iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** III **Month of publication:** March 2022

DOI: https://doi.org/10.22214/ijraset.2022.41131

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue III Mar 2022- Available at www.ijraset.com

D-Dos Attack Prediction Using Machine Learning Algorithms

Prof. Amit Narote[1], Vamika Zutshi[2], Aditi Potdar[3], Radhika Vichare [4]
[1]*Professor,* [2, 3, 4] *Student, Department of IT, Xavier Institute of Engineering, Mumbai, India*

Abstract: ***The risk of cyber-attack keeps on growing irrespective of development of new technologies for protection. One of the most frequent cyber-attacks is the DOS attack. A Denial-of-Service (DoS) attack is an attack which tries to shut down a machine or network, by flooding the target with unwanted traffic or triggers a crash by sending it some information, which makes it challenging for the users to access their network. A higher version of DoS attacks is the DDoS attacks that have recently become quite severe in security companies. Many organizations have begun facing these issues. Such attacks are very well coordinated that disrupts the normal functioning of the networking system from large firms to small scale business. Hence, detecting such attacks has become a tedious task. However, such a classification problem can be resolved using machine learning. Also, the same problem can be addressed using the concepts of cloud computing in order to detect and identify the computational effort carried out by the attacks. A DoS is generally considered to be an organized attack by hackers that is implemented from a single source of origin and targeted towards the victim's end. In order to attack these systems such attackers impersonate themselves as legit users and gain access from the users by asking them their personal credential and details. As compared to this, a DDoS attack is limited to a single source of origin and takes place on distributed computers all together. Hence the primary aim of this thesis is to identify such attacks caused by hackers and detect them using machine learning algorithms.***

Keywords: Denial-of-Service (DoS), Distributed Denial-of-Service (DDoS), Machine Learning, Bots, Botnets, flooding attacks.

I. INTRODUCTION

A DoS is generally considered to be an organized attack by hackers that is implemented from a single source of origin and targeted towards the victim's end. In order to attack these systems such attackers impersonate themselves as legit users and gain access from the users by asking them their personal credential and details. In contrast to this, a DDoS attack is limited to a single source of origin and takes place on distributed computers all together. Hence the primary aim of this thesis is to identify such attacks caused by hackers and detect them. The authors make use of machine learning algorithms to make use of the same and further tries to prevent the occurrence of such attacks. Historically witnessed, a (DoS) Denial of Service attack is injected into a system to interrupt the normal functioning of a computational server in a network. These attacks are originated from a single server with the pure intention of the hacker to attack a targeted server. A commonly injected DoS attack could possibly be a PING attack and a more complex attack observed would be a PING of death attack. On the other hand, a (DDos) Distributed Denial of Service attack is carried out in a distributed environment, different from that of a DoS attack performed through a single server. Hence, it is said that a DDoS attack is executed in a distributed environment by an attacker who targets the server and intentionally attacks it to reduce it normal performance, making it inaccessible to legitimate users. He achieves this through numerous frameworks in a system and targets a website or a server by making multiple requests over a span of time. The most traditional form of a DDoS attack is brute force attack that is triggered using a Botnet which results into infected malwares on networking devices.

DoS (Denial of Service) attacks deprive the bandwidth of the network and computational capabilities of a target device by flooding malicious traffic, restricting the target system from providing regular services to authorized customers. DDoS (Distributed Denial of Service) takes things a step further. DDoS attacks take control of the majority number of compromised systems, known as a botnet, and release simultaneous attacks on the victim system. DDoS attacks are emerging and propagating in scale, frequency, and sophistication in tandem with the occurrence and advancement of disruptive Internet technologies (Genie-Networks).

II. RESEARCH OBJECTIVES

The goal of this attack is to overwhelm the network or server with traffic. It achieves success by using various compromised systems as attack traffic sources. DDoS attacks are classified into subcategories based on the layer of the network connection that they attempt to attack in accordance with the OSI model. SYN Flood, UDP Flood, MSSQL, LDAP, Portmap, and NetBIOS are some of the subcategories that we identified during our research. Machine Learning and Deep Learning are two of the most common A.I. backbones today. We use these methodologies to solve problems in a variety of domains with near-human precision. Through this research, we have once again tested the limits of A.I. in exposing threats in the domain of cybersecurity.

A. Literature Survey

At a high level, a DDoS attack can be viewed as obstructed unexpected traffic on the highway that prevents regular traffic flow from arriving at its destination. DDoS attacks are typically carried out using a network of interconnected devices that are all linked via the internet. The main issue with these types of attacks is that it is difficult to distinguish between normal and attack traffic because each Bot acts as if it is legitimate. The use of technological advancements is observed for various functions of the business and is a part of modern evolution. Technological advancements have given rise to various negative impacts that include cyber-attacks, and the DDoS attack is one of them. Approximately 45% of cyber-attacks across the globe are identified as DDoS attacks. The attack needs to be prevented and the study focuses on providing mitigation techniques for the issue. However, the study also includes the necessary approach to mitigate the problem. The main focus of this chapter is to provide some information regarding DDOS with the help of a literature review. Different types of concepts as well as the evaluation of acceptable scenarios are described here. Valuable results, framework and the application perspective of advanced technologies are also involved with the study as well. The study also put some importance on the future perspective where the developmental aspect (software technologies) needs to be maintained.

III.IMPLEMENTATION AND DESIGN

In recent times, one of the most observed internet threats happen to be the DDoS attack. One of the basic working principles of this concept is the detection of attack packets much before time. However, traditional methods are still incapable to distinguish between attack strategies and legitimate network traffic. Therefore, the fundamentals of Machine learning are used to detect the same using statistical features. This chapter of the thesis focuses on the design and implementation of the project using various ML techniques by deducing a hybrid model.

The concerned research has been developed by following the deductive research approach, and the deductive approach for this research study helps to find out the reasoning of the collected data. It helps within the analysis of secondary data from every possible angle. In aspects of conclusion, logical arguments will be provided by following this specific approach. in keeping with information from secondary resources, a Denial-of-Service attack can affect business by minimizing its value because it disrupts communication and it can last quite 24 hours also. It generally prevents the websites from working properly and users get harassed for this reason as business operations and other important activities have gotten hampered for uncommon behavior of internet sites. status online servers like credit or revolving credit payment gateways, internet banking services often experience DDoS attacks.

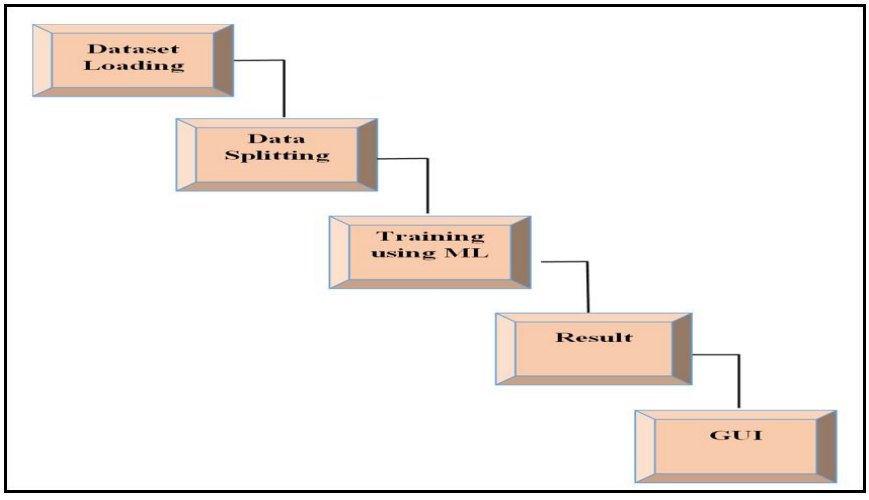


Fig. 1 Block Diagram

A. Dataset used: -CICIDS2017

CICIDS2017 dataset become currently advanced by ISCX and incorporates benign visitors and consequently the maximum modern-day commonplace attacks. This new IDS dataset consists of seven not unusual up to date circle of relatives of attacks that met the real-international standards CICIDS2017 dataset contains benign and therefore the most recent common attacks, that resembles verity real-world knowledge (PCAPs). It additionally includes the results of the network traffic analysis victimisation CICFlowMeter with labelled flows supported the time stamp, source, and destination IPs, supply and destination ports, protocols and attack (CSV files).

The dataset chosen for experimentation consisted of five-day log records from weekday to Friday in csv format. For experiment analysis, we've thought of the log file of Friday afternoon that additionally consisted of 2 category labels. the category labels square measure Benign (Normal) and DDoS (attack). the entire range of traffic packets within the log file enclosed 225,746 traffic packets. Initially, the number of attributes within the weekday afternoon logfile area unit seventy-eight with the last attribute being the category label, i.e., there are a unit seventy-nine dimensions beside category label.

B. Machine Learning Algorithms

The Machine Learning algorithms that we have chosen for detecting the Ddos Attack are as follows:

1) *Naïve Bayes:* One of the most commonly used machine learning algorithms is Naïve Bayes. It is completely based on the Bayes theorem that allows the likelihood of an event to take place based on the prior knowledge of specific conditions associated with the event. This is a classification method based on Bayes' theorem and assumes predictor independence. Simply put, the naive Bayes classifier assumes that the presence of a particular feature in a class is independent of the presence of other features. The Naive Bays model is easy to build and is especially useful for very large datasets. In addition to simplicity, Naive Bayes is known to be superior to the most sophisticated classification methods. Naive Bayes is a machine learning model used for big datasets. Even if you are working with data that contains millions of records, the recommended approach is Naive Bayes. Very good results are obtained for NLP tasks such as sentiment analysis. This is a fast and simple classification algorithm.
2) *Logistic Regression:* Logistic regression is a "supervised machine learning algorithm" that can be used to model the probabilities of a particular class or event. This is used when the data is linearly separable and the result is essentially binary or dichotomous. This means that logistic regression is usually used for binary classification problems. As many of the previous examples suggest, logistic regression is used in data science as a supervised classification model for machine learning. This helps predict category trends with high accuracy. Using the examples of high and low risk of cancer, this prediction can be categorized into more detailed categories according to the needs of the researcher. The proposed methods primary research goal is to create a machine learning model based on multiple linear regression analysis. The proposed approach is designed to investigate the feasibility of using multiple linear regression analysis on the dataset, which is the benchmark dataset widely used in some of the most noteworthy recent research studies.
3) *Random Forest:* A random based classifier is a collection of decision trees that are chosen at random from a subset of the training set, and then the votes from all the decision trees are aggregated and the final class of the object tested is determined. This classifier is commonly used because it is very efficient with large datasets and can handle a large number of input variables without removing any variables. Furthermore, it prevents overfitting by increasing the accuracy score while training on the dataset. Furthermore, as the forest grows, it generates unbiased generalization error estimates.
4) *Ada Boost:* The AdaBoost algorithm, which stands for Adaptive Boosting, is a boosting method used as an ensemble method in machine learning. This is called adaptive boosting because the weights for each instance are reassigned and the misclassified instances are assigned higher weights. Boosting is used to reduce the bias and variance of supervised learning. It works on the principle that learners grow in turn. With the exception of the first learner, each subsequent learner grew up from a previous adult learner. Simply put, a weak learner turns into a strong learner. The AdaBoost algorithm works on the same principle as boost, with one subtle difference. This algorithm is used to increase the accuracy of other algorithms.

C. Dataset Loading

The log file of Friday afternoon that additionally consisted of 2 category labels. the category labels square measure Benign (Normal) and DDoS (attack). the entire range of traffic packets within the log file enclosed 225,746 traffic packets. We will import different packages required accordingly. There are no recent datasets found in the public domain that are solely for DDoS, though IDS data sets are available. As a result, we extracted DDoS flows from the following public IDS dataset CICIDS2017.

Out[1]:

	Destination Port	Flow Duration	Total Fwd Packets	Total Backward Packets	Total Length of Fwd Packets	Total Length of Bwd Packets	Fwd Packet Length Max	Fwd Packet Length Min	Fwd Packet Length Mean	Fwd Packet Length Std	...	min_seg_size_forwar
0	36206	54	1	1	0	0	0	0	0.0	0.0	...	3
1	53524	1	2	0	0	0	0	0	0.0	0.0	...	3
2	53524	154	1	1	0	0	0	0	0.0	0.0	...	3
3	53526	1	2	0	0	0	0	0	0.0	0.0	...	3
4	53526	118	1	1	0	0	0	0	0.0	0.0	...	3

5 rows × 79 columns

Fig. 2 Dataset

D. Train and Test split data

The dataset is split for training and testing in this operation. These two datasets are needed to train the estimator and then test the performance of the corresponding model. These training and test datasets are created using two common techniques. The percent split and K-fold cross validation are the techniques used.

```
In [4]: X_train, X_test, y_train, y_test =train_test_split(X, y ,test_size=0.4)
```

Fig. 3 Data Splitting

E. Training

The creation of a model for classification or other related tasks is at the heart of ML-based work. That is what the training accomplishes. An ML algorithm is trained on a subset of the overall dataset, the training dataset, which was previously prepared in the data split section. An algorithm that has been trained produces a model that has learned from data. There are several estimators available for classification.LR, RF, and others contributed to this work.

F. GUI

Python has lots of GUI frameworks, however Tkinter is the handiest framework that's constructed into the Python general library. Tkinter has numerous strengths. It's cross-platform, so the equal code works on Windows, macOS, and Linux. Visual factors are rendered the use of local running device factors, so programs constructed with Tkinter appear to be they belong at the platform wherein they're run. Although Tkinter is taken into consideration the de-facto Python GUI framework, it's now no longer without grievance. One terrific grievance is that GUIs constructed with Tkinter appearance outdated. If you need a shiny, cutting-edge interface, then Tkinter won't be what you're searching for. However, Tkinter is light-weight and comparatively painless to apply as compared to different frameworks. This makes it a compelling preference for constructing GUI programs in Python, especially for programs wherein a cutting-edge sheen is unnecessary, and the pinnacle precedence is to construct something that's purposeful and cross-platform quickly.

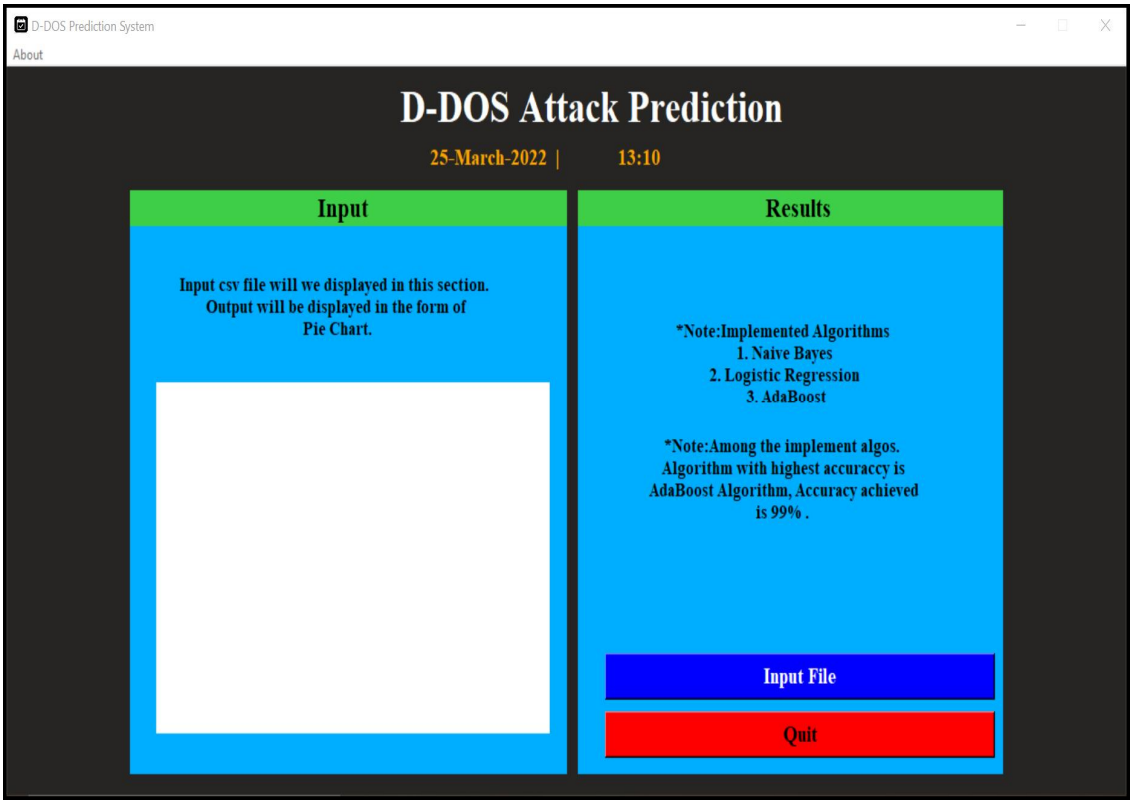


Fig. 4 GUI

G. Metrics for Model Testing

1) *True positive rate*: is the proportion of illustrations that were named as class DDoS, amongst all samples i.e., indicated mathematically as: *True positive rate* = $T_P/(F_N+T_N)$
2) *False positive rate:* is the proportion of illustrations that were identified as class benign, amongst all samples which were not a part of class X i.e., indicated mathematically as: *False positive rate* = $F_P/(F_P+T_N)$
3) *Precision:* positive predictive value is another term for precision. It is the proportion of true positives by the classifier to the aggregate number of positive instances in the experiment i.e., indicated mathematically as: *Precision* = $T_P/(T_P+F_P)$
4) *Recall:* also known as sensitivity or true positive rate. It is the number of true positives by the total number of possible instances i.e., indicated mathematically as: *Recall* = $T_P/(T_P+F_N)$
5) *Accuracy Score:* The most common and widely used parameter to evaluate a model and derive output results is an accuracy score. To find this accuracy score; the sklearn library of Python is used. The score so obtained from this algorithm further varies depending on the algorithms used. Accuracy score = (TP+TN)/ (TP+TN+FP+FN)
6) *ROC Curve:* the full form of this parameter metric stands for Receiver Operating Characteristic and this metric is used to derive a plot based on the results so obtained from a model. It is responsible to generate and plot rates such as; true positives and false positives. This classifier however has the capability to differentiate between different classes and further generates an ROC Curve. The area under this curve is known as Area Under Curve (AUC) and can be generated using the AUC score. Once this graph related data is generated an idea is planned of how the model performs in the later stages.
7) *Classification Report:* Another parameter metric used for output evaluation is the classification report. This report proves to be a valid authority for the purpose of evaluation in the machine learning algorithm. The library that is used in the process of classification is the most commonly used library of Python; sklearn. The usage of this library makes the mathematical calculations to be executed in an effective manner. The classification report further makes use of various algorithms depending on the model used. The classification report consists of parameters such as Precision, Recall, F1 score, Support, etc. The calculation of this parameter is meant to be a ratio of correctly predicted positive to the total predicted positive sample.

IV. EXPERIMENTAL ANALYSIS

A. Naïve Bayes Algorithm

1) Accuracy & Classification Report:

```
Naive_Bayes:Confusion Matrix:  [[21680 17371]
 [   76 51157]]
Naive_Bayes:Accuracy :  80.67542421691552

Figures now render in the Plots pane by default. To make them also a
Plotting" under the Plots pane options menu.

AUC: 0.79
              precision    recall  f1-score   support

         Yes       1.00      0.56      0.71     39051
          No       0.75      1.00      0.85     51233

    accuracy                           0.81     90284
   macro avg       0.87      0.78      0.78     90284
weighted avg       0.85      0.81      0.79     90284
```

Fig. 5 Accuracy

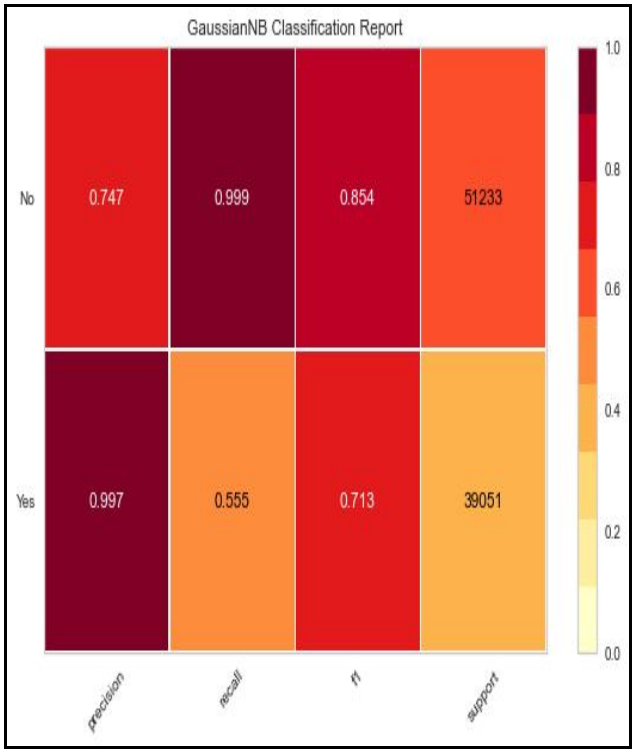


Fig. 6 Classification Report

2) Confusion Matrix & ROC Curve

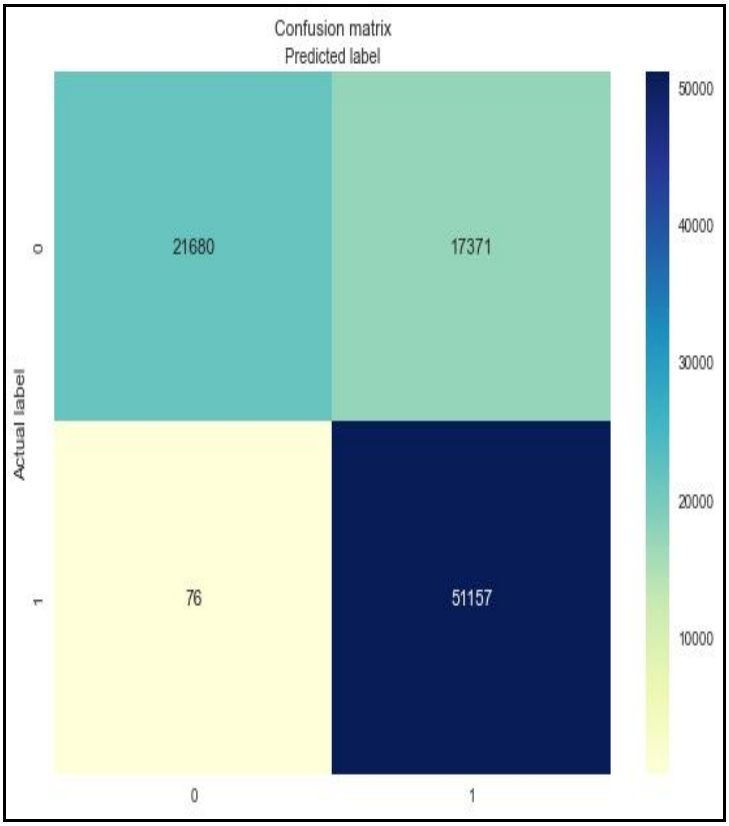


Fig. 7 Confusion Matrix

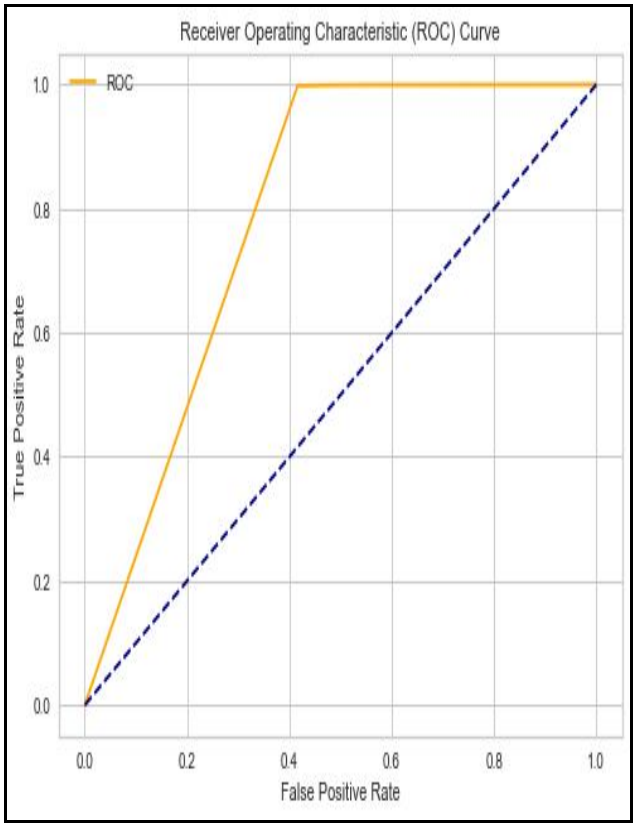


Fig. 8 ROC Curve

B. Logistic Regression Algorithm

1) Accuracy & Classification Report

```
LogisticRegression:Confusion Matrix:  [[34173  4878]
 [  513 50720]]
LogisticRegression:Accuracy :  94.02884231979088
AUC: 0.98
              precision    recall  f1-score   support

         Yes       0.99      0.88      0.93     39051
          No       0.91      0.99      0.95     51233

    accuracy                           0.94     90284
   macro avg       0.95      0.93      0.94     90284
weighted avg       0.94      0.94      0.94     90284
```

Fig. 9 Accuracy

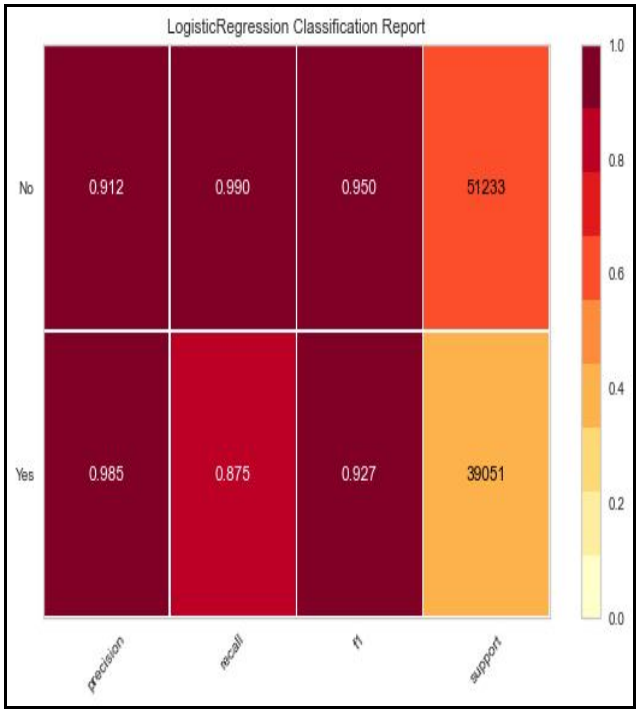


Fig. 10 Classification Report

2) Confusion Matrix & ROC Curve

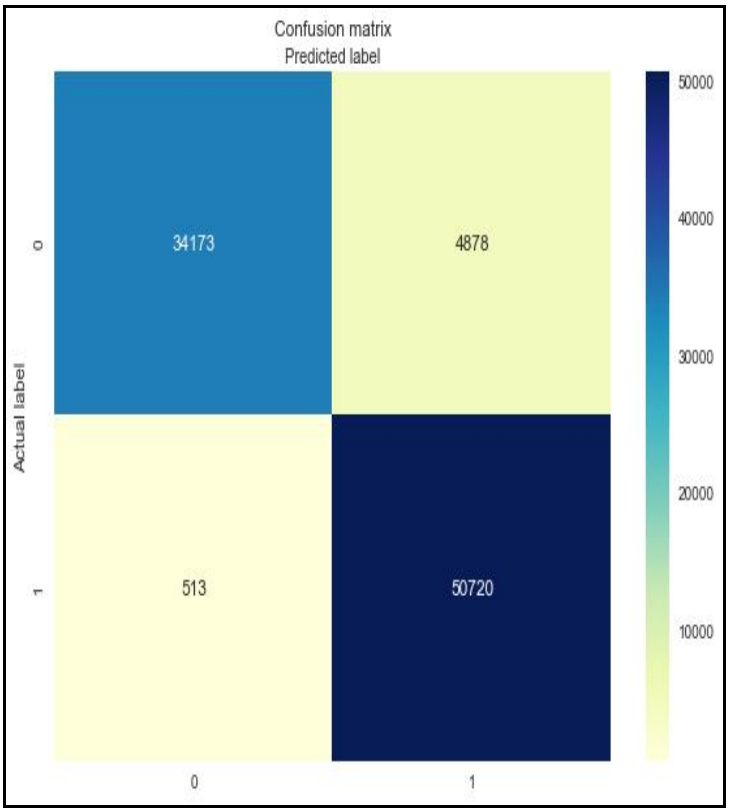


Fig. 11 Confusion Matrix

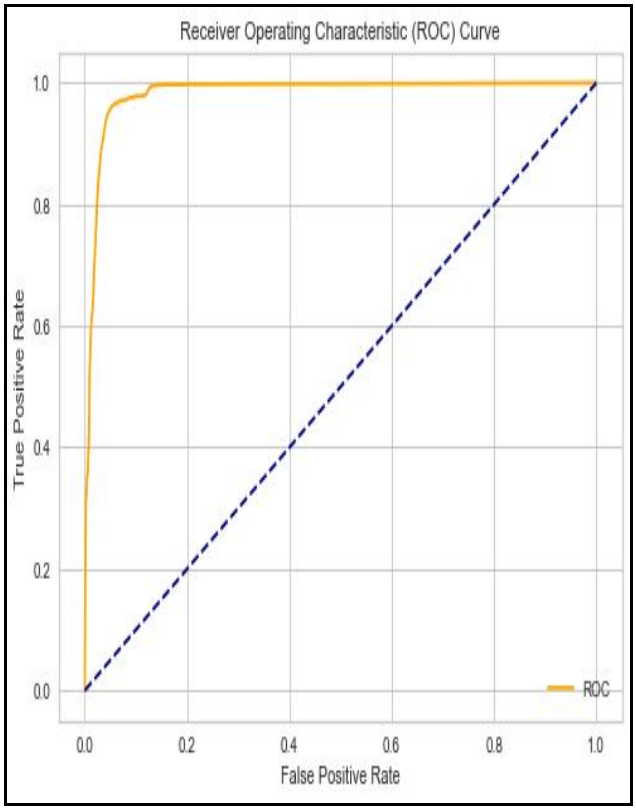


Fig. 12 ROC Curve

C. Ada Boost Algorithm

1) Accuracy & Classification Report

```
Ada Boost:Confusion Matrix:  [[39050     1]
 [   17 51216]]
Ada Boost:Accuracy :  99.98006291258696
AUC: 1.00
              precision    recall  f1-score   support

         Yes       1.00      1.00      1.00     39051
          No       1.00      1.00      1.00     51233

    accuracy                           1.00     90284
   macro avg       1.00      1.00      1.00     90284
weighted avg       1.00      1.00      1.00     90284
```

Fig. 13 Accuracy

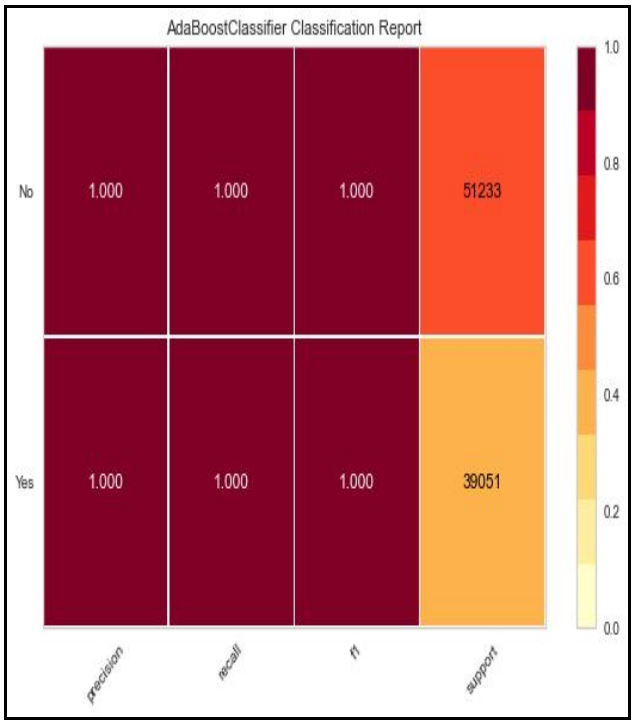


Fig. 14 Classification Report

2) Confusion Matrix & ROC Curve

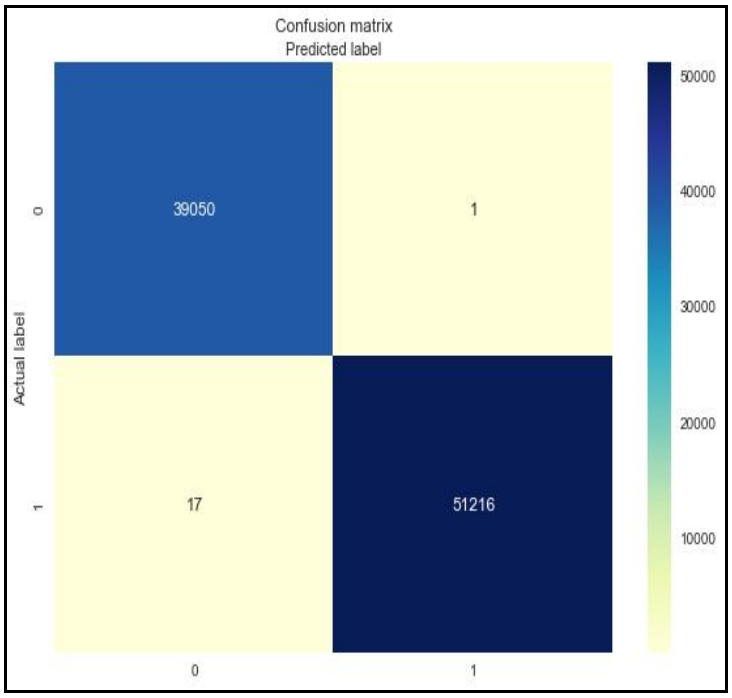


Fig. 15 Confusion Matrix

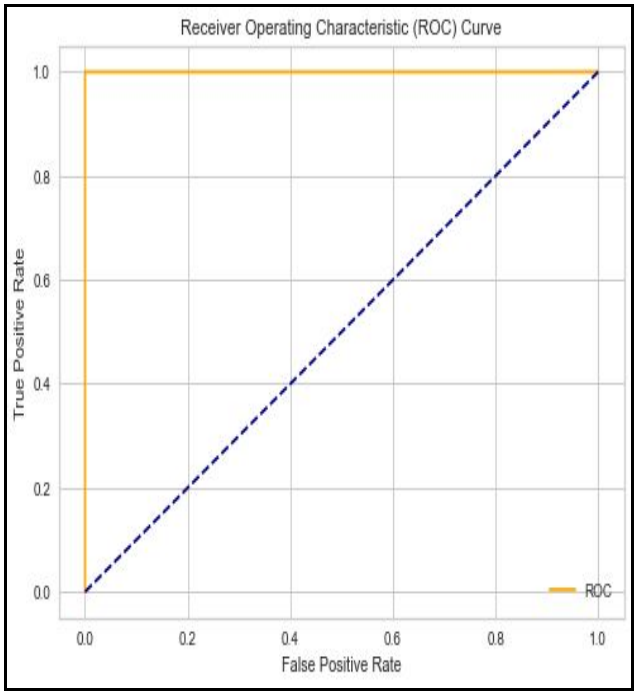


Fig. 16 ROC Curve

V. RESULT

Giving different input csv files and obtaining their outputs

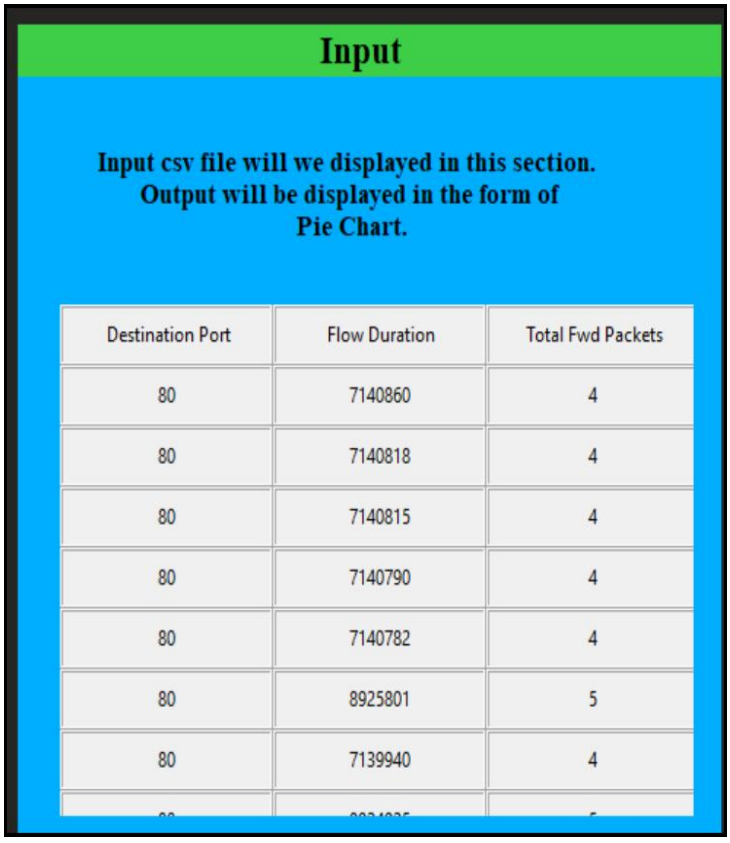


Fig. 17 Input data

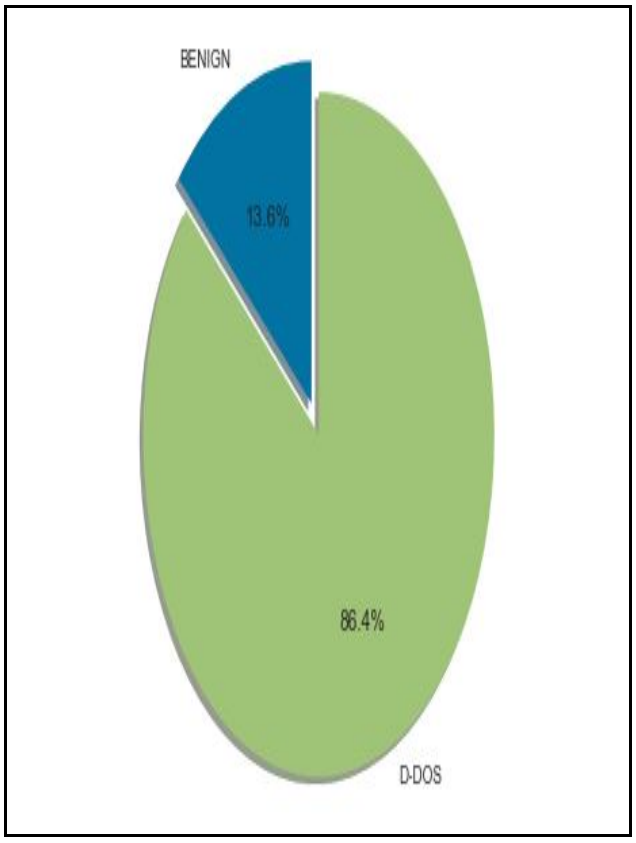


Fig. 18 Output

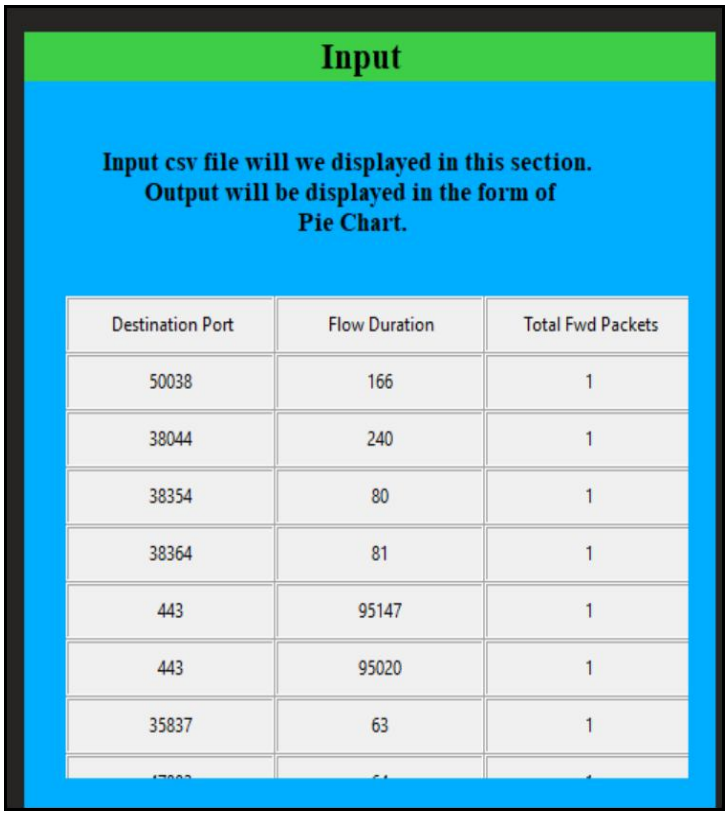


Fig. 19 Input data

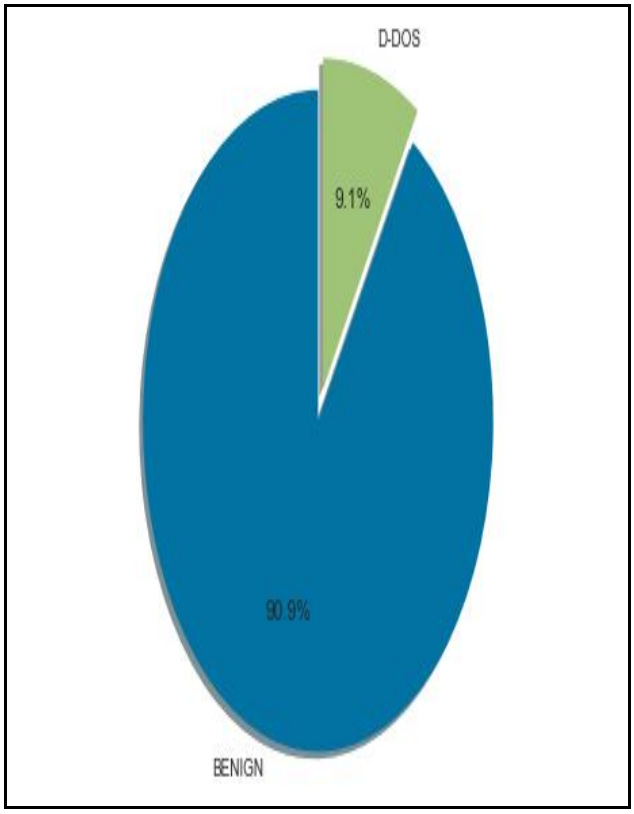


Fig. 20 Output

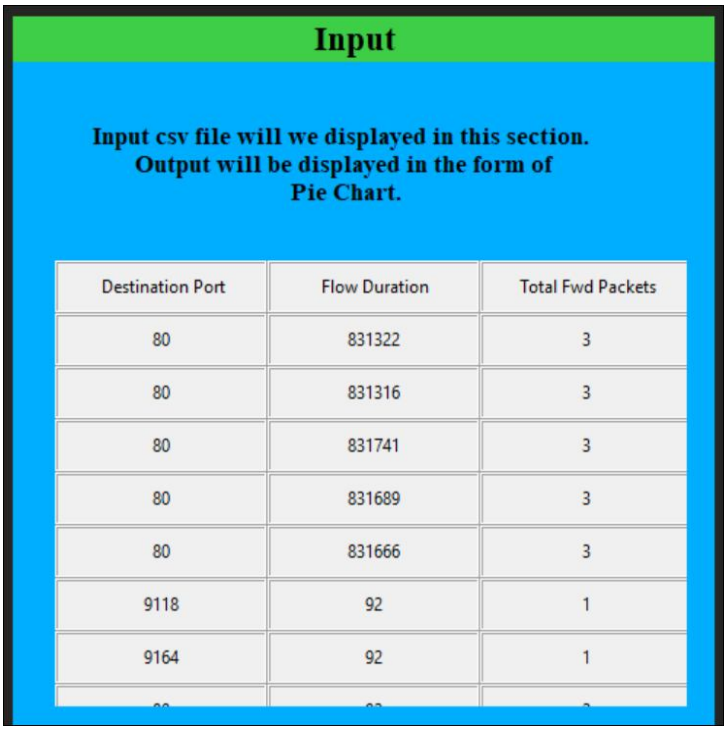


Fig. 21 Input data

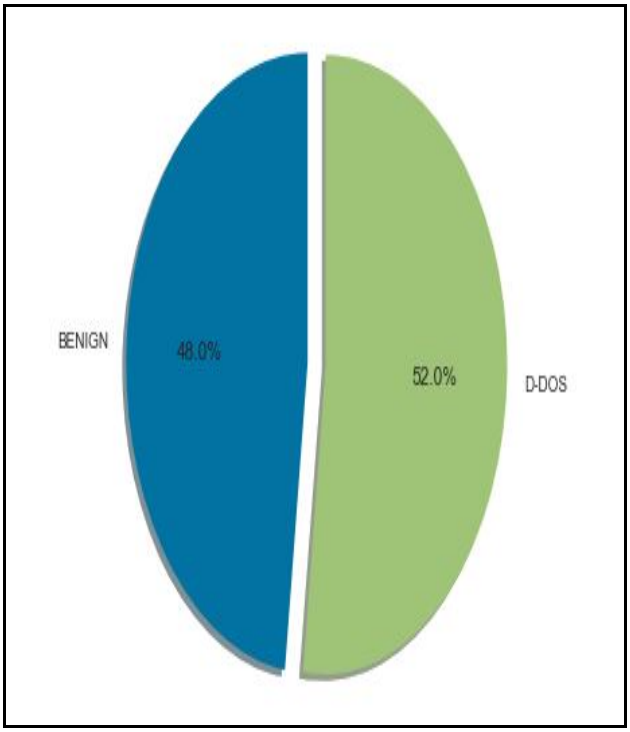


Fig. 22 Output

VI. CONCLUSION

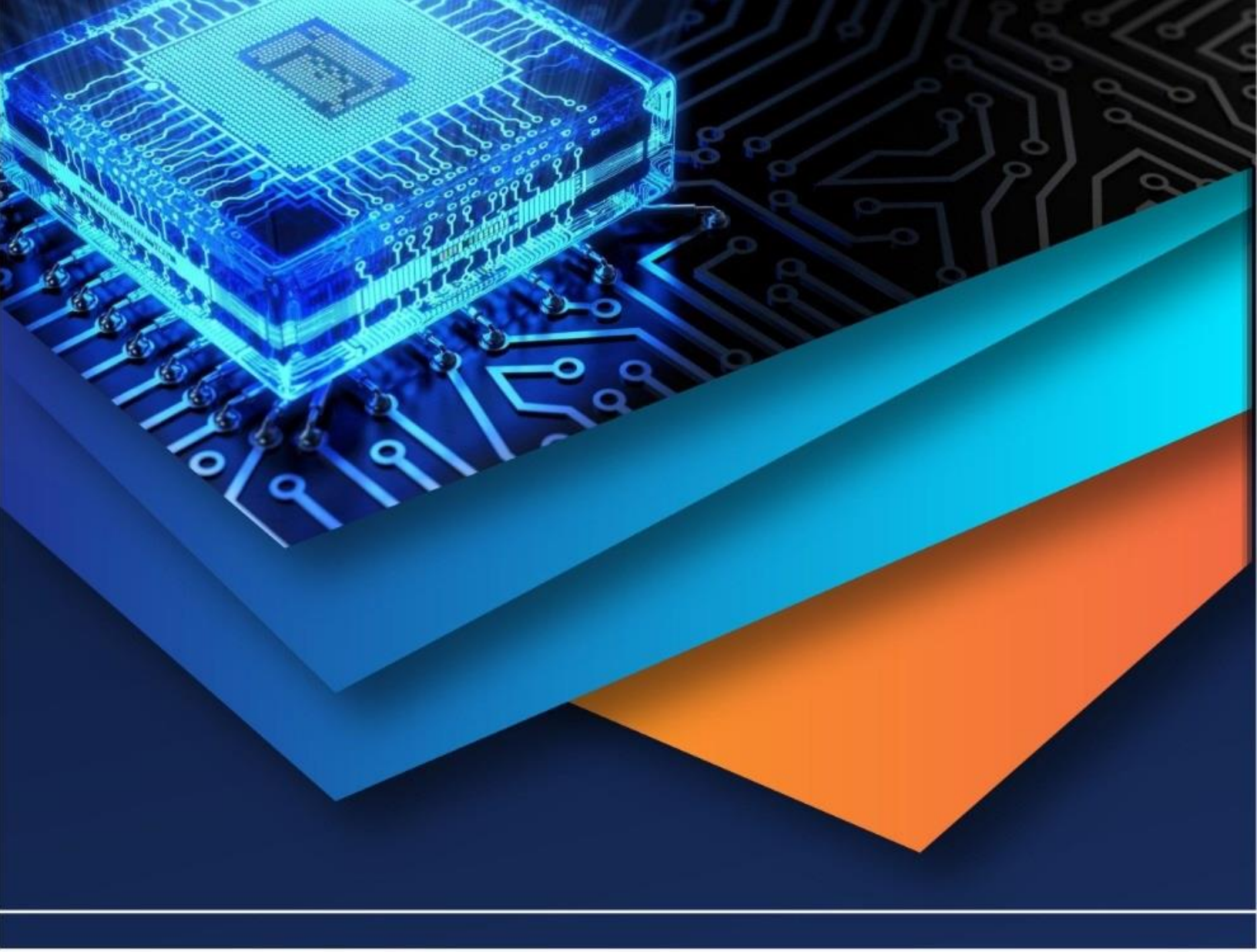
The presented thesis aims to focus on the challenges associated with respective attacks. Since DDoS is considered to be a major threat to computing devices, developing an intrusion detection system, proved to maintain the security of confidential files. However, the existing techniques are still not intentionally built to bring down the malicious attacks taking place. Hence, the goal of the study revolved around investigating the attacks and establishing a co-relation between model performances and design specifications.

VII. FUTURE WORK

In order to magnify the amount of data generated in real time and simultaneously achieve low latency, the concept of an XGBOOST algorithm, can be implemented. This concept is however considered to be a clustering computing framework. The point of convergence here is how these fundamentals shall help in reaching towards better metric parameters. Also, as the number and kind of attacks that are rapidly increasing; attempting to search for a zero-day attack based labelled data is difficult to develop and generate. Hence, this work can further be considered for future work.

REFERENCES

[1] Mahjabin, T.; Xiao, Y.; Sun, G.; Jiang, W. A survey of distributed denial-of-service attack, prevention, and mitigation techniques. Int. J. Distrib. Sens. Netw. 2017, 13. [CrossRef].
[2] Genie-Networks. DDoS Attack Statistics and Trends Report for 2020. 2021. Available online: https://www.genie-networks.com/ gnnews/ddos-attack-statistics-and-trends-report-for-h1-2020/ (accessed on 6 May 2021).
[3] Priya, S.S.; Sivaram, M.; Yuvaraj, D.; Jayanthiladevi, A. Machine learning based DDoS detection. In Proceedings of the 2020 International Conference on Emerging Smart Computing and Informatics, Pune, India, 12–14 March 2020; IEEE: Piscataway Township, NJ, USA, 2020; pp. 234–237.
[4] Pouyanfar, S.; Sadiq, S.; Yan, Y.; Tian, H.; Tao, Y.; Reyes, M.P.; Shyu, M.; Chen, S.; Iyengar, S.S. A survey on deep learning: Algorithms, techniques, and applications. ACM Comput. Surv. 2018, 51, 1–36. [CrossRef]
[5] Lucian Constantin. DDoS Attack against Spamhaus was Reportedly the Largest in History. http://features.techworld.com/security/3437607/ddos-attackagainst-spamhaus-was-reportedly-the-largest-in-history/, 2013.
[6] Vyas Sekar, Nick G Duffield, Oliver Spatscheck, Jacobus E van der Merwe, and Hui Zhang. LADS:Large-Scale Automated DDoS Detection System. In USENIX Annual Technical Conference, General Track, pages 171–184, 2006.
[7] Jelena Mirkovic and Peter Reiher. D-WARD: A Source-End Defense against Flooding Denial-of-Service Attacks. IEEE Transactions on Dependable and Secure Computing, 2(3):216–232, 2005.
[8] Roshan Thomas, Brian Mark, Tommy Johnson, and James Croall. NetBouncer: Client-Legitimacy-based High-Performance DDoS Filtering. In Proceedings of the 2003 DARPA Information Survivability Conference and Exposition, volume 1, pages 14–25. IEEE, 2003. “PDCA12-70 data sheet,” Opto Speed SA, Mezzovico, Switzerland.
[9] Haiqin Liu and Min Sik Kim. Real-time Detection of Stealthy DDoS Attacks using Time-series Decomposition. In IEEE International Conference onCommunications (ICC), 2010, pages 1–6. IEEE, 2010.
[10] Jer´ ome Franc¸ois, Issam Aib, and Raouf Boutaba. FireCol: A Collab- ˆ orative Protection Network for the Detection of Flooding DDoS attacks. IEEE/ACM Transactions on Networking (TON), 20(6):1828–1841, 2012.
[11] Dayanandam, G.; Reddy, E.S.; Babu, D.B. Regression algorithms for efficient detection and prediction of DDoS attacks. In Proceedings of the 2017 3rd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Tumkur, India, 21–23 December 2017; pp. 215–219.
[12] Do, C.T., Tran, N.H., Hong, C., Kamhoua, C.A., Kwiat, K.A., Blasch, E., Ren, S., Pissinou, N. and Iyengar, S.S., 2017. Game theory for cyber security and privacy. ACM Computing Surveys (CSUR), 50(2), pp.1-37.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY