ISSN 2278-3091
Volume 10, No.4, July - August 2021
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse271042021.pdf
https://doi.org/10.30534/ijatcse/2021/271042021

An Efficient Method for Detection of DDoS Attacks on the Web Using Deep Learning Algorithms

Amal Al-Harbi[1],Randa Jabeur[2]
[1]Jouf University , Saudi Arabia ,40125980@ju.edu.sa
[2]Jouf University ,Tunisia ,rjabeur@ju.edu.sa

ABSTRACT

Recently, DDoS attacks is the most significant threat in network security. Both industry and academia are currently debating how to detect and protect against DDoS attacks. Many studies are provided to detect these types of attacks. Deep learning techniques are the most suitable and efficient algorithm for categorizing normal and attack data. Hence, a deep neural network approach is proposed in this study to mitigate DDoS attacks effectively. We used a deep learning neural network to identify and classify traffic as benign or one of four different DDoS attacks. We will concentrate on four different DDoS types: Slowloris, Slowhttptest, DDoS Hulk, and GoldenEye. The rest of the paper is organized as follow: Firstly, we introduce the work, Section 2 defines the related works, Section 3 presents the problem statement, Section 4 describes the proposed methodology, Section 5 illustrate the results of the proposed methodology and shows how the proposed methodology outperforms state-of-the-art work and finally Section VI concludes the paper.

Key words: DDoS attacks, deep learning neural network.

1. INTRODUCTION

Since the advent of computing, malware and attacks have existed. However, it was not until the internet's explosive growth that security and digital assets became a major concern. The growing number of computers on the internet creates a new goldmine for those looking to exploit vulnerabilities, making the internet a new liability. New ways for attackers to target network systems and their users emerge as access increases. A distributed denial of service (DDoS) remains the most destructive and serious form of attack due to its potential effects, and the threat continues to increase, necessitating intrusion detection for network protection and defense. The DDoS attack is one in which numerous systems launch DoS attacks on a single system. requests that make it incapable of serving legitimate user requests. It represents one of the most frightening risks that modern businesses face. A successful DoS attack can have far-reaching financial consequences. DDoS attacks are estimated to cost between $20,000 and $40,000 per hour, and nearly 50,000- 2.3$ million every year according to security surveys [3]. This is a huge sum that may put even the most powerful companies under strain.

Organizations around the world witnessed an average of 237 DDoS attacks attempts per month in Q3 2017, according to Corero Network Security (A DDoS defense and mitigation provider), which averages 8 DDoS attacks per day. This was an improvement of 35 percent over Q2 that year and an unprecedented 91 percent rise over Q1.

Software that recognizes and mitigates a DDoS attack is commercially available, but the high cost of this software makes it difficult for small and mid-scale enterprises to afford it. As more organizations and companies in various industries are shifting to a digital transition, malware is increasingly growing, confronting data theft or service interruptions resulting from cyber-attacks on the network or device that affect the customer experience. DDoS attack detection using deep learning algorithms helps in taking action and decrease the consequences of such events.

There is a critical need for faster, automated malware detection, visualization, and reaction. This challenge could be solved via machine learning, which would entail teaching cyber defense systems to detect these attacks.

As many papers focus on classifying traffic as benign or DDoS attacks, we focus in our paper on addressing each type of attack. In this paper, we will present a machine

learning method for detection and classification of DDoS. Furthermore, the majority of studies that have been published to detect and categorize DDoS attacks use datasets from the Canadian Institute of Cybersecurity or perform detection using their data. In our study, we combine the CICIDS2017[20] from the Canadian Institute of Cybersecurity with the data we collected using CICFlowMetre. This improved the accuracy of our classifier. Experimental results will show that this approach can identify and classify DDoS attacks with a high accuracy.

2.RELATED WORKS

The most efficient technique to defend against a DDoS attack is to detect and drop the attack traffic automatically and precisely. Machine learning is essential for efficiently mitigating DDoS attacks .A variety of machine learning algorithms are used for DDoS defense such as Naïve Bayes, Neural Network, Support Vector Machine, Decision Tree, K-Nearest Neighbor.

Different machine learning methods were utilized by Sofi et al. [7] to recognize and analyze modern DDoS attacks. In this paper, they propose “improved RI algorithm" (IRI), which reduces the search area for generating classification rules by eliminating all uninteresting candidate rule-items as the classification model is being built. The fundamental benefit of IRI is that it generates a set of rules that are brief, easy to understand, and simple to implement. They use four machine learning algorithms, including decision trees, naïve Bayes, support vector machines (SVM), and multi-layer perceptron to detect and analyse modern forms of DDoS Attacks. Sharma et al. [8] also conducted a systematic review of machine learning techniques used in dos attack detection.

Perakovicet al. (2016)[24]and Saied et al. (2016)[25]employed an artificial neural network to analyze the results and look at different parameters. Saied et al.(2016)[24]employed an artificial neural network to identify distinct patterns of distinguishing features that differ from legitimate traffic and DDoS attack traffic in order to detect known and unknown DDoS attacks with the most recent patterns in their work based on real-time detection[9].

Machine learning has been used to detect DDoS attacks in a number of different studies. For web-log analysis, Stevanovic suggests using two unsupervised neural networks: self-organising map (SOM) and modified adaptive resonance theory 2 (modified ART2). They applied this technology to assess the relative differences and similarities between malicious web crawlers and non-malicious visitor groups, as well as to analyze visitors' browsing behaviors [10]. Lee et al. suggested an improved DDoS attack detection approach that maximized detection capability by optimizing the parameters of the traffic matrix using a genetic algorithm (GA) [11].

The study, done by Zhang, DDoS Detection and Prevention Based on Artificial Intelligence Techniques, they evaluate the most recent advancements on detecting DDoS attacks using artificial intelligence techniques in their study. Number of packets, average packet size, time interval variance, packet size variance, number of bytes, packet rate, and bit rate are some of the parameters that can be utilized to detect DDoS attacks. For optimal performance, they recommend using the random forest tree and Naive Bayes artificial intelligence approaches to categorize harmful and non-malicious traffic. In addition, they suggested that to identify DDoS assaults, many machine algorithms can be coupled to improve accuracy and performance [12].

Deng, L., & Yu, D. (2014) [23],presented "A study of architectures, algorithms, and applications for deep learning, APSIPA," which describe a neural deep of networks. There are some methods accorded to deep learning that are divided into three groups: hybrid, generative and discriminative. The aim of this paper is to present the emerging area of deep learning. It is referring to a class of machine learning techniques, where many stages of non-linear information processing in hierarchical architectures are exploited for feature learning and pattern classification.

Bouyeddou, B., Kadri, B., Harrou, F., & Sun, Y. (2020) [3],presented the “DDOS-attacks detection using an

efficient measurement-based statistical mechanism". This study provided a credible mechanism of detection that depended on a statistical measure of likelihood scores.

Suresh & Anitha (2011) [14],presented the "Evaluating Machine Learning Algorithms for Detecting DDoS Attacks". Recently, it has been observed that the massive damage from DDoS attacks has increased and the rapid detection of these attacks and the proper response mechanisms are significant due to the signature-based detection systems of DDoS. This study presented several models of machine learning such as C4.5, Navies Bayes, K-Nearest-Neighbor (KNN), SVM, Fuzzy c-mean, and K-mean to effectively identify DDoS threats.

Li, Meng, Zhang, & Yan (2019)[25], proposed "DDoS Attacks Detection Using Machine Learning Algorithms". A DDoS service threat denial is a hostile effort to disrupt the targeted server, service, network, or normal traffic by flooding the infrastructure with a traffic flood, which harms the security of the network. This paper proposed a Principal Component Analysis - Repetitive Neural Network (PCA-RNN), which is a framework that identifies DDoS threats and understands the traffic comprehensively as most network characteristics describe traffic. The algorithm of PCA was implemented.

Our proposed method differs from other approaches by using deep learning. Pande et al. (2021) [26]published a paper titled "DDOS Detection Using Machine Learning Technique." They used a random forest algorithm to do the classification. They noted in the study that in the future, they will try to classify DDoS attacks using deep learning techniques, which we employed in our study. Furthermore, unlike other DDoS attack detection methods, our method classifies traffics by designating the attack type. The papers [21, 22] perform classification as normal and attack traffics.

3. PROBLEM STATEMENT

In the past decade, there has been a major development in computer networks. But with growth, so do the threats to computer networks. As discussed, a DDoS attack is a fundamental threat to computer network especially in application layer.

Many researchers have used intrusion detection techniques for machine learning, but some have shown poor detection, and some techniques take longer to train. Some surveys revealed that the algorithm of Naïve Bayes (NB) [12] offers to depend on the wrong presumption of equally essential and independent features compared to deep learning. Therefore, we suggest developing a Network Penetration Discovery System (IDS) with the deep learning algorithm.

HOW DOES DEEP LEARNING WORK?

The neural network's design is inspired by the structure of the human brain. Neural networks can be taught to perform the same tasks on data that our brains do when identifying patterns and classifying different sorts of information. Individual layers of neural networks can also be considered a kind of filter that works from the most obvious to the most subtle, improving the possibility of detecting and producing the correct result. The human brain operates similarly. When we get new knowledge, our brain attempts to compare it to previously encountered objects. Deep neural networks make use of the same notion. Neural networks are use to accomplish various tasks, such as grouping, classification, and regression. The layers of a neural network are divided into three categories: input layers, hidden layers, and output layers.

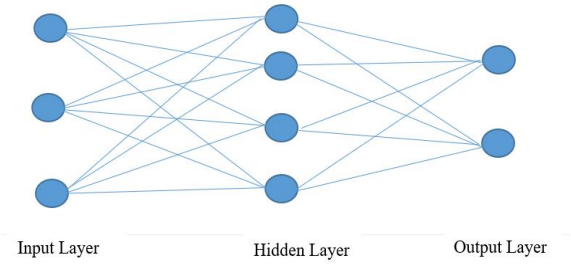

Figure 1: The Basic Architecture of the Neural Network

Figure1shows the most popular neural network architecture. The input is input nodes, while the active nodes are the rest of the nodes. Hidden layer nodes are linked to input layer nodes, and output units are connected to hidden layer nodes. The weights assigned to hidden layer nodes determine how this neural network behaves. The input nodes' primary function is to display the raw data that the network receives. The action of the hidden layer units is determined by this

input and the weight on the connections between hidden nodes and input nodes. The efficiency and behavior of the output layer nodes are determined by the hidden layer nodes' action or operation, as well as the weight between output layer nodes and hidden layer nodes.

4. PROPOSED METHODOLOGY

DDoS attacks represent a significant threat to network security. A variety of methodologies and tools have been developed to detect DDoS attacks and minimize the damage they cause. Despite this, the majority of approaches are unable to achieve efficient detection while minimizing false alarms. Deep learning techniques are the most suitable and efficient algorithm in this case for categorizing both normal and attack data. Hence, a deep neural network approach is proposed in this study to effectively mitigate DDoS attacks. The input dataset is first pre-processed, with the min-max normalization technique used to substitute all of the input in a given range. The normalized data is then fed into a deep neural network classifier, which divides the data into regular and attacked categories.

Our methodology consists of four key steps: data collection, data pre-processing, deep neural model training and testing, and finally, evaluation.

4.1 Data Collection

The data Collection is the first step of the proposed methodology to acquire both normal and attack traffic. We used two data sources to make our prediction: the CICIDS2017 dataset and data captured through running live mode by using CICFlowMeter.

CICIDS2017 Dataset

The Canadian Institute of Cybersecurity provide CICIDS2017 dataset which contains benign and up-to-date common attacks. The CICIDS2017 dataset was split into eight files,eight containing five days of usual and attack traffic data from the Canadian Institute of Cybersecurity .The dataset contains attack information in the form of five days of traffic data. Our workfocuses on normal traffic and four types of DDos attacks: DoS GoldenEye, DoS Hulk, DoS Slowhttptest, and DoS Slowloris attacks.

Since Slowhttptest and Slowloris act like normal traffic or users, only a small amount of traffic was captured. They generate low traffic, keep the connection open, and slowly make more and more connections, leading the server to be overloaded and crashed.

SimulatingDataset

In CICDDoS2017 dataset, there were 2359087 (90.3%) records classified as normal traffic and 252,660 (9.7%) classified as attack traffic. Therefore, we did not just lay on the data we got from CICIDS2017 since most of the data are benign. We collected new data by running in live mode. To collect new data, we used CICFlowMeter. CICFlowMeter is a tool that creates and analyses network traffic flows [14]. The simulating data are live traffic captured using CICFlowMeter. It represents a generator and analysis tool for detecting normal and attack traffic. It is executed to monitor network traffic running on the server X. In the CIC FlowMeter tool, the configuration is set to run network traffic capturing on the website. All network traffic data are recorded and captured by CICFlowMeter. Then, the captured data are saved as a PCAP format. However, the PCAP format is not following the proper format in the data training process, so it is necessary to convert the data into a CSV format according to the training process. CICFlowMeteralso was used to translate the data into CVS format, which is suitable for the neural model. When we run in live mode, we continuously capture live traffic, which keeps on updating, and this traffic continuously goes through CICFlowMeter. The raw traffic keeps on getting converted to a CSV file which is dropped into a folder. Our neural code keeps on picking up new CSV data from the folder and merging them with the data extracted from CICIDS2017 dataset.

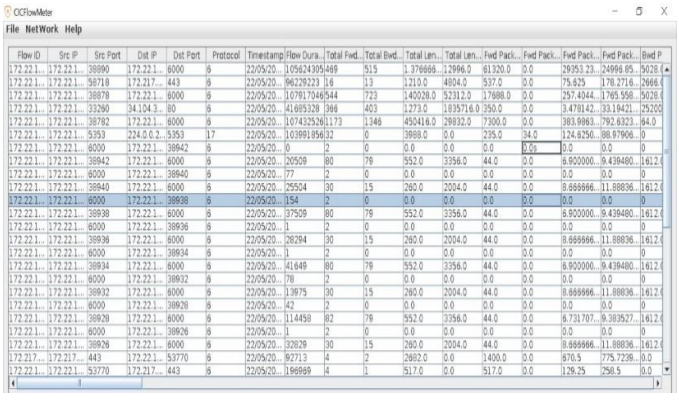

Figure 2:CICFlowMeterInterface after Capturing live traffics

Fig.2. CICFlowMeter converts raw traffic data and extracts machine learnable features from the network traffic flow. It generates analytical data that can be used to train our deep learning model. It can be used to generate bidirectional flows, where the first packet determines the forward (source

to destination) and backward (destination to source) directions. Hence 80 statistical network traffic features such as Duration, Number of packets, Number of bytes, Length of packets, etc., can be calculated separately in the forward and backward directions. Additional functionalities include selecting features from the list of existing features, adding new features, and controlling the duration of flow timeout. The application's output is the CSV format file with six columns labelled for each flow (FlowID, SourceIP, DestinationIP, SourcePort, DestinationPort, and Protocol) with more than 66 network traffic analysis features. After setting up our CICFlowMeter, it saves all the files in a given directory/folder from which our python script fetches all new data to analyse for detecting any DDOS attacks.

4.2 Data Pre-processing

Data pre-processing is a technique for transforming data into a format that is both useful and efficient. In this step, the dataset is prepared in a structured manner for modeling. This process entails two tasks: cleaning and transforming the data.

1) Data Cleaning

We observed that the dataset extracted from CICIDS2017 contains instances with missing class labels, missing information, NaNs, or infinity. These instances have been removed which are 1297 instances, to form a dataset containing unique instances of complete information.

The network traffic data was collected in CSV format and included over 66 flow features. In a neural network, it is essential to use the important features. Therefore, we started by removing categorical features that were not useful for classifying attacks and normal traffic. These features are Flow ID, Source IP, Source Port, Destination IP, Destination Port, Protocol, and Timestamp.

2) Dataset Transformation

This step is taken in order to transform the data in appropriate forms suitable for modelling process. In transformation, we used min-max normalization. The scaling is just applied on the numeric features since we already removed the categorical features. The purpose of scaling is to bring all the values in the same range. Before feeding that data to the neural network, the numerical data should be in the same range. There are multiple scalers and different algorithms for normalization that are beneficial. we decided to use min-max scaling.

One of the most popular methods of data normalization is min-max normalization. The minimum value of each function is converted to a 0, the maximum value is converted to a 1, and all other values are converted to a decimal between 0 and 1. For min-max scaling, the following equation is used.

$$x = \frac{x - XMin}{XMax - XMin} \qquad (1)$$

Where XMin and XMax are the minimum and the maximum value of the feature column X.

- When the value of x is the minimum value in the column, the numerator will be 0, and hence x is 0 [13].
- On the other hand, when the value of x is the maximum value in the column, the numerator is equal to the denominator and thus the value of x is 1[13].
- If the value of x is between the minimum and the maximum value, then the value of x is between 0 and 1[13].

4.3 Training and Testing the Deep Neural

The deep learning model is made up of three layers: input, hidden layers, and output. The network gets data from the input layer. The non-linearly separable connections are handled by the hidden layer, which also transfers data from the input layer to the output layer. The traffic is classified as benign or DoS attack in the output layer.

Our classification model uses a fully connected feed forward deep network. The input layer in this model takes flow level parameters as inputs. The number of neurons in the input layer is set at 66 as wechose 66 features from the flow data. The input layer's flow level features range from FP1 to FP66. The number of neurons in the output layer is equal to the number of classes in the dataset which are Benign, DoS

GoldenEye, DoS Hulk, DoS Slowhttptest, DoS slowloris attacks. As a result, the number of neurons in the output layer is set to five.

Training model

For defining our neural network, we used Karas library. Karas is a python-based deep learning framework which is the high-level API of Tensorflow. It is designed to easily create a neural network. For saving all the experiments,we did while running the neural model, we created experiments directory where all the experiments results will be saved. So, whenever we run a new experiment, the changes or the new model will be recorded into experiment directory. The dataset is divided into a training set 80% and a testing set 20 %. The 80% of the training data split furthermore into 80% for training and 20% for validation. The training data represent the collection of instances on which the model is trained, while the testing data is used to assess the model's generalisability, or performance.Table 1 shows the number of the data or traffics in each phase. The total number of data is 946,706 where 605,891 used for training the model, 151,473 traffics are used in validation process and 189,342 traffics for testing process.Table 2 gives more details on the traffics used in the training process. It lists the total number of traffics in each category, such as there are 281,397 benign traffics. The same manner for table 3. It displays the total number of traffics in each class. There are 70,349 benign traffics, 76,584 DoS Hulk traffics, 1,647 traffics represent DoS GoldenEye, DoS slowloris are shown in 2,013 traffics, and finally, there are 880 traffics that are DoS Slowhttptest.

Table 1: Total data in training, validationand testing sets

Total records	**Training**	**Validation**	**Testing**
946,706	605,891	151,473	189,342

Table 2: Summary of Training Dataset

Training Data	
Benign	281,397
DoS Hulk	306,334
DoS GoldenEye	6,588
DoS slowloris	8,053
DoS Slowhttptest	3,519

Table 3: Summary of Validation Dataset

Validation Data	
Benign	70,349
DoS Hulk	76,584
DoS GoldenEye	1,647
DoS slowloris	2,013
DoS Slowhttptest	880

The structure of the deep learning model is built using Keras TensorFlow.

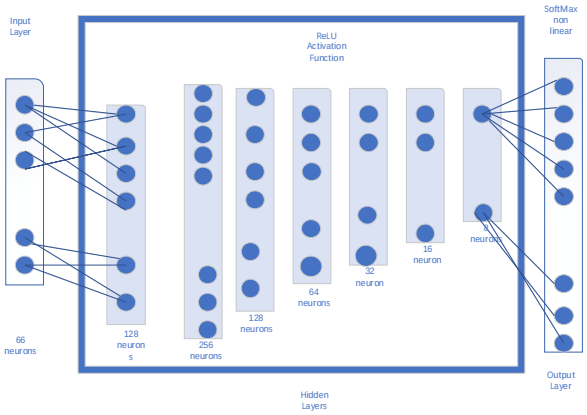

Figure 3: The Neural Network Model

We defined the initial input shape of the neural network as a 128 list of features that will be an input from our CICFlowMeter data as shown in Fig.3. After applying multiple layers of batch normalization, activation function, and dropouts the last layer of our neural network that consists of 5 outputs, there are four types of DDoS attacks our neural network will detect and one output being normal traffic. This deep learning network is a categorical model which takes 128 features as input and categories five outputs.

We also set some hyper parameters for the neural network where we used dropout as 0.2, batch_size as 1024, epochs as 300, classes as 5, alpha as 0.001. These hyper-parameters were giving the best results and output for the model after various trials and experiments.

Our model's keeps decreasing in every layer after adding activation and dropout until we make the last layer of (5) which are our categories that we are to predict.Our model is a categorical model so it's using categorical cross entropy as a loss function.We are using 66 features as input, so our input to the model is (n,6).It's starts to expand initially to 256 and then decreases with the 2's power to 128, 64, 32, 16, 8 eventually downsizing to 5.This allows the network to hold more dimensions and information and the concept of

dimensionality reduction is applied to reduce the results to 1D. After training the model, we calculated its Accuracy and Confusion matrix.

After training the neural network on collected data to distinguish benign traffic from DDoS traffic, we use this model to read newly fetched data from the same directory to predict and send this information to our react application.

Our react application and python scripts communicate using sockets, where every result the model makes is sending to the react application that displays a real-time graph as shown on Fig.6. We display lines that show traffic status. If the line is red, the model classifies incoming signals as malicious traffic, and if it is green, it is normal network traffic.

Testing model

In the last stage of the model-building phases, the models are tested data. At this stage, the data abused is the test set that results from the data break (20 percent).The data used in the test dataset, which totaled 189,342 traffics, is shown in Table 4. The number of benign traffics is 87,937, while the number of DDos attack traffics is 101,405. The majority was DoS Hulk with 95,730 traffics.

Table 4: Summary of Testing Dataset

Testing Data	
Benign	87,937
DoS Hulk	95,730
DoS GoldenEye	2,058
DoS slowloris	2,517
DoS Slowhttptest	1,100

5 RESULTS AND DISCUSSION

There are 189,342 instances in testing dataset. The truly predicted as benign traffics was 84167, while 7 were falsely predicted as Dos Golden Eye, 3723 were falsely predicted as Dos Hulk, 32 were falsely predicted as Dos SlowHttpTest and 8 was falsely predicted as Dos Slowloris. The trulypredicted as Dos Golden Eye was 1882, while 3 were falsely predicted as benign, 172 was falsely predicted as Dos Hulk, one traffic was falsely predicted as Dos SlowHttpTest. There are 95,537 traffics was correctly predicted as Dos Hulk. However,the falsely predicted traffics were 96predicted as benign, 90 predicted as Dos GoldenEye, 4 predicted as Dos Slowhttptest, 4 predicted as Dos SlowHttpTest and 3 was falsely predicted as Dos slowloris. In the Dos SlowHttpTest class, the truly predicted traffics were 1085, while 6 were falsely predicted as Benign, 1 was predicted as Dos Golden Eye and 4 were predicted as Dos Slowloris. There are 2,501 instances predicted correctly as Dos Slowloris. However, there were 4 traffics falsely predicted as benign, 1predicted as Dos Golden Eye, 3 were falsely predicted as Dos SlowHttpTest and 8 was falsely predicted as Dos Slowloris.

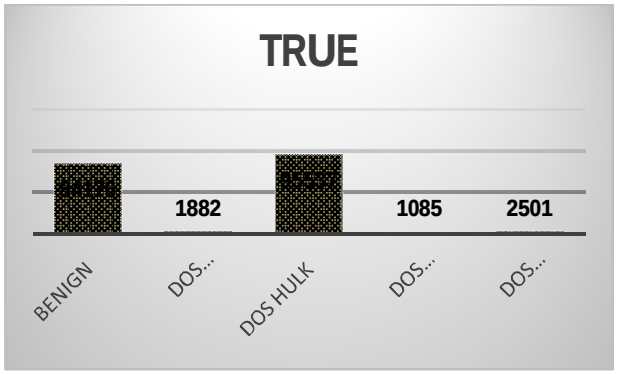

Figure 4: The Truly Predicted Traffics in Each Class

Fig. 4 presents a Confusion matrix for predictions using deep learning. The first column is for benign packets (84176) .There are 189,342 instances in testing dataset. The truly predicted as benign traffics was 84167, while 7 were falsely predicted as Dos Golden Eye, 3723 were falsely predicted as Dos Hulk, 32 were falsely predicted as Dos SlowHttpTest and 8 was falsely predicted as Dos Slowloris. As, we mentioned after making the prediction, our model send the result to react application. The react application show a graph of the prediction. The graph shows the type of data that are being captured and the time. Green line in the graph means the data are normal, where red line means the traffic is capturing in this particular time attack.

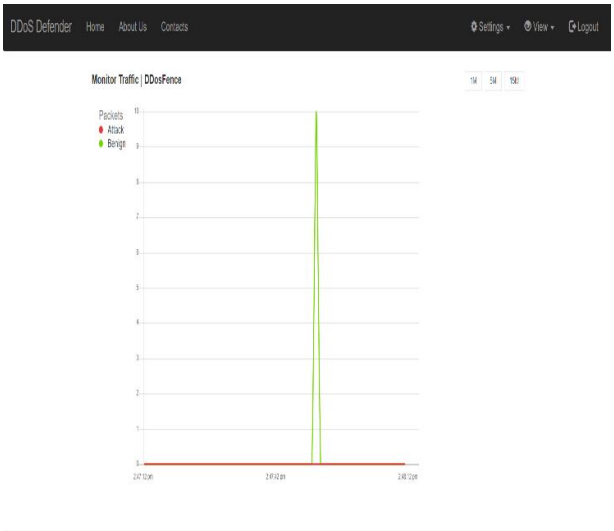

Figure 5: Green Graph Represents Benign Traffics

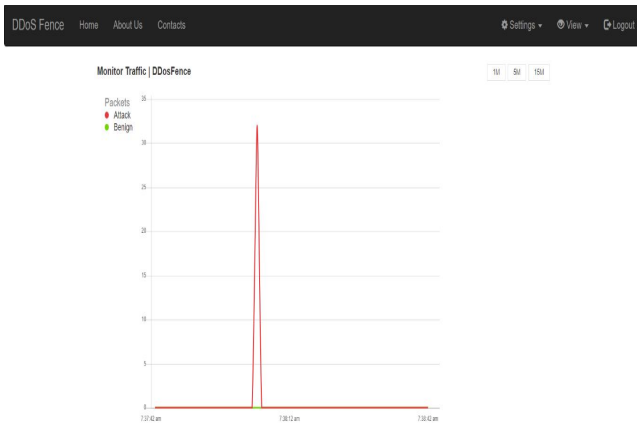

Figure 6: Red Graph Represents DDoS Traffics

Fig.6. shows our react application and python scripts that communicate using sockets, where every result the model makes is sending to the react application that displays a real-time graph. In this graph, we display lines that show traffic status. If the line is red, the model classifies incoming signals as malicious traffic, and if it is green, it is normal network traffic.

Overall, the experimental results on the CICIDS2017 combined with real-time dataset confirm that the proposed machine learning approach can effectively detect DDoS attacks with high detection rate.The proposed methodology using deep learning neural network showed 97%correctly predicting the four DDoS attacks.

6. CONCLUSION

DDoS attacks are a type of critical attack that endangers the availability of network resources and detecting them is challenging. The goal of this work is to use machine learning algorithms to detect DDoS attacks. We concentrated on four different types of DDoS attacks: Slowloris, Slowhttptest, Hulk, and GoldenEye. A deep learning neural network was used to recognize and classify traffic as benign or one of four types of DDoS attacks.The major contribution is to create real-time dataset and to use the deep learning neural network as a classifier for detecting DDoS attacks. Therefore, we used two data sources to create the prediction: the CICIDS2017 dataset and data acquired using CICFlowMeter through running in live mode which we called simulated data. We merged the CICID2017 and the simulated data. The total number of instances are 946,706 traffics.The proposed methodology achieved 97 % for identifying and distinguishing the four categories of DDoS attacks, which is an excellent result when compared to previous works like the study [24], which achieved 95.6 % for detecting DDoS attacks.

As future work, we plan to adapt our methodology to a broader range of DDoS attacks. We will also use other machine techniques like decision trees and Support Vector Machines to find out which model is the most accurate. Finally, rather than just predicting DDoS attacks, we will devise a strategy to avoid them.

REFERENCES

1. Apale, S., Kamble, R., Ghodekar, M., Nemade, H., & Waghmode, R. (2014). **Defense mechanism for DDoS attack through machine learning. International Journal of Research in Engineering and Technology**, 3(10), 291-294.
2. Bindra, N., & Sood, M. (2019). **Detecting DDoS attacks using machine learning techniques and contemporary intrusion detection dataset. AutomaticControl and Computer Sciences**, 53(5), 419-428.
3. Bouyeddou, B., Kadri, B., Harrou, F., & Sun, Y. (2020). **DDOS-attacks detection using an efficient measurement-based statistical mechanism. Engineering Science and Technology, an International Journal**, 23(4), 870-878.
4. Idhammad, M., Afdel, K., & Belouch, M. (2018). **Detection system of HTTP DDoS attacks in a cloud environment based on information theoretic entropy and random forest. Security and Communication Networks**, 2018.
5. Jaafar, G. A., Abdullah, S. M., & Ismail, S. (2019). **Review of recent detection methods for HTTP DDoS attack. Journal of Computer Networks and Communications**, 2019.
6. Kim, M. (2019). **Supervised learning□based DDoS attacks detection: Tuning hyperparameters**. ETRI Journal, 41(5), 560-573.
7. I. Sofi, A. Mahajan, and V. Mansotra**, "Machine Learning Techniques used for the Detection and Analysis of Modern Types of DDoS Attacks**," Int. Res. J. Eng. Technol., 2017
8. Chopra, A., Behal, S., & Sharma, V. (2021, March). **Evaluating Machine Learning Algorithms to Detect and Classify DDoS Attacks in IoT**. In 2021 **8th International Conference on Computing for Sustainable Global Development (INDIACom)** (pp. 517-521). IEEE.

9. Yusof, A. R. A., Udzir, N. I., & Selamat, A. (2019). **Systematic literature review and taxonomy for DDoS attack detection and prediction. International Journal of Digital Enterprise Technology**, 1(3), 292-315.
10. Stevanovic, D., Vlajic, N. and An, A. (2013) **'Detection of malicious and non-malicious website visitors using unsupervised neural network learning', Applied Soft Computing, January**, Vol. 13, No. 1, pp.698–708.
11. Lee, S.M., Kim, D.S., Lee, J.H. and Park, J.S. (2012) '**Detection of DDoS attacks using optimized trafficmatrix**', Computers & Mathematics with Applications, January, Vol. 63, No. 2, pp.501–510.
12. Zhang, B., Zhang, T., & Yu, Z. (2017, December). **DDoS detection and prevention based on artificial intelligence techniques**. In 2017 3rd IEEE International Conference on Computer and Communications (ICCC) (pp. 1276-1280). IEEE.
13. Wankhede, S., & Kshirsagar, D. (2018, August). **DoS attack detection using machine learning and neural network**. In 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA) (pp. 1-5). IEEE.
14. M. Suresh and R. Anitha ,**"Evaluating Machine Learning Algorithms for Detecting DDoS Attacks,"** CNSA 2011, CCIS , p. 441–452, 2011.
15. CISCO, **Defeating DDoS Attacks**, [Online]. Available: https://www.cisco.com/web/IT/events/pdf/iin2005/cisco_guard.pdf. [Accessed March 2021].
16. "**The New York Times**," [Online]. Available: https://www.nytimes.com/2016/10/22/business/internet-problems-attack.html. [Accessed March 2021].
17. Garcia, N. (2018). **The use of criminal profiling in cybercrime investigations (Doctoral dissertation, Master's Thesis)**. Available from ProQuest Dissertations & Theses Global database.(Accession Order No. AAT 10839020))
18. Kaviani, P., & Dhotre, S. (2017). **Short survey on naive bayes algorithm. International Journal of Advance Engineering and Research Development**, 4(11), 607-611.
19. Hussain, F., Abbas, S. G., Husnain, M., Fayyaz, U. U., Shahzad, F., & Shah, G. A. (2020). **IoT DoS and DDoS Attack Detection using ResNet**. arXiv preprint arXiv:2012.01971.
20. **Intrusion Detection Evaluation Dataset (CIC-IDS2017)**, [Online]. Available: https://www.unb.ca/cic/datasets/ids-2017.html
21. Pande, S., Khamparia, A., Gupta, D., & Thanh, D. N. (2021). **DDOS detection using machine learning technique. In Recent Studies on Computational Intelligence** (pp. 59-68). Springer, Singapore.
22. Lima Filho, F. S. D., Silveira, F. A., de Medeiros Brito Junior, A., Vargas-Solar, G., & Silveira, L. F. (2019). **Smart detection: an online approach for DoS/DDoS attack detection using machine learning. Security and Communication Networks**, 2019.
23. Deng, L., & Yu, D. (2014). **Deep learning: methods and applications. Foundations and trends in signal processing**, 7(3–4), 197-387.
24. Peraković, D., Periša, M., Cvitić, I., & Husnjak, S. (2016, November). **Artificial neuron network implementation in detection and classification of DDoS traffic**. In 2016 24th Telecommunications Forum (TELFOR) (pp. 1-4). IEEE.
25. Tian, G., Zhang, M., Zhao, Y., Li, J., Wang, H., Zhang, X., & Yan, H. (2019). **High Corrosion Protection Performance of a Novel Nonfluorinated Biomimetic Superhydrophobic Zn–Fe Coating with Echinopsis Multiplex-Like Structure. ACS applied materials & interfaces,** 11(41), 38205-38217.
26. Mochari-Greenberger, H., & Pande, R. L. (2021). **Behavioral health in America during the COVID-19 pandemic: meeting increased needs through access to high quality virtual care.**