Detection of DDoS attacks using Machine Learning

Ms. J Anusha1, Mr.B. Akhil Krishna2 ,U. Mr. Munindhar3, Ms.K Vaishnavi4, Ms.B.Srinidhi5

1. Assistant Professor of ECE Dept. & (2,3,4,5) Students,

Department of Electronics & Communication Engineering, KITS(S), Huzurabad, Karimangar, T.S..

### Mail ID: j.anushaece@gmail.com

## ABSTRACT

Distributed Denial of Service (DDoS) assaults are cyberattacks that use numerous computers to transmit massive data packets that exhaust the resources of the computer network services. The computer network service's port mirroring allows for the observation and capture of the entire data packet as well as significant data in the form of log files delivered by the attacker. Network traffic must be divided into two conditions: regular traffic and attack traffic, according to the classification system.

Various machine learning methods, including support vector machines (SVM), Random Forest, XGBOOST, ADABOOST, KNN, and Naive Bayes, have been used in this research. The random forest model outperformed traditional algorithms in terms of performance. We used the Canadian Institute for Cyber Security (CIC) dataset to train these algorithms. It covers 10 possible attacks of the IOT environment and benign (normal) class. One approach for processing numerical attributes as input and determining whether access to a computer network service will be "normal" access or access under "attack" by DDoS, is the Random Forest classification.

The purpose of the research is to train a model using machine learning approaches that can detect and categorize the type of DDoS assault with more accuracy than each individual machine learning technique utilized.

**Keywords:** DDoS Attack, IOT Environment, SDN, Forest Fire, LoRa.

## Introduction

Deep packet analysis is provided through the whole network view in the revolutionary architecture environment of SDN. It encourages prompt response and updating of traffic laws and regulations. The SDN is capable of perceived control over the overall visualization view, flexible and schedule-able rapid deployment, and service-open intelligent scheduling. The software defined network improves user experience and makes it easier to promote the implementation of the entire network while ensuring network services and lowering deployment costs. Numerous DDoS attack detection techniques were presented by researchers who focused on traditional network design. The control and data planes, which are typically mixed together in traditional networks, are separated by a novel architecture introduced by software-defined networking (SDN). In essence, SDN separates a network into three layers: the data layer, the control layer, and the application layer. SDN switches in the control plane are governed by a centralized controller while being cognitively incapable in the data plane. The simplicity of administration makes this revolution's benefit clear. The controller, however, has a high risk of becoming a single point of failure. In other words, if the controller goes down, all the linked SDN switches could stop functioning if they lose contact with the controller. The Open Flow protocol defines the link between the control and data planes, and it consults with the controller about how to process specific packets.

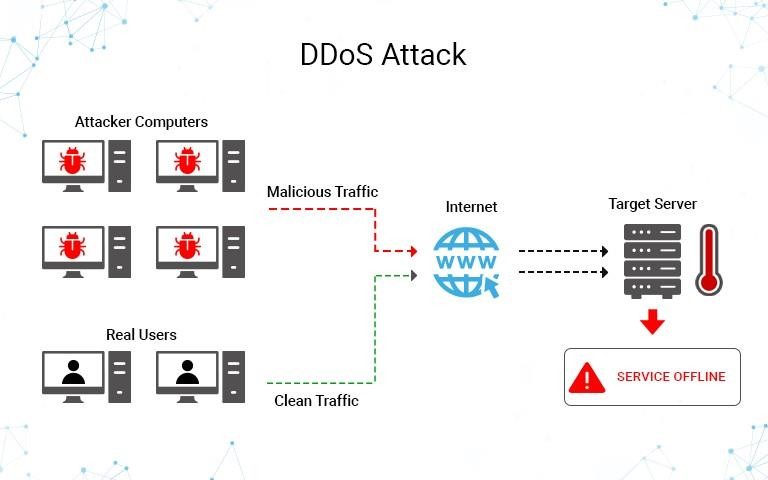


Fig: 1 DDos Attack

IoT makes it possible for data sharing and machine-to-machine connectivity, which increases the area of coverage. IoT aims to integrate a wide range of devices and networks to enable localisation, monitoring, management, etc. through sensors identification and ubiquitous computing. The number of devices connected to the network in outdoor deployments has an exponential increase as a result of the high need for data collecting and environment monitoring. Recently, new technologies have emerged, such as LoRa, DASH7, and Narrowband (NB-IoT), that promise to deliver low-power and long-range connectivity solutions for IoT applications. These technologies also promise to meet the essential criteria of affordability, battery life, coverage area, scalability, security, and privacy. IoT networks face new security challenges as a result of the enormous number of IoT devices connected to the same network, which widens the attack surface.

## Problem Overview

The services of networks with crucial business and industry information have been spread to the production and life of contemporary society as a result of the ongoing development of network technology, the endless expansion of network business requirements, and the explosive growth of the Internet economy in the Internet age. The emergence of DDoS attacks can cause anomalies in the connected network services, resulting in significant financial losses and potentially other disastrous effects. One of the major dangers to network security that the Internet is subject to is DDoS attacks. Accurate and rapid DDoS attack detection is a major study area in the security industry. The network data plane and the control plane, which has the properties of network programmability and centralized management control, are separated by the new network innovation architecture known as SDN. Network attackers attack network bandwidth, system resources, and application resources, to achievethe effect of denial of service attacks.

The following are the challenges in detecting DDoS attacks:

(1) Attack traffic characteristics that are difficult to identify;

(2) Lack of cooperation between coherent network nodes;

(3) Strengthening of the attack tool with a lower threshold of use;

(4) Widespread address fraud making it difficult to identify the attack's source;

(5) Attack duration and response time limitations.

The two primary types of DDoS attack detection technologies used in traditional network architecture are attack detection based on traffic characteristics and attack detection based on traffic abnormality. The former primarily develops a DDoS attack characteristics database by gathering various types of attack characteristic information. We can determine whether a network is being attacked by DDoS by comparing and analyzing the data information of the current network data packet and characteristics database. Characteristics matching, model reasoning, state transition, and expert systems are the primary implementation techniques. The purpose of the latter is primarily to create a traffic model and analyze variations in flow that are aberrant, in order to determine if the traffic is abnormal or not and identify whether the server has been attacked.

## Literature Survey

A DDoS assault detection and defense mechanism based on SDN was proposed by Lin and Wang [5], however the deployment and operation were complicated due to the method's use of three Open Flow management tools using the sFlow standard to perform anomaly detection. Using a single flow information and IP entropy characteristic information, Yang et al.'s [6] technique for detecting IP traffic with a greater and more precise detection effect combines both the flow information and the IP entropy characteristic information. Even though information entropy is adaptable and practical, additional technologies must still be used to determine the threshold and multi element weight distribution.

Saied et al. [7] presented the idea that the approach needs to identify packet protocol, which is complex and inefficient, based on analysis of the features of each TCP/UDP/ICMP protocol through the training ANN algorithm to detect DDoS attacks. By identifying the flow data associated with DDoS attacks, the SOM method is utilized to identify DDoS attacks. This approach provides a high detection rate while using little energy. The extraction of the time period is the crucial component. The drawback of this approach is that the assault behavior is not promptly and precisely identified, and there is some hysteresis in the detection.

A framework for DDoS attack detection and mitigation in a large-scale network was proposed by the authors in, although it is not appropriate for small-scale deployment.

In a genuine source and destination IP address database-based DDoS attack detection system is suggested. Based on the nonparametric cumulative algorithm CUSUM, the technique must be adjusted and the threshold must be determined. When a DDoS assault happens, it analyzes the anomalous characteristics of the source IP address and the destination IP address and effectively verifies the DDoS attack.

The SOM algorithm must anticipate the amount of neurons due to the high information entropy false positive rate. As a result, we list the features of various DDoS assaults, gather data from switch flow tables, extract the six-tuple characteristic values matrix, and create an SVM classification model for each attack. The technique has the ability to analyze multidimensional data and translate low-dimensional nonlinearly separable data into a high-dimensional feature space to make it linearly separable and capable of being classed accurately. The technique is currently utilized extensively in anomaly detection and categorization. During a DDoS assault, Dao et al. [12] construct a table in the controller to track the packets by IP address. To distinguish between a legitimate request and an attack, the quantity of packets using that connection is also compared with a minimum value. According to the simulation, this strategy successfully lowers flow entries in the switch while maintaining the controller-switch channel's bandwidth during DDoS attacks. If the attacker updates the source address, this approach uses quite a lot of controller resources. Entropy may assess unpredictability, where time period and threshold are two crucial components, and Mousavi and St-Hilaire [13] propose using it for DDoS detection. The offered solutions exclusively deal with detection without offering countermeasures, even if it might increase detection accuracy in the actual network. The Sequential Probability Ratio Test (SPRT), a statistical method, is recommended by Dong et al. [14] to address the current false positive and false negative problems. The decision is based on the log-probability ratio and predefines two boundaries (A and B, B A) related to the probabilities of false positive (a) and false negative (b) (it is proposed A = b/ (1 a), and B = (1 b)/a). The DARPA Intrusion Detection Data Sets examination demonstrates its promptness and accuracy. The proposed approach, however, is assessed solely on the basis of mathematical findings rather than simulations, which can include random factors.

In order to execute requests in each time slot so that legitimate users can correctly connect with one another during DDoS assaults, Yan et al. [15] suggest the "Multislot" method.

## Low rate DDOS attack detection

Low-rate Distributed Denial-of-Service (DDoS) attacks provide a fresh danger to the internet since they obstruct legitimate traffic by sending a large volume of attack packets that are identical to other types of traffic. Zhang and co. For identifying and filtering DDoS attacks at low rates, a congestion-participation (CPR) metric and CPR-based strategy were proposed. They discovered that although regular TCP flows actively reduce network congestion, low-rate DDoS flows actively cause network congestion. The suggested technique was designed to separate attack flows from legal flows. However, more research and analyses utilizing actual datasets are needed to evaluate its efficacy. In order to identify both short-term high-rate DDoS attacks and long-term low-rate DDoS attacks, Du and Abe [6] suggested an entropy metric for the size of an IP packet. They stated that the distribution of the packet size changes under assaults, and this can be utilized to identify attacks to some extent. This was based on the idea that many applications have typical packet sizes dependent on requests for and answers to data and acknowledgments. However, the proposed solution is limited in terms of scalability because it significantly depends on the packets in the observation window, and it requires a lengthy detection period to obtain a high chance of detection while being subject to a low-rate DDoS attack. An effective, impartial approach of entropy-based DDoS assault low-level detection was proposed by Jadhav and Patil[5]. This method significantly outperforms the use of traditional metric entropy. However, because there is a very minor difference between normal traffic and attack traffic, the false positive rate is rising.

**High rate DDOS attack detection** A thorough analysis of the defense mechanisms against spoof DDoS assaults has been conducted. Each strategy has benefits and drawbacks. To start a spoof DDoS attack, the attacker forges TCP/IP header information. While it is possible to forge any TCP/IP header field, it is impossible to create the Time-To-Live (TTL) field. TTL value is therefore utilized to distinguish faked IP packets. Since the header only provides the final TTL value, this computation presents a problem. Each Operating System (OS) has a unique initial TTL value, and the OS for a specific IP address may vary over time. If the valid packet originates from an unlisted OS, the method will result in a false positive, and if the attacker properly estimates the hop count between the source and victim, a false negative. to distinguish between legal traffic and attack traffic. It is suggested to use a path fingerprint strategy, where each packet has a different path fingerprint. The route that a packet takes to get to its destination is represented by the path fingerprint. The packet is labeled as spoofed when the path fingerprint is inaccurate. Since the packet arrives at the same subnet and requires calculation at the intermediary nodes, the technique is unable to recognize subnet faking. The TTL, IP Don't Fragment (DF), window size, and total length values in the TCP/IP header fields are used to determine the OS of a packet. These values are used to construct a fingerprint, which is then appended to the packet at the source. The packet is regarded as valid if the fingerprint matches at the receiver side; otherwise, it is handled as a faked packet.

## EXISTING METHODOLOGIES

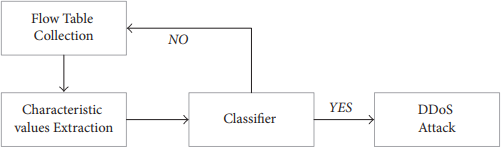
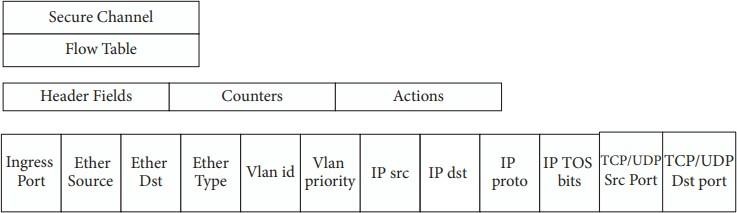
The SDN architecture's Open Flow switch expeditiously forwards the primary network data. The SDN controller is responsible for the collection of switch traffic data, administration of the forwarding decision, and forwarding. The key data structure for the forwarding policy management control in the SDN switch is the flow table. The SDN searches the flow table entries to handle the relevant network traffic, which can send the packet to one or more interfaces. A header field, counters, and actions are all present for each entry. The switch's packet routing is built on top of the flow table. Each flow table consists of several flow entries. The entries in the flow table define the rules for data forwarding. Figure 1 displays the flow table entry structure diagram. The majority of the attack detection flow diagram is represented by the extraction characteristic values, classifier judgment, and flow state collection, as shown in Figure 2. The Open flow switch responds with the flow table information after the flow state collection periodically asks a flow table from it. The characteristic values extraction, which is principally accountable for getting the characteristic values connected with the DDoS attack from the switch flow table, produces the six-tuple characteristic values matrix. The Open Flow protocol talks to the controller about how to handle specific packets in order to define the connection between the control and data planes.

Figure 2: Flow table structure. Figure 3: Attack detection process

## PROPOSED METHODOLOGY

Although the attacker usually has no intention of stealing any data, DDoS attacks are a frequent hazard to the network. DDoS assaults basically try to exhaust system resources until the target is unable to provide its services. Application layer attacks, protocol attacks, and volumetric attacks are the three subtypes of DDoS attacks. An attacker can use a volumetric attack to consume all of the victim's resources or bandwidth directed at the target. This type of attack might affect not just the data plane in the SDN but also the controller and southbound interface since a client host could cause an inquiry from the data plane to the control plane. Although DDoS assaults in SDN and IoT networks have been extensively discussed, the abundance of IoT devices and the SDN communication link between controllers and switches still present a strong opportunity for attacks. More validations in the actual network are also necessary. Additionally, the SDN's programmability and centralized control allow users more ways to investigate this problem. Volumetric attack is used in this paper.

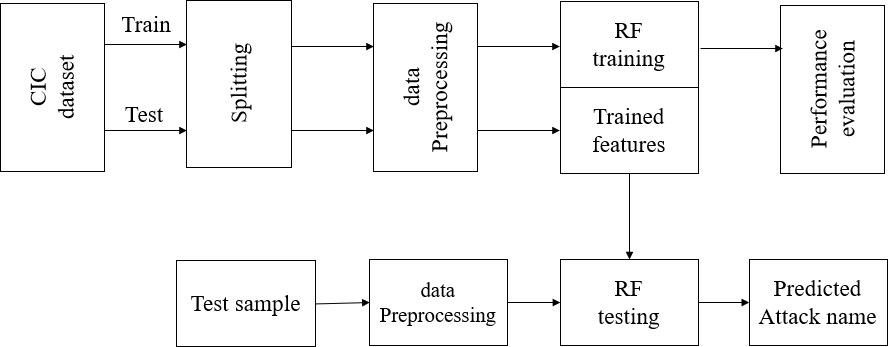


Figure 4 Proposed block diagram

# The CIC dataset is initially divided into 20% for testing and 80% for training. The entire dataset is then normalized using a dataset preprocessing operation. Additionally, a random forest classifier is utilized to anticipate DDoS attacks using test data.Performance testing is done to demonstrate the superiority of the suggested approach.

# Performance EvaluationCIC dataset:

The benign and most recent DDoS attacks found in CICDoS2019 closely mimic PCAPs from actual real-world data. Additionally, it contains the output of the CICFlowMeter-V3 network traffic analysis, labeled flows based on the time stamp, source and destination IP addresses, source and destination ports, protocols, and attack (CSV files). In the proposed testbed, we have employed our B-Profile system to produce naturalistic benign background traffic and profile the abstract behavior of human interactions (Figure 2). On the basis of the HTTP, HTTPS, FTP, SSH, and email protocols, we constructed the abstract behavior of 25 users for this dataset.

# Preprocessing:

Preparing raw data to be acceptable for a machine learning model is known as data preparation. In order to build a machine learning model, it is the first and most important stage. It is not always the case that we come across the clean and prepared data when developing a machine learning project. Additionally, any time you work with data, you must clean it up and format it. Therefore, we use a data pretreatment activity for this.

**Need of Data Preprocessing:** Real-world data typically includes noise, missing values, and may be in an undesirable format, making it impossible to build machine learning models on it directly. Data preprocessing is necessary to clean the data and prepare it for a machine learning model, which also improves the model's accuracy and effectiveness.

* Getting the dataset
* Importing libraries
* Importing datasets
* Finding Missing Data
* Encoding Categorical Data
* Splitting dataset into training and test set
* Feature scaling

# Feature Selection:

# One of the fundamental ideas in machine learning, feature selection has a significant impact on the effectiveness of the model. The performance your machine learning models can attain will be greatly impacted by the data characteristics you use to train them. The most crucial phase of creating your layout is feature collection and data cleaning. The process of feature selection involves choosing, either manually or automatically, the features that have the greatest impact on the predictive variable or output you are interested in. The accuracy of your model can be decreased if your data contains irrelevant characteristics, and your model can be trained using irrelevant information.

# Model Training:

Run Algorithms: Using this module, we will feed 80% of the training data into Random Forest, XGBOOST, ADABOOST, SVM, Nave Bayes, and KNN algorithms to train a model, which will then be used to test data to measure prediction accuracy.Comparison Graph: We will provide a comparison table and graph of all methods utilising this module.

**Predict Attack from Test Data:** We will submit test data to this module, and machine learning models will predict attacks based on that data. You can discover test data within the test folder, and this test data contains all features without a class name, which will be predicted by the machine learning model.

Use Case Diagram

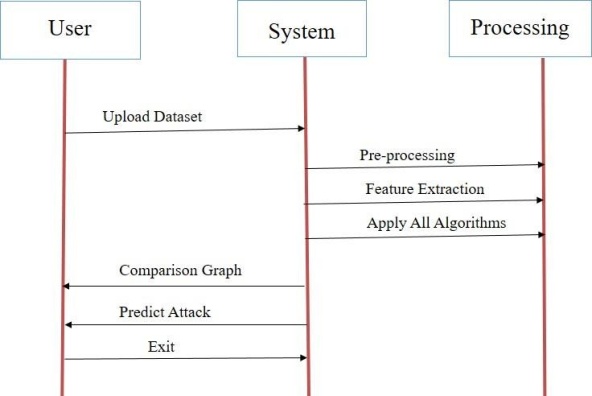
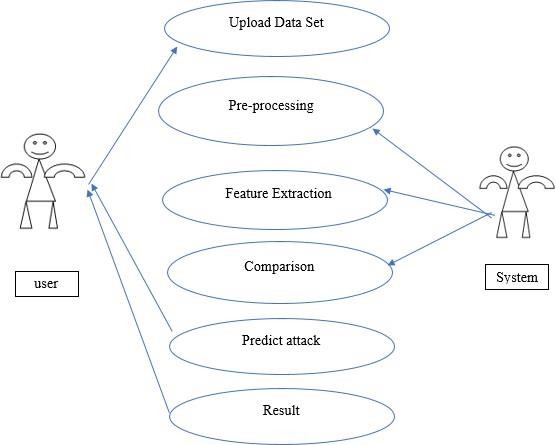


Figure 5.1: Use case diagram Figure 5.2: Sequence diagram

## RESULTS

Test Data you can find inside test folder and this test data contains all featureswithout any class label and this label will be predicted by machine learning model.

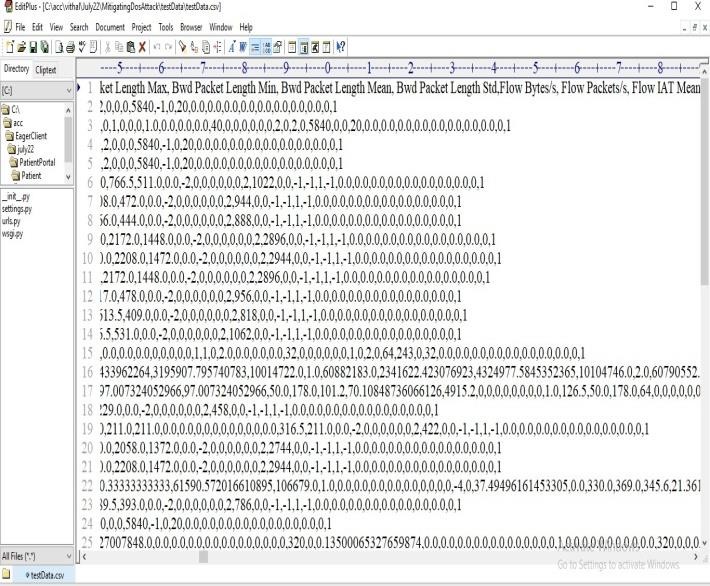


Figure 6.1: Test Data

In above TEST DATA screen there is no class label or attack name and this will be predicted by ML model.

**SCREEN SHOTS:**

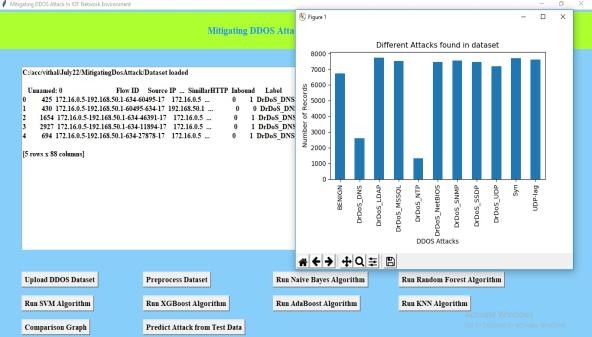


Figure 6.2: Graphical user interface (GUI) Application Figure 6.3: Different Attacks found in dataset

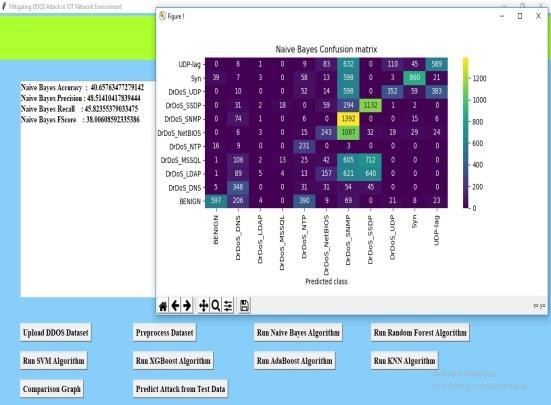




Figure 6.4: Preprocessing Dataset Output Figure 6.5: Naïve Bayes Confusion

To launch the project, double-click the 'run.bat' file to obtain the screen matrix shown below. In the above screen, click the 'Upload DDOS Dataset' button to upload the dataset and obtain the output shown below.

On the panel up top, where the dataset is loaded, we can see that it contains both numerical and non-numerical data. Attack names are displayed on the x-axis, and the number of such recordings is displayed on the y-axis. After closing the previous graph, select "Preprocess Dataset" to process the dataset and display the screen shown below.The dataset, which has more than 70000 records and 87 features per record, is displayed in its entirety in the screen above. The dataset has been divided into train and test applications, with the training application employing 56685 records for training and 14172 for testing. Once the train and test data are prepared, select "Run Nave Bayes."

In above screen with Naïve Bayes we got 40% accuracy and in confusion matrix graph x- axis represents PREDICTED classes and y-axis represents TRUE classes and prediction count in same row and column names are the correct prediction and count in different row and column names are the incorrect prediction and we can see Naïve Bayes predicted so many wrong prediction and close above graph and then click on 'Run Random Forest Algorithm' button to get below output. With Random Forest, we achieved greater than 96% accuracy in the image above, and the graph also shows that many of the predictions were accurate. Now that the above graph is closed, click the "Run SVMAlgorithm" button to obtain the output shown below. Close the graph in the above screen after achieving 67% accuracy with SVM, and then click the "Run XGBOOST Algorithm" button to obtain the output shown below.

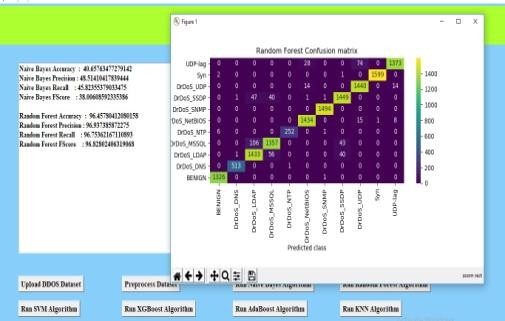
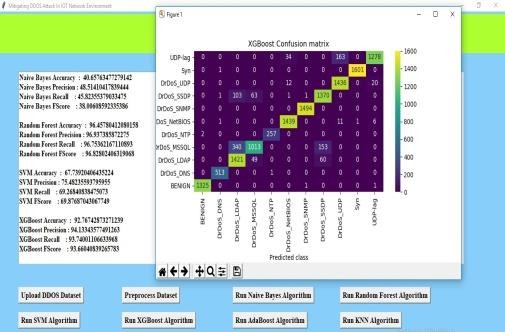
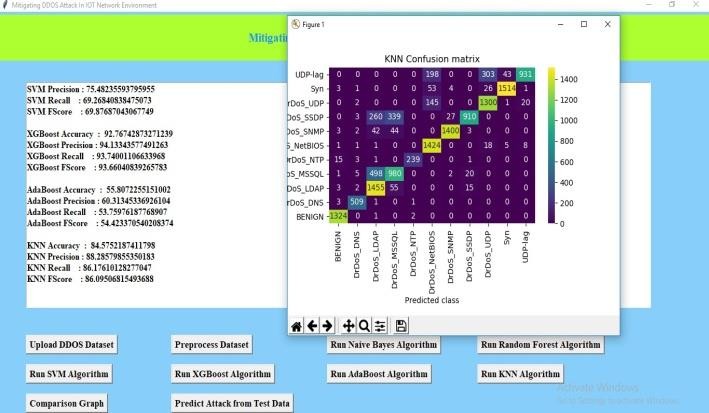
  

Figure 7.1: Random Forest Confusion matrix Figure 7.2: SVM Confusion matrix Figure 7.3: XGBOOST Confusion matrix

Close the above graph after achieving 92% accuracy with XGBOOST, and then click the "Run ADA BOOST Algorithm" button to obtain the output shown below .



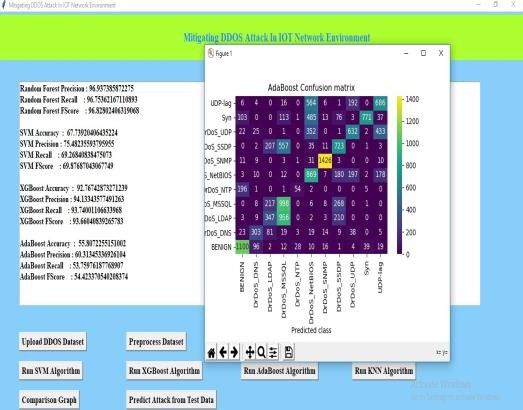


Figure 7.4: ADABOOST Confusion matrix Figure 7.5: KNN confusion matrix





Figure 7.6: Comparison Table Figure 7.7: Test Data Output In above

In above screen with ADABOOST we got 55% accuracy and now close above graph andthen click on ‘Run KNN Algorithm’ button to get below output In above screen with KNN we got 84% accuracy and now close above graph and then clickon ‘Comparison Graph’ button to get below ggraph and comparison table we can see Random Forest got high accuracy and in above graph different colour bar represents different metrics such as accuracy, precision, recall and FSCORE. Now click on ‘Predict Attack from Test Data’ button to upload test data and get below output

## Selecting and adding the TEST DATA file to the screen above, then clicking the 'Open' button to view the results, will produce. With each individual test record, several attacks and benign (normal) classifications are anticipated in the displays above.

## Conclusion and Future Scope

DDoS assaults have grown in significance as a network security problem with the advancement of network technology. When attacking, it employs widely used protocols and services, making it challenging to identify using conventional techniques. DDoS attack detection can be modelled as a classification issue that distinguishes between "rational" and "irrational" network flow states based on the concept of rational thought. The common TCP flood attack, UDP flood attack, and ICMP flood assault are all thoroughly examined in this article. To describe attack behavior, define the properties of data stream information entropy. It is suggested to use a random forest classification algorithm to detect DDoS attacks. Create classification models for the aforementioned three categories of common assault techniques. It is finally anticipated whether the network traffic is normal by training and learning.

# Future Scope

Nowadays, static and dynamic analysis of request data is used to find cyberattacks. Static analysis is based on signatures, and to determine if a packet is normal or contains an attack signature, we compare the new request packet contents with the current attack signature. To find malware or attacks, dynamic analysis will use dynamic program execution, however dynamic analysis takes time. We are using machine learning algorithms to solve this issue and improve the detection accuracy of both old and new malware attacks. These algorithms include Support Vector Machine (SVM), Random Forest, Decision Tree, Naive Bayes, Logistic Regression, K Nearest Neighbors, and Deep Learning Algorithms like Convolution Neural Networks (CNN) and LSTM (Long Short-Term Memory). Various models are among them. When compared to other models, deep learning CNN performed better.

**REFERENCES:**

[1]. H. Zhang, Z. Cai, Q. Liu, Q. Xiao, Y. Li, and C. F. Cheang, “A surveyon security-aware network measurement in SDN,” Security and Communication Networks, Article ID 2459154, 2018.

[2]. J. Cao, M. Xu, Q. Li, K. Sun, Y. Yang, and J. Zheng, “Disrupting SDN via the data plane: a low-rate fow table overow attack,” in Proceedings of the 13th EAI International Conference onSecurity and Privacy in Communication Networks, Niagara Falls, Canada, October 2017. 8 Security and Communication Networks

[3]. Z. Cai, Z. Wang, K. Zheng, and J. Cao, “A distributed TCAM coprocessor architecture for integrated longest prefx matching, policy fltering, and content fltering,” IEEE Transactions on Computers, vol. 62, no. 3, pp. 417–427, 2013.

[4]. Y. Li, Z. Cai, and H. Xu, “LLMP: exploiting LLDP for latency measurement in sofware-defneddata center networks,” Journal of Computer Science and Technology, vol. 33, no. 2, pp. 277– 285, 2018.

[5]. H. Lin and P. Wang, “Implementation of an SDN-based security defense mechanism against DDoS attacks,” in Proceedings of the 2016 Joint International Conference on Economics and Management Engineering (ICEME 2016) and International Conference on Economics and Business Management (EBM 2016), Pennsylvania, Penn, USA, 2016.

[6]. J. G. Yang, X. T. Wang, and L. Q. Liu, “Based on trafc and IP entropy characteristics of DDoSattack detection method,” Application Research of Computers, vol. 33, no. 4, pp. 1145–1149, 2016.

[7]. A. Saied, R. E. Overill, and T. Radzik, “Detection of known and unknown DDoS attacks usingartifcial neural networks,” Neurocomputing, vol. 172, pp. 385–393, 2016.

[8]. R. Braga, E. Mota, and A. Passito, “Lightweight DDoS fooding attack detection using NOX/OpenFlow,” in Proceedings of the 35th Annual IEEE Conference on Local Computer Networks (LCN ’10), pp. 408–415, Denver, Colo, USA, October 2010.

[9]. N. Z. Bawany, J. A. Shamsi, and K. Salah, “DDoS attack detection and mitigation using SDN: methods, practices, and solutions,” Arabian Journal for Science and Engineering, vol. 42, no. 2, pp. 425– 441, 2017.

[10]. X. Wang, M. Chen, C. Xing, and T. Zhang, “Defending DDoS attacks in sofware-defned networking based on legitimate source and destination IP address database,” IEICE Transaction on Information and Systems, vol. E99D, no. 4, pp. 850–859, 2016.

[11]. J. Xia, Z. Cai, G. Hu, and M. Xu, “An active defense solution for ARP Spoo ng in OpenFlow network,” Chinese Journal of Electronics, vol. 3, 2018.

[12]. Dao, N.N.; Park, J.; Park, M.; Cho, S. A feasible method to combat against DDoS attack in SDN network. In Proceedings of the 2015 International Conference on Information Networking (ICOIN), Siem Reap, Cambodia, 12– 14 January 2015; pp. 309–311. doi:10.1109/ICOIN.2015.7057902.

[13]. Mousavi, S.M.; St-Hilaire, M. Early detection of DDoS attacks against SDN controllers. In Proceedings of the 2015 International Conference on Computing, Networking and Communications (ICNC), Anaheim, CA, USA, 16–19 February 2015; pp. 77–81. doi:10.1109/ICCNC.2015.7069319.

[14]. Dong, P.; Du, X.; Zhang, H.; Xu, T. A detection method for a novel DDoS attack against SDN controllers by vast new low-traffic flows. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 23–27 May 2016; pp. 1–6. doi:10.1109/ICC.2016.7510992.