment. *International Journal of Advanced Technology and Engineering Exploration*, 8, 76 (2021), 445.
[30] Sharafaldin, I., Lashkari, A. H., Hakak, S. and Ghorbani, A. A. *Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy*. IEEE, City, 2019.
[31] Draper-Gil, G., Lashkari, A. H., Mamun, M. S. I. and Ghorbani, A. A. *Characterization of encrypted and vpn traffic using time-related*. sn, City, 2016.
[32] Ahlashkari *GitHub - ahlashkari/CICFlowMeter: CICFlowmeter-V4.0 (formerly known as ISCXFlowMeter) is an Ethernet traffic Bi-flow generator and analyzer for anomaly detection that has been used in many Cybersecurity datsets such as Android Adware-General Malware dataset (CICAAGM2017), IPS/IDS dataset (CICIDS2017), Android Malware dataset (CICAndMal2017) and Distributed Denial of Service (CICDDoS2019).* City, 2017.

Authors' Profiles

Bassam M. Kanber is currently studying Master's degree in school of Computer Science and Technology, Xidian University, Xi'an, China. He received a Bachelor of Science in Software Engineering, Taiz University, Taiz, Yemen. His research interest includes applications of Machine Learning and Network Security.

Naglaa F. Noaman currently studying Master's degree in school of Artificial Intelligence, Xidian University, Xi'an, China. She received a Bachelor of Science in Software Engineering, Taiz University, Taiz, Yemen. Her research interest includes applications of Artificial Intelligence and Machine Learning.

Amr M. H. Saeed is currently studying Master of Science in School of Computer Science and Technology Engineering, Northwestern Polytechnical University, Xian, China. He has completed Bachelor of Science in Information Technology Engineering, Taiz University, Yemen. His current research area is Artificial Intelligence and Machine Learning.

Mansoor Malas is currently studying PhD in School of Information and Communication Engineering, State Key Lab. of ISN, Xidian University, Xian, China. He has completed Master of Science in Information and Communication Engineering, Xidian University, Xian, China. He has also completed Bachelor of Science in Telecommunication Engineering Northwestern Polytechnical University, Xian, China. His current research area includes Applications of Wireless Communications and Array Signal Processing.

How to cite this paper: Bassam M. Kanber, Naglaa F. Noaman, Amr M. H. Saeed, Mansoor Malas, "DDoS Attacks Detection in the Application Layer Using Three Level Machine Learning Classification Architecture", International Journal of Computer Network and Information Security(IJCNIS), Vol.14, No.3, pp.33-46, 2022. DOI: 10.5815/ijcnis.2022.03.03

