IJITCE

ISSN 2347- 3657

International Journal of
Information Technology & Computer Engineering

Email : ijitce.editor@gmail.com or editor@ijitce.com

DISTRIBUTED DENIAL OF SERVICE ATTACK DETECTION USING RANDOM FOREST ALGORITHM IN MACHINE LEARNING

1. Mr. Shaik Anjimoon , Assistant Professor , Department of Computer Science and Information Technology, Institute Of Aeronautical Engineering , Hyderabad, Telangana, India, shaik.anjimoon@iare.ac.in

2. Md Abdul Rahman , Department Of Computer Science and Information Technology, Institute of Aeronautical Engineering Hyderabad, Telangana, India. mohdmohi80@gmail.com

3. Edulakanti Vignesh , Department of Computer Science and Information Technology, Institute of Aeronautical Engineering, Hyderabad, Telangana, India, edulakantivignesh@gmail.com

4. Chinthamalla Sampath, Department of Computer Science and Information Technology, Institute of Aeronautical Engineering, Hyderabad, Telangana, India, samery6281@gmail.com

Abstract: The Internet has changed the world in the last few decades, but its growth has also brought about many online dangers, most notably Distributed Denial of Service (DDoS) attacks. These attacks stop servers from working by flooding them with strange traffic, which can slow them down or even cause the system to crash. In today's digitally dependent world, fighting this threat is very important. This project uses advanced machine learning (ML) methods to find DDoS attacks quickly, which are becoming a bigger problem. Because DDoS attacks change all the time, there isn't just one answer that works. Classification methods like Decision Trees, Random Forest, and KNN are used to tell the difference between regular traffic and harmful behavior after a lot of study. The main goal of the study is to create a strong DDoS monitoring system that uses machine learning techniques to look at trends in network data. The system tries to quickly spot possible DDoS attacks by learning from past data and tracking in real time. To find DDoS attack trends in network data, ensemble learning, especially the Random Forest method, is used. By using various decision trees, this method provides a strong defense against DDoS threats.

Index Terms: Distributed denial of service, Machine learning, Random Forest, Decision Tree, Intrusion Detection System.

1. INTRODUCTION

In this complicated digital world, connection is the very fabric of modern life. It makes it easier to communicate, do business, and do many other important things. But because everything is linked, we are also open to many online dangers. Distributed Denial of Service (DDoS) attacks are one of the most common ones. DDoS attacks, which are simply "an attempt to make a targeted system, like a website or app, unavailable to legitimate end users" [11], are a big problem for the safety and stability of digital systems around the world.

DDoS attacks have been around since the early days of the internet. Some of the most famous ones happened in the late 1990s [12]. Over the years, these attacks have become more complex and large. They are done for a wide range of reasons, from hacktivism to bribery and more. Hacktivist groups, in particular, have used DDoS attacks as a way to protest or be politically active online, hoping to stop organizations or governments from doing their jobs and say something about what they see as wrong [13].

When it comes to protecting against these growing threats, old ways of doing things have not worked. Conventional security measures like firewalls, intrusion detection systems, and others often aren't able to handle DDoS attacks because they are spread out and change all the time. So, we need new methods right away that can change with the times to keep up with bad players' changing strategies.

Machine learning (ML) allows computers to learn from data and improve without being programmed. The accuracy of predictions and choices is due to ML algorithms' ability to discover patterns, connections, and trends in datasets [14].

One of the greatest DDoS detection machine learning approaches is Random Forest. Random Forest uses several decision trees, each learnt on a separate set of data and attributes [15]. This ensemble technique reduces overfitting and enhances model stability and forecast accuracy by combining information from numerous models.

What makes Random Forest unique is the clever way it chooses random features at each split point while building the tree. The method decorates the individual trees by only looking at a subset of features at each node. This keeps them from just remembering the training data and encourages generalization [16]. This function is especially useful for finding DDoS attacks because it's important to be able to pick out small trends in the noise of network data.

Also, in a Random Forest, the final forecast is made by voting for classification tasks or average for regression tasks. This makes for a strong and reliable model that is less affected by errors and changes in the data [17]. Random Forest is the best choice for real-world tasks where data is confusing or incomplete, like finding DDoS attacks, because it is so resilient.

We are starting a deep look into the area where machine learning and cybersecurity meet in this study. Our goal is to make our digital defenses stronger against the growing threat of DDoS attacks. We want to create a strong and aggressive defense system that can find and stop DDoS attacks in real time by using the power of Random Forest in the field of machine learning. We want to show that this method works by using data research and experiments. This will help with the current efforts to protect our digital infrastructure.

2. LITERATURE SURVEY

Cybersecurity experts have been looking into a lot of different methods and techniques to stop Distributed Denial of Service (DDoS) threats. This literature review looks at some of the most important studies, methods, and tools that are used to find and stop DDoS attacks, with a focus on machine learning techniques.

A group of researchers led by C. Zhang suggested a way to find and block low-rate DDoS attacks based on flow level. The research stresses how important flow-based analysis is for finding low-rate DDoS attacks that are hard for regular methods to pick up. The suggested way looks at flow-level features like the rate at which packets arrive and the time between arrivals to tell the difference between regular and bad traffic.

An important study by P.N. Jadhav and B.M. Patil [2] introduces the Optimal Objective Entropy approach for discovering low-rate DDoS assaults. The approach employs entropy to quantify packet and traffic randomness. The recommended approach rapidly and reliably detects low-rate DDoS assaults by detecting traffic entropy variations.

P. Du and S. Abe [3] suggested a different way to find DoS and DDoS attacks by using the entropy of IP packet sizes. The method looks for strange trends that could be attack traffic by looking at the entropy of packet size distributions. This method based on entropy is a quick and easy way to find DDoS attacks at the network level.

Cloud computing and software-defined networking have made DDoS attack security harder and more intriguing. Wang Bing et al. discuss how cloud computing and SDN effect DDoS protection systems [4]. Cloud defenses must adapt to large-scale DDoS attacks, according to the research.

By looking at current defenses against DDoS flooding attacks, Zargar et al. [5] give a full picture of many different ways to stop them. The study looks at many different methods, such as internet blocking, rate limiting, and methods that look for unusual activity. The study gives useful information about good DDoS protection tactics by looking at the pros and cons of each method.

New machine learning approaches may help detect and analyze DDoS assaults. Irfan Sofi et al. investigate how machine learning might identify new DDoS assaults [6]. Decision Trees, Support Vector Machines, and Neural Networks are investigated for DDoS attack trend detection.

Network-mimicking DDoS attacks can also be stopped with detecting methods based on chaos theory. The work of Ashley Chonka et al. [7] describes a way to find network-imitating DDoS attacks that is based on chaos theory. The suggested method looks at how network traffic behaves randomly to accurately tell the difference between valid and attack traffic.

It can be hard to keep an eye on application-layer DDoS attacks on famous websites because web applications and protocols are so complicated. In their paper [8], Yi Xie et al. describe a way to keep an eye on application-layer DDoS strikes that are aimed at popular websites. For finding and stopping application-layer threats, the study stresses how important it is to use real-time tracking and anomaly detection methods.

Using network self-similarity to find DDoS attacks is another way to protect yourself before they happen. Network self-similarity research is what Y. Xiang et al. [9] suggest as a way to find DDoS attacks. The method measures the self-similarity of network data to look for changes from normal behavior that could be signs of DDoS attacks.

It is very important to be able to quickly find and stop DDoS and TCP flood attacks in cloud settings. Aqeel Sahi et al. [10] show a good way to find and stop DDoS TCP flood attacks in cloud settings. The system uses network traffic analysis and machine learning to find and stop TCP flood attacks in real time.

To sum up, the literature review shows the wide range of approaches and tools that are used to find and stop DDoS attacks. Researchers keep coming up with new ways to make networks safer and more resistant to new cyber threats. These include flow-based analysis, entropy-based measures, and machine learning algorithms.

3. METHODOLOGY

a) Proposed Work:

We want to use the Random Forest method in our suggested work as a powerful way to find and stop Distributed Denial of Service (DDoS) attacks. We expect to be able to tell the difference between normal network traffic and harmful intrusions very accurately by using its ability to handle big datasets and find complex patterns. On both the academic and practical levels, this study is likely to make a big difference. In theory, we want to give new information about how Random Forest can be used to find DDoS attacks. This could lead to new discoveries in feature extraction and model training. In real life, our goal is to create a real and effective way to protect against DDoS attacks, which will make the internet safer and more robust. Because DDoS attacks are so disrupting and real-time discovery and prevention are hard, our suggested solution has a lot of potential to protect network security and lower the risks that cyber threats could bring.

b) System Architecture:

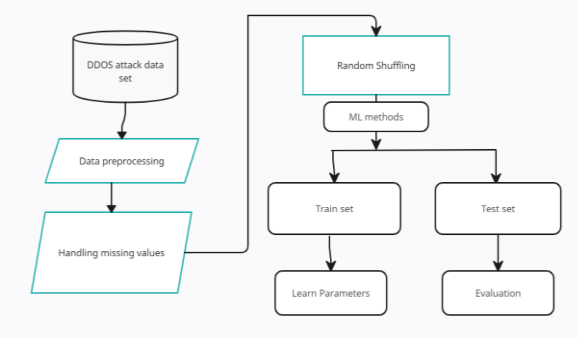


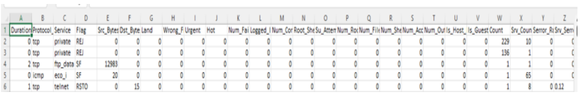
Fig 1 Proposed Architecture

"DDoS Attack Detection Using Random Forest in Machine Learning" starts with gathering a variety of datasets that include network traffic data. This makes sure that both regular and attack situations are represented. Once the data has been cleaned up and any missing or unusual values have been taken care of, it is labeled as either normal or a DDoS attack based on trends that have already been found. Information gain and other feature selection methods help find the best set of useful features. Using the preprocessed and feature-selected dataset to train a Random Forest model is what the system is all about.

Ensemble learning is used to improve accuracy and stability. To get the best results from hyperparameters, they are carefully tested using validation datasets and measures such as accuracy and recall. The Random Forest model is put into production after it has been trained, and it works perfectly with the current network security infrastructure. Continuous tracking makes sure that it keeps working, and changes are made on a regular basis to keep up with changing DDoS attack trends. The architecture map shows all the parts of the system clearly, making sure that a complete plan is made to improve network security.

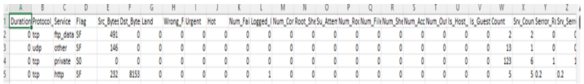
c) Dataset Collection:

Either the CCIDS2017 dataset or the NSL-KDD dataset is used to train and test the algorithm for finding DDoS attacks. These files have important information in them, like the target port, packet properties like length and header length, flow time, and different TCP flags. Attack marks are added to the dataset to show if the traffic is from a DDoS attack. DDoS attacks can be identified by things like the size of the packets, the length of the flow, and the number of forward and backward packets. The dataset is split into separate groups for training and testing. This makes sure that the algorithm is trained and tested on a wide range of examples of normal and attack traffic patterns.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	Duration	Protocol_	Service	Flag	Src_Bytes	Dst_Byte	Land	Wrong_F	Urgent	Hot	Num_Fai	Logged_I	Num_Cor	Root_She	Su_Atten	Num_Roc	Num_File	Num_She	Num_Acc	Num_Out	Is_Host_	Is_Guest	Count	Srv_Coun	Serror_Ra	Srv_Serr
2	0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	229	10	0	(
3	0	tcp	private	REJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	136	1	0	(
4	2	tcp	ftp_data	SF	12983	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	(
5	0	icmp	eco_i	SF	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	65	0	(
6	1	tcp	telnet	RSTO	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	8	0	0.12

Fig 2 Train Dataset



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	Duration	Protocol_	Service	Flag	Src_Bytes	Dst_Byte	Land	Wrong_F	Urgent	Hot	Num_Fai	Logged_I	Num_Cor	Root_She	Su_Atten	Num_Roc	Num_File	Num_She	Num_Acc	Num_Out	Is_Host_	Is_Guest	Count	Srv_Coun	Serror_Ra	Srv_Serr
2	0	tcp	ftp_data	SF	491	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	0	(
3	0	udp	other	SF	146	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	1	0	(
4	0	tcp	private	S0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	123	6	1	1
5	0	tcp	http	SF	232	8153	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	5	5	0.2	0.2

Fig 3 Test Dataset

d) Data Processing:

Two very important steps are needed to handle data for DDoS attack detection: collecting the data and preparing it. At first, a large sample is gathered that includes both normal network traffic and DDoS strikes. To make sure the method works in all situations, this collection should be very big and include a lot of different attack scenarios and real-world network conditions.

Preprocessing steps are done on the information after it has been collected to improve its quality and reliability. This includes features for cleaning and organizing data to make the code consistent and get rid of any errors. It is very important to deal with missing values and errors to keep the research from being biased and to make sure that the model is trained correctly. During this step, methods like imputation, which fills in empty values, and statistical methods or grouping algorithms, which find and treat outliers, are used. By carefully handling the data, we get a sample that is perfect for teaching machine learning models. This makes it possible to find DDoS attacks even when network traffic trends change.

e) Feature Extraction:

Finding important features that successfully catch the traits of network data that are telling of DDoS attacks is a key part of detecting DDoS attacks. This is called "feature extraction." This process looks at things like packet size, packet rate, traffic flow, and different TCP flags. Domain understanding and data analysis are very important for getting useful traits that help with the discovery process. To find traits that can tell the difference between regular and attack traffic, methods like association analysis, principal component analysis (PCA), and information gain are used. The feature extraction phase uses domain knowledge and statistical insights to make sure that the chosen features accurately reflect the dataset's underlying patterns. This lets the machine learning algorithms tell the difference between normal traffic and DDoS attack traffic.

f) Model Training:

During the model training phase, the Random Forest method is taught how to use the set of training data. The algorithm is shown examples of both normal traffic and DDoS attack traffic. This lets it learn the patterns and traits that are unique to each type. Random Forest is made up of many decision trees working together as an ensemble. This helps the model understand the complex relationships in the data and make it easier to generalize. The Random Forest algorithm changes its internal settings over and over again during training to get better at telling the difference between regular and attack traffic and reduce the number of wrong predictions. It is through this training process that the Random Forest model learns to recognize the signs of DDoS attacks, which sets the stage for accurate spotting during the operating phase.

g) Model Evaluation:

During the model review step, the testing dataset is used to check how well the learned Random Forest model did. To measure how well the model can tell the difference between regular traffic and DDoS attack traffic, important metrics like accuracy, precision, recall, and F1 score are calculated. Accuracy shows how accurate the model's predictions are generally, while accuracy shows what percentage of DDoS attack cases were correctly identified out of all DDoS attack cases that were identified. On the other hand, recall tests how well the model can correctly identify all DDoS attack cases out of all the real DDoS attacks in the dataset. The F1 score is a fair measure of how well the model worked because it finds the harmonic mean of accuracy and memory. Iteratively, changes can be made to the model's settings based on the results of tests to improve speed and make the model better at finding DDoS attacks.

h) Algorithms:

Decision Tree:

Machine learning methods called decision trees can be used for both classification and regression tasks. For making decisions, they are used in many areas, such as banking, healthcare, and marketing. DecisionTreeClassifier is a class in machine learning tools like scikit-learn that makes it easier to use decision trees to sort information into groups.

A decision tree is a structured model that shows how people make decisions. It divides data into groups based on the values of characteristics, which makes it simple to understand and see. Decision trees are very good at classifying or predicting results because they divide the feature space over and over again. They can handle both number and categorical data well and are especially good at helping you figure out which factors are most important in making decisions.

Random Forest:

Random forests are often used for classification and regression jobs that need to be accurate and reliable. In fields like biology, scam detection, and suggestion systems, they are used. Python tools like scikit-learn are used to make Random Forest work.

Random Forest is a type of ensemble learning that uses the strengths of many decision trees to make predictions more accurate and cut down on overfitting. Random Forest finds a wide range of trends in data by building many decision trees from random parts of the training data and features. When you vote or average the predictions of these trees together, you get a more stable and accurate model. Random Forest is often used in machine learning applications because it is good at dealing with large amounts of data and traits that are related in complicated ways.

4. EXPERIMENTAL RESULTS

Accuracy: How well a test can tell the difference between sick and healthy people is called its accuracy. To get an idea of how accurate a test is, we should figure out what percentage of cases are true positives and true negatives. In terms of math, this can be written as

$$\text{Accuracy} = \text{TP} + \text{TN TP} + \text{TN} + \text{FP} + \text{FN}.$$

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

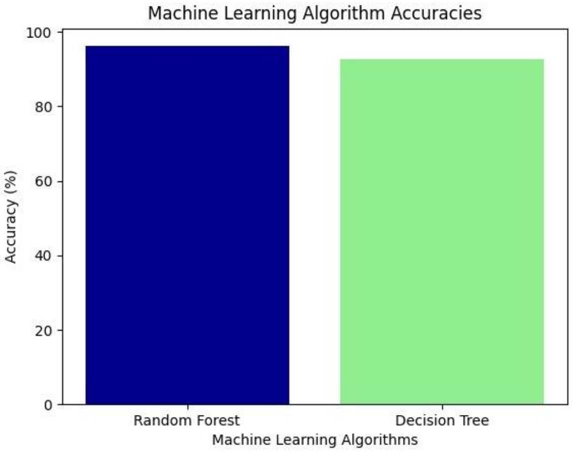


Fig 4 Accuracy Comparison Graph

F1-Score: The F1 score is a way to rate the correctness of a machine learning model. It takes a model's accuracy and memory scores and adds them together. The accuracy measure counts how many times, across the whole collection, a model made a correct guess.

$$\textbf{F1 Score} = \frac{2}{\left(\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}\right)}$$

$$\textbf{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

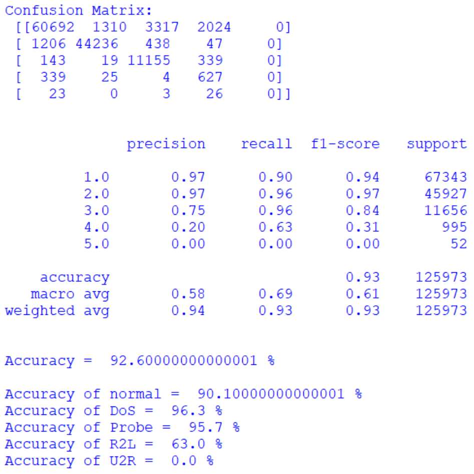
Precision: Precision is the percentage of correctly classified cases or samples compared to those that were correctly classified as hits. So, here is the method to figure out the precision:

Precision = True positives/ (True positives + False positives) = TP/ (TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall: In machine learning, recall is a parameter that shows how well a model can find all the important cases of a certain class. It indicates how effectively a model captures class cases. Divide the number of accurately anticipated positive observations by the total genuine positives.

$$Recall = \frac{TP}{TP + FN}$$



```
Confusion Matrix:
 [[60692  1310  3317  2024     0]
 [ 1206 44236   438    47     0]
 [  143    19 11155   339     0]
 [  339    25     4   627     0]
 [   23     0     3    26     0]]

              precision    recall  f1-score   support

         1.0       0.97      0.90      0.94     67343
         2.0       0.97      0.96      0.97     45927
         3.0       0.75      0.96      0.84     11656
         4.0       0.20      0.63      0.31       995
         5.0       0.00      0.00      0.00        52

    accuracy                           0.93    125973
   macro avg       0.58      0.69      0.61    125973
weighted avg       0.94      0.93      0.93    125973


Accuracy =  92.60000000000001 %

Accuracy of normal =  90.10000000000001 %
Accuracy of DoS =  96.3 %
Accuracy of Probe =  95.7 %
Accuracy of R2L =  63.0 %
Accuracy of U2R =  0.0 %
```

Fig 5 Comparison Graphs

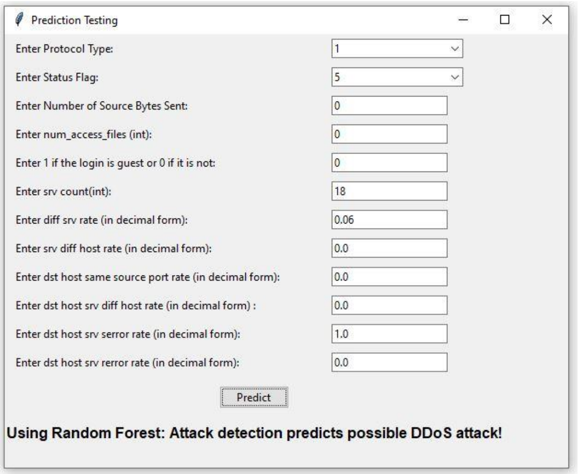
Algorithm Performance with Comparison:

Decision Tree and the Random Forest algorithm are compared in terms of how accurate they are and how fast they can predict events. The Random Forest algorithm is better at detecting DDOS than the Decision Tree algorithm.

Algorithm	**Accuracy %**
Random Forest	96.3999
Decision Tree	92.6000

We will use the **Random Forest** method to help us guess when the DDOS attack will happen because it is more accurate.

Python's Tkinter package was used to make the DDOS attack detection app, which will use the random forest method in the suggested model to find the attack. Figure 6: Application to Detect DDOS shows how the application works. The picture not only showed the user interface, but it also used information from the administrator to guess whether an attack was likely and showed a message if one was found.



Prediction Testing

Enter Protocol Type: 1

Enter Status Flag: 5

Enter Number of Source Bytes Sent: 0

Enter num_access_files (int): 0

Enter 1 if the login is guest or 0 if it is not: 0

Enter srv count(int): 18

Enter diff srv rate (in decimal form): 0.06

Enter srv diff host rate (in decimal form): 0.0

Enter dst host same source port rate (in decimal form): 0.0

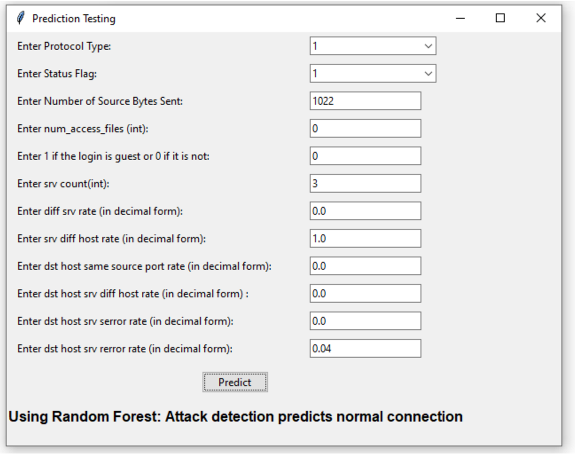
Enter dst host srv diff host rate (in decimal form) : 0.0

Enter dst host srv serror rate (in decimal form): 1.0

Enter dst host srv rerror rate (in decimal form): 0.0

Using Random Forest: Attack detection predicts possible DDoS attack!

Fig 6 Application to detect DDOS Attack.



Prediction Testing

Enter Protocol Type: 1

Enter Status Flag: 1

Enter Number of Source Bytes Sent: 1022

Enter num_access_files (int): 0

Enter 1 if the login is guest or 0 if it is not: 0

Enter srv count(int): 3

Enter diff srv rate (in decimal form): 0.0

Enter srv diff host rate (in decimal form): 1.0

Enter dst host same source port rate (in decimal form): 0.0

Enter dst host srv diff host rate (in decimal form) : 0.0

Enter dst host srv serror rate (in decimal form): 0.0

Enter dst host srv rerror rate (in decimal form): 0.04

Using Random Forest: Attack detection predicts normal connection

Fig 7 Application to detect DDOS Attack.

5. CONCLUSION

In conclusion, our model's goal is to correctly find DDoS attacks by looking at trends of network traffic and telling the difference between regular traffic and traffic that is part of an attack. The model is made to find different kinds of DDoS attacks, like protocol, application, and massive attacks. Decision Trees and Random Forest are two machine learning methods that we used to sort different types of DDoS attacks, like Smurf, UDP flood, HTTP flood, and Neptune. Random Forest did better than Decision Trees and had the best accuracy rate, so it was chosen to predict DDoS attacks.

After teaching and trying the model, we were able to accurately predict when DDoS attacks would happen. The Random Forest method in machine learning has been used to find DDoS attacks, and the results are looking good. Our model achieved a respectable 96% success rate. This shows how useful it is to use advanced machine learning methods to fight cyber dangers and keep network assets safe from attacks. Our model could improve network security and lessen the damage that DDoS attacks do to digital platforms if it is improved further and used in real-world cybersecurity systems.

6. FUTURE SCOPE

The success of this study makes it possible for more research to be done and for DDoS attack tracking to get better. It is very important to test the model's success on different datasets and network designs to make sure it works in all kinds of situations. By looking into how different machine learning algorithms and ensemble methods can be used together, we can learn more about the best ways to handle certain network situations. In the future, researchers may also look into adding real-time adaptable features that use dynamic learning methods to proactively fight new dangers. It is also important to look into how different combinations of hyperparameters affect model improvement. Working together with people in the industry and cybersecurity experts will make it easier to use the model we've created in real-world situations. This will help us learn more about how well it works in different operating situations and contribute to the ongoing development of DDoS defense tactics.

REFERENCES

[1] C. Zhang, Z. Cai, W. Chen, X. Luo, and J. Yin, “Flow level detection and filtering of low-rate DDoS,” Computer Networks, vol. 56, no. 15, pp. 3417–3431, 2012.

[2] P.N.Jadhav and B. M. Patil, “Low-rate DDOS AttackDetection using Optimal Objective Entropy Method,” International Journal of Computer Applications, vol. 78, no. 3, pp. 33–38, 2013.

[3] P. Du and S. Abe, “IP packet size entropy-based scheme for detection of DoS/DDoS attacks,” IEICE Transaction on Information and Systems, vol. E91-D, no. 5, pp. 1274–1281, 2008.

[4] Wang Bing, Zheng Yao,Lou Wenjing, “DDoS attack protection in the era of cloud computing and software defined networking”.Computer Networks,2015,308—319.

[5] Zargar S T,Joshi J,Tipper D, “A survey of defense mechanisms against distributed denial of service (DDoS)flooding attacks”. IEEE Communications Survey & Tutorials,2013，15(4)：2046—2069.

[6] Irfan Sofi, Amit Mahajan, Vibhakar Mansotra, “Machine Learning Techniques used for the Detection and Analysis of Modern Types of DDoS Attacks”,International Research Journal of Engineering and Technology (IRJET) / 2017.

[7] Ashley Chonka, Jaipal Singh, Wanlei Zhou, “Chaos theory-based detection against network mimicking DDoS attacks,” IEEE Communications Letters Volume: 13, Issue: 9, Sept. 2009.

[8] Yi Xie , Shun-Zheng Yu, “Monitoring the Application-Layer DDoS Attacks for Popular Websites,” IEEE/ACM Transactions on Networking Volume: 17 , Issue: 1 , Feb. 2009.

[9] Y. Xiang ; Y. Lin , W.L. Lei , S.J. Huang “Detecting DDOS attack based on network self-similarity,” IEE Proceedings - Communications Volume: 151 , Issue: 3 , June 2014.

[10] Aqeel Sahi , David Lai , Yan Li , Mohammed Diykh, “An Efficient DDoS TCP Flood Attack Detection and Prevention System in a Cloud Environment,” IEEE Access , Volume: 5, April 2017.

[11] Banitalebi Dehkordi, Mohammad Afsaneh, Reza Soltanaghaei and Farsad Zamani Boroujeni, "The DDoS attacks detection through machine learning and statistical methods in SDN", The Journal of Supercomputing, vol. 77.3, no. 2021, pp. 2383-2415.

[12] Rohan Doshi, Noah Apthorpe and Nick Feamster, "Machine Learning DDoS Detection for Consumer Internet of Things Devices", 2018 IEEE Symposium on Security and Privacy Workshops.

[13] Marwane Zekri et al., "DDoS attack detection using machine learning techniques in cloud

computing environments", 2017 3rd international conference of cloud computing technologies and applications (CloudTech), 2017.

[14] Hoyos Ll, S. Manuel et al., "Distributed denial of service (DDoS) attacks detection using machine learning prototype", Distributed Computing and Artificial Intelligence 13th International Conference, 2016.

[15] Aditya Khamparia et al., "Multi-level framework for anomaly detection in social networking", Library Hi Tech, 2020.

[16] Jelena Mirkovic and Peter Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms", ACM SIGCOMM Computer Communication Review, vol. 34.2, pp. 39-53, 2004.

[17] Tanweer Alam, "A reliable communication framework and its use in the internet of things (IoT)", vol. 10, pp. 450-456, 2018.

[18] D.K. Bhattacharyya and J.K Kalita, DDoS attacks: evolution detection prevention reaction and tolerance, CRC Press, 2016.

[19] K. Sonar and H. Upadhyay, "A survey: DDOS attack on Internet of Things", International Journal of Engineering Research and Development, vol. 10, no. 11, 2014.

[20] Sunny Behal, Krishan Kumar and Monika Sachdeva, "D-FACE: An anomaly-based distributed approach for early detection of DDoS attacks and flash events", Journal of Network and Computer Applications, vol. 111, pp. 49-63, 2018.