Short Communication

iMedPub Journals
http://www.imedpub.com

American Journal of Computer Science and Information Technology

2020
Vol. 8 No. 4: 59

Detection of DDoS Attack using Traffic Analysis

Akarsh Joice*

Department of Computer Science and Engineering National Institute of Technology Calicut, Kerala, India

Corresponding author:
Joice A, Department of Computer Science and Engineering National Institute of Technology Calicut, Kerala, India

akashj@gmail.com

Abstract

DDoS attack is one of the biggest threads we are facing in today's networking world. DDoS attacks generating mass traffic deplete network bandwidth and/or system resources. Conventional solutions identify network traffic attack activities from legitimate network traffic based on statistical divergence. In this paper, we present deep learning based solution for predicting the DDoS attack. We design a RNN (Recurrent Neural Network) to learn network patterns from network traffic and use this learning to detect the attack.

Keywords: DDos; HTTP Headers; Algorithm; Deep Learning; Neural Network

Citation: Joice A (2020) Detection of DDoS Attack using Traffic Analysis. Vol.8 No.4: 59.

Received: September 07, 2020; **Accepted:** September 25, 2020; **Published:** October 2, 2020

Introduction

Problem statement

To design a Recurrent Neural Network for the detection of DDoS attack by traffic analysis.

Motivation

DDoS attack is one of the growing threads in the recent days. The objective of a DDoS attack is to prevent legitimate users from accessing your website. The DDoS attack will test the limits of a web server, network, and application resources by sending spikes of fake traffic. Some attacks are just short bursts of malicious requests on vulnerable endpoints such as search functions. DDoS attacks use an army of zombie devices called a botnet. These botnets generally consist of compromised IoT devices, websites, and computers. When a DDoS attack is launched, the botnet will attack the target and deplete the application resources. A successful DDoS attack can prevent users from accessing a website or slow it down enough to increase bounce rate, resulting in financial losses and performance issues. It causes website suffer performance issues or crash the server completely by overwhelming the server resources such as CPU, memory or even the entire network. This could be disastrous to a blogger whose livelihood depends on content distribution or ad revenue. Imagine what could happen to a business owner whose revenue depends on his e-commerce website [1]. So it is necessary to detect this attack at the early stage itself. Decision trees, random forests, distance based KNN (K Nearest Neighbors), discriminative based SVM (Support Vector Machines) were used to detect DDoS but where not successful as they cannot remember previous state of the network as it is very important to keep the state information so that change can be easily identified.

Objective

With the advent of Deep Learning, Recurrent Neural Network (RNN) was introduced, which has a memory unit that can remember the previous layer of the network. They are highly useful as they can keep track of the state changes at different times, which is necessary to detect attack. Our objective to build a Recurrent Neural Network based approach for DDoS attack Detection and calculates its accuracy.

Methodology

Data generation

The data was collected running Wire shark on Apache Server. Two sets of data was generated one for normal and the other for the attack. The normal data was collected by making a normal user to access the apache Server. The attack dataset was gen-rated by performing DDoS attack on the Apache Server by the Slow Loris attack tool [2].

Feature extraction

The following 29 fields of network traffic were taken for training the model. They are:

- Frame encoding

- Frame length
- Frame protocol
- IP header length
- IP total length
- IP flag reserved bit
- IP flag don’t fragement
- IP flag more fragments
- IP flag fragment offset
- Time to live
- IP protocol
- Source IP address
- Destination IP address
- Source port
- Destination port
- TCP length
- TCP ack
- TCP flag reserved
- TCP flag nonce
- TCP flag congestion window reduced
- TCP flag ecn-echo
- TCP flag urgent
- TCP flag acknowledgement
- TCP flag push
- TCP flag reset
- TCP flag syn
- TCP flag fin
- TCP window size
- TCP time delta

Transformation

After feature extraction, we get an mxn matrix, where m indicates the number of packets and n’ indicates the number of new features after transformation. In order to learn patterns in both long and short term, we use a sliding window to separate continuous packets and reshape the data into a series of time windows with window size T. The label y in each window illustrates the last packet [3]. After reshaping, we have a three-dimensional matrix with shape (m-T)xTxn. In this way, we change the features from conventional packet-based to window-based, by which we can learn network patterns from both previous (T-1) packets and current packet.

LSTM

LSTM is Recurrent Neural Network that can remember longer context information than RNN and also by having different cell states it can decide what information is important and not important. LSTM contains different gates and a cell state. The forget gate is used to decide whether to keep the information or not. It contains a sigmoid function which outputs values between 0 and 1. If the value of the activation functions in 0, the information is forgotten and if the value is 1, then the information is kept. The cell state is updated by the input gate. Input gate takes the previous hidden state and current input. It contains tan and sigmoid activation function and also has their multiplied values. Now the cell state is calculated by performing point wise addition with the output of the input gate. Finally, the output gates decide the next hidden state value.

GRU

Gated Recurrent Unit (GRU) is similar to the LSTM network but it contains only the update gate which decides whether to pass the previous output to the next cell or not. It also does not have a specific forget gate as it is replaced by an additional mathematical operation [4].

Proposed model

Deep learning model is built by stacking the Bidirectional LSTM followed by Bidirectional GRU. We design 64 neurons in each cell. This is fed to fully connect hidden layers. Tanh activation function is used for both the RNN layers and ReLU activation function is used for fully connected hidden layers. Adam is used as the optimizer and binary cross entropy used as the loss function (**Figure 1**).

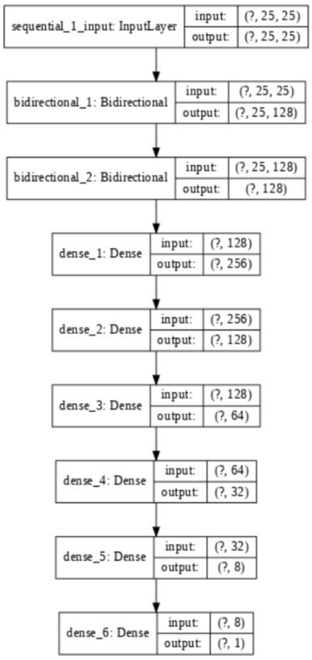

Figure 1: Favourable outcomes.

Favorable outcomes

Following are the favorable outcomes of the project:

- It provides security to the server from the attackers.
- It will be able to detect the DDoS attackers is the early stages of attack itself.

It helps the business and business owner whose revenue depends on his E-commerce website.

Equipment Required

Software specification

Tensor flow: It is developed by the Google for numerical computation, which is now widely used by many large companies. Tensor flow provides an interface for expressing machine learning algorithms and an application for executing these algorithms. It provides a framework which can be modified for working with machine learning algorithms with a set of reference models. It has a library of python in which the convolutional neural networks can be trained [5].

Slow Loris: Slow Loris is a type of denial of service attack tool which allows a single machine to take down another machine's web server with minimal bandwidth and side effects on unrelated services and ports. Slow Loris tries to keep many connections to the target web server open and hold them open as long as possible. It accomplishes this by opening connections to the target web server and sending a partial request. Periodically, it will send subsequent HTTP headers, adding to but never completing the request. Affected servers will keep these connections open, filling their maximum concurrent connection pool, eventually denying additional connection attempts from clients (**Figures 2-4**).

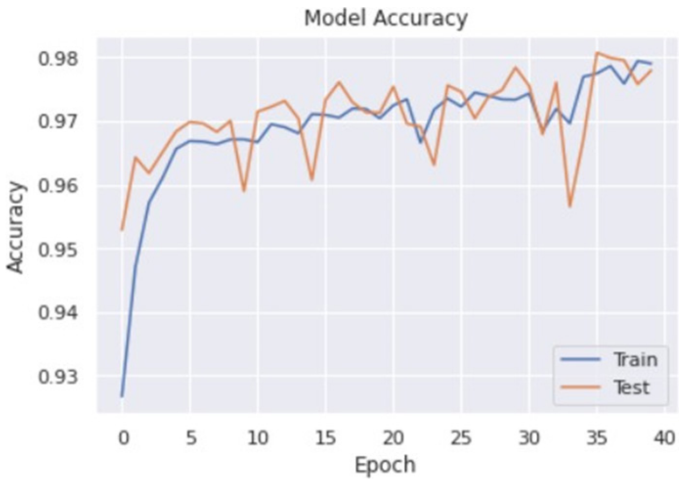

Figure 2: Plot on model accuracy.

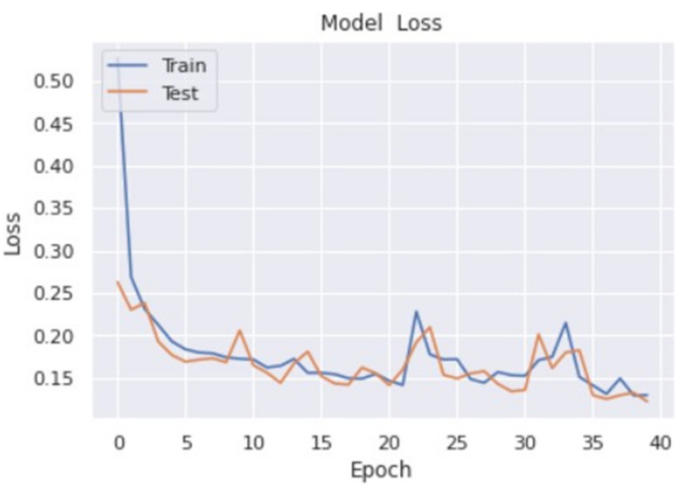

Figure 3: Plot on model loss.

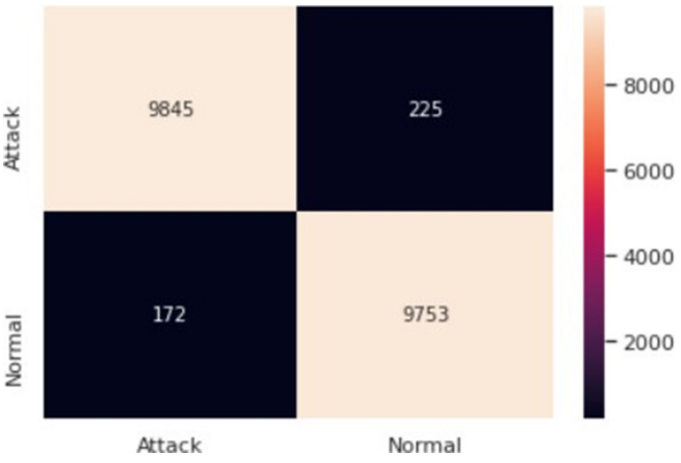

Figure 4: Confusion matrix.

Results and Discussion

Our proposed model has secured an accuracy of 98.01% and performs better than the decision tree model. We also observe that with the increase of window size, the performance of RNN model improves.

Conclusion

In this project, we have proposed a new model for DDoS detection system. The proposed model adds more contexts to the network which can result in faster learning and accurate results. It helps improve the performance of identifying DDoS attack traffic. We formulate the DDoS detection as a sequence classification problem and transform the packet-based DDoS detection to the window based detection. Our model can pave a path for more robust DDoS Detection Systems which can utilize the additional context information and time sensitivity.

References

1. Yuan X, Li C, Li X (2017) Deep defense: Identifying DDoS attack via deep learning. IEEE International conference on smart computing 1-8.
2. Kollu PK, Prasad RS (2019) Bidirectional LSTM based approach for network intrusion detection. Int J Recent Technol Eng 8: 1-6.
3. Mikolov T, Zweig G (2012) Context dependent recurrent neural network language model. IEEE Spoken language technology workshop 234-239.
4. Robinson T, Hochberg M, Renals S (1996) The use of recurrent neural networks in continuous speech recognition. Automatic speech and speaker recognition 233-258.
5. Amari SI (1998) Natural gradient works efficiently in learning. Neu Compu 2: 251-76.