TRIBHUVAN UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

CENTRAL DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

DETECTING DDOS ATTACKS USING LOGISTIC REGRESSION

By

Brihat Ratna Bajracharya

19/075

SUBMITTED TO THE CENTRAL DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE MASTER'S DEGREE IN COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

CENTRAL DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

KIRTIPUR, KATHMANDU

December, 2019

TRIBHUVAN UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

CENTRAL DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

SUPERVISOR'S RECOMMENDATION

I hereby recommend that this seminar report is prepared under my supervision by **Mr. Brihat Ratna Bajracharya** entitled "**Detecting DdoS Attacks using Logistic Regression**" be accepted as fulfillment in partial requirement for the degree of Masters of Science in Computer Science and Information Technology.

……

Mr. Jagdish Bhatta

Central Department of Computer Science

and Information Technology, TU

Date:

TRIBHUVAN UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

CENTRAL DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

CERTIFICATE OF APPROVAL

The undersigned certify that they have read, and recommended to the Institute of Science and Technology for acceptance, a seminar report entitled "**Detecting DdoS Attacks using Logistic Regression**" submitted by **Mr. Brihat Ratna Bajracharya** in partial fulfillment of the requirements for the Master's degree in Computer Science and Information Technology

Evaluation Committee

……..	……...
Mr. Nawaraj Paudel	Mr. Jagdish Bhatta
Head of Department	Supervisor
Central Department of Computer Science and Information Technology	Central Department of Computer Science and Information Technology

……..

Internal Examiner

DATE OF APPROVAL:

TABLE OF CONTENT

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
DDoS	Distributed Denial of Service
DNS	Domain Name Server
DoS	Denial of Service
FBI	Federal Bureau of Investigation
FSM	Finite State Machine
HTTP	Hyper Text Transfer Protocol
IAT	Inter Arrival Time
ICMP	Internet Control Message Protocol
IP	Internet Protocol
LDAP	Lightweight Directory Access Protocol
LS-SVM	Least Square Support Vector Machine
MSSQL	Microsoft Structured Query Language
NetBIOS	Network Basic Input/Output System
NTP	Network Time Protocol
PNN	Probabilistic Neural Network
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

LIST OF FIGURES

ACKNOWLEDGMENT

I would like to express my indebted gratitude and sincere thanks my supervisor Mr. Jagdish Bhatta for his continuous support, guidance and supervision by which this seminar has been possible. I would also like to thank Central Department of Computer Science and Information Technology for arranging such a schedule for the academic course.

Every attempt has been made to most of the details in each and every aspects of the seminar in this documentation so that the reader can clearly understand about the seminar. We would be pleased to get the feedback on this report. Finally I would like to express our gratitude to my family, friends and well wishers for encouraging and supporting me.

Brihat Ratna Bajracharya
19/075

ABSTRACT

In the present day digital world, most of our electronic devices have become smart and most of them have capability to connect to the internet. This capability has both pros and cons. The benefit is that we can control those devices from anywhere but possess a serious risk of being attacked and compromised. Those attack may have different intentions behind it like for fun, for ransom, theft of data, exploitation of privacy, and personal reasons. So, it has been crucial that we detect those attack before the attack could do any damage and also prevent similar attack in the future. In this report, we mainly focus in distributed denial of service (DDoS) because it is one of the most common type of attack. The victim's machine is attacked from multiple systems across the internet using infected botnets of networks. The attacker controls those infected botnet and is programmed to launch attack packet flood. In this report, we studied some of the researches relating to evidence gathering, attack detection using proposed system. We organized this report to briefly describe how logistic regression can be used to detect DdoS attack from incoming traffic.

Keywords: ***network security, DDoS attack, classification, logistic regression***

1. INTRODUCTION

1.1. Denial of Service (DoS) Attack

A DoS attack is a denial of service attack where a computer (or computers) is used to flood a server with TCP or UDP packets. In this attack, the service is put out of action as the packets sent overload the server's resources and make the server unavailable to other devices and users throughout the network.

1.2. Distributed Denial of Service (DDoS) Attack

A DDoS attack is the most common type of DoS attack in which multiple systems target a single system with a DoS attack. The targeted network is bombarded with packets from multiple locations. DDoS attack is more complicated to recover. DDoS attacks are not only a widespread attack but also the second most common cybercrime attack to cause financial losses [3] according to the United States Federal Bureau of Investigation (FBI) and since the attack is generated from multiple computers across distributed network, it becomes very hard to differentiate DDoS attack from the genuine traffic.

DDoS attacks can be done in two different ways, direct and indirect. Direct attack target the victim's machine in its weakness of the system while in indirect attack, attacks are performed on the elements associated with the victim's machine.

1.3. DDoS vs Flash Events

DDoS attacks corresponds to large number of compromised systems flood requests to one or more servers in distributed environment while Flash events corresponds to large number of genuine traffic that occurs due to certain flash events. Flash events how ever resembles to that of DDoS traffic. The main difference between two is that the access intent in case of flash event is genuine and shows natural flow pattern.

2. LITERATURE REVIEW

Many researchers have contributed their time in research of effectively detecting the DDoS traffic differentiating from genuine traffic. Researchers have dived into detection using various methodologies and techniques. This section briefly points out some of the detection techniques used for the detection of DDoS attacks and to correctly identify flash events

2.1. Classification Based Approach

Sahi et al. [1] has proposed a new classifier system for the detection as well as prevention of DDoS TCP flood attacks in their paper. Their proposed system can identify these attacks regardless of the form of in which these attacks comes to the system. In this classification based system, the detection sub-system collects the incoming packet within a time frame, and then passed through the blacklist checker where the source IP of the packet is checked with the database of blacklist. If the source IP is matched in blacklist, it is directly sent to prevention for safety, If the packet passed the blacklist check, it is then passed through the classifier which decides whether the packet is normal or abnormal (from attacker). The normal packets are forwarded to the destination IP while the abnormal packets' source IP is added to the blacklist thus simplifying future detection and the packet is sent to prevention sub-system

2.2. Entropy based system

Gera et al. [6] has proposed the system based on the anomaly of traffic patterns to differentiate between DDoS flood packets and genuine flash event traffic. This system is based on the fact that DDoS are artificial packets generated with the sole purpose of attacking the victim to flood its resources. So, this attack must have some kind of pattern. On the other hand, flash events pattern are more genuine and random and this randomness factor between two can be used for the detection of DDoS flood packets. Some of the parameters that can be used for this kind of detection are as follows:

a. Time Interval – before the DDoS attack, the network traffic is generally low. Suppose at time t1 the network shows less traffic and when the DDoS attack starts, the network traffic will spike up. Let this time be t2. In DDoS attack the network traffic abruptly increase while in case of flash event, the traffic will increase gradually (i.e. spread of a breaking news takes place in gradually increasing pace and end in same fashion)

b. Source entropy – It is the number of source IP addresses from which attack is launched. The traffic that comes from the same network is considered traffic cluster. A threshold entrophy is defined. Then while analysing the incoming traffic, if it is found that there are more source IPs but less traffic cluster, it is considered a flash event (i.e. more number of people pinging the news some few times). Similarly, if there are more source IPs and more traffic cluster, it is considered as a DDoS attack (spoofed) (i.e. attackers are using multiple IPs to attack the victim's machine and these multiple source IPs are used multiple times). Finally if the source IP is same and the traffic cluster is same, it is also considered DDoS attack (i.e. attackers attacking the victim's machine from the same IP; non-spoofed).

2.3. Regression Based Network Traffic Analysis

Divakaran et al. [5] proposed a framework of gathering evidences to detect traffic sessions related to attacks and malicious activities. They applied regression models to detect fundamental anomalous patterns. Their framework include three stage approach for evidence gathering.

a. Modeling and analyzing sessions to detect anomalous patterns – they defined a flow as a set of packets localized in time, and studied the inter-arrival time (IAT) of the flows in a session. Attack bots are likely to generate flows in fixed interval of time, i.e. the randomness in the arrival time is lesser than in normal scenario. The flow size is also another vital parameter to distinguish anomalous session with normal genuine session. Another parameter is the degree of the end host (number of distinct IP addresses that an end host communicates to). DoS attack can be detected using the modeling session using the degree of destination IP address.

b. Detecting scans and illegitimate TCP state sequences – The anomaly can also be detected using the TCP state sequence, the normal flow of a TCP packet can be modeled in a finite state machine with SYN, SYN+ACK, DATA and FIN packet. Any other flow that does not corresponds to this normal flow is considered illegitimate. From this FSM, we can say TCP flow is either legitimate or illegitimate is a binary decision depending on the FSM.

c. Evidence correlation and decision making – Spacial correlation is performed in this stage i.e. using IP addresses to correlate the anomalous pattern. One example is to look for the flows with the same IP address. Instead of spacial correlation, correlation in time can also be used but spacial correlation technique is more natural option as attacker may be passive for a while after successful exploit.

After the framework stages are completed, the analyzed data is processed to remove the outlier using a threshold for higher accuracy. The attack count if found nearly same will definitely raise suspicion. Also increasing count of attack packet with respect to time is also another type of anomaly detection technique. Regression models are then used to detect anomalous pattern. Authors have used linear regression to find the relationship between destination IP addresses with time. Quadratic regression is also used to detect anomalies using polynomial fit.

2.4. Using Artificial Neural Network

Saied et al. [2] used supervised ANN (feed forward, error back propagation) approach on three different type of packets: TCP, UDP and ICMP. The ANN ICMP uses three input nodes (source_ip, icmp_sequence, and icmp_ip) with four hidden layer nodes to get the result in one output node. Similarly, ANN TCP uses five input nodes (source_ip, tcp_sequence, source_port, destination_port, and tcp_flags) and four hidden layer nodes to get outpur output in one output node. And in ANN UDP, four input layer nodes (source_port, destination_port, source_ip, and packet_length) are taken with three hidden layer to get result in one output layer node. The ANN is trained to get output in either 0 or 1 indicating whether the packet is normal or abnormal respectively.

The basic flow for ANN based supervised learning is as follows:

1. DDoS detectors are installed in different network, each detector registering the IP of neighboring detectors and communicate via encrypted messages and continuously monitors the number of passing packets

2. If the number of packets are greater than certain threshold, then there is suspicion of the attack. The packets are sorted and IP of the victim is identified. The ANN retrieves the required patterns and prepares for the ANN engine. The trained ANN engine takes the input and produces the output.

3. The second step is repeated for three times and majority result is considered final out of the three outcomes.

This ANN engine needs to be trained regularly for detecting up-to-date DDoS attacks.

3. LOGISTIC REGRESSION CLASSIFIER

3.1. Theory

Logistic regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous (only two possible outcomes) variable. In other words, it is used to predict a binary outcome (1/0, Yes/No, True/False) for given a set of independent variables. Logistic regression can be considered as special case of linear regression when outcome variable Is categorical which depends on lots of dependent variables (called features). Logistic Regression can also be used to determine cases of getting more than two outcome for e.g. married/unmarried/divorced. Some familiar applications of logistic regression are email spam filter, fraud detection, cancer diagnosis, etc. [10]

The idea is to estimate the probability of an outcome being a 1 or a 0. Given that, the probability of the outcome being a 1 is given by p then the probability of it not occurring is given by 1-p. This can be seen as a special case of Binomial distribution called the Bernoulli distribution. Then the problem is converted in the form of generalized linear regression model given by $y=\beta_0+\beta_1 x_1+\beta_2 x_2+\ldots+\beta_n x_n$ where y is the predicted value, $x_1,x_2,\ldots,x_n$ are independent variables and $\beta_0, \beta_1,\ldots,\beta_n$ are coefficients to be determined. We can express this in vector form as $y=w^T X$ where $w=[\beta_0\beta_1\beta_2\ldots\beta_n]$ and $X=[1x_1x_2\ldots x_n]$.

Next we compute the odds as $odds=\frac{p}{1-p}$ and then take the natural log of the odds to make it continuously linear which is called logit given as $logit(p)=\log(\frac{p}{1-p})$. Now, we can write $logit(p)=y=w^T X$ This logit function acts as link between logistic and linear regression. We can now estimate the values for p by taking natural exponential on both sides. The final result will be $p(y=1)=\frac{1}{1+e^{-y}}$. This is known as the Sigmoid function

A sigmoid function is a bounded, differentiable, real function that is defined for all real input values and has a non-negative derivative at each point. A sigmoid function has a characteristic S- shaped curve or sigmoid curve. A standard choice for a sigmoid function is the logistic function shown in the figure 1 and defined by the formula $S(y)=\frac{1}{1+e^{-y}}$. This function is also known as logistic function.

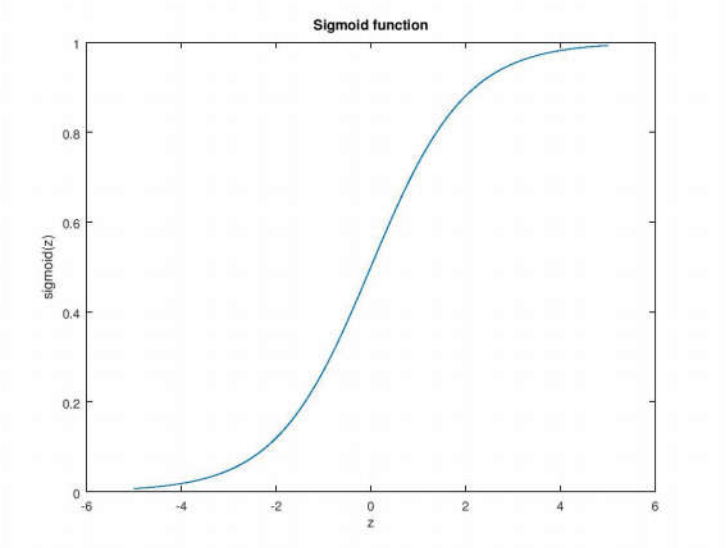

Figure 1: Sigmoid Function

From the above visual representation of sigmoid function, we can decipher that this curve describes many real-world situations, like population growth. In the initial stages it shows an exponential growth, but after some time, due to the competition for certain resources (bottle neck), the growth rate decreases until it gets to a stalemate and there is no growth.

3.1.1. Steps in Logistic Regression

Step 1: Classifying inputs to be in class zero or one.

First, we need to compute the probability that a training set belongs to class 1 (we can also call it to be a positive class) using the Logistic Function. In this case, our z parameter is, as seen in the below given function.

$$P(y=1)=\frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\ldots+\beta_n x_n)}}$$

The coefficient β_0 , β_1, …, β_n in formula are selected to maximize the likelihood of predicting a high probability for observations belonging to class 1 and predicting a low probability for observations actually belonging to class 0.

Step 2: Defining a boundary values for the classifier

We now define a threshold boundary in-order to clearly classify each given input values into one of the classes. We can choose a threshold value as per the business problem we are trying to solve, generally which is circled around 0.5. So if your probability values come out to be > 0.5 we can classify such observation into class 1 type, and the rest into class 0. The choice of threshold value is generally based on error types, which are of two types , false positives, and false negatives.

A false positive error is made when the model predicts class 1, but the observation actually belongs to class 0. A false negative error is made when the model predicts class 0, but the observation actually belongs to class 1. The perfect model would classify all classes correctly: all 1´s (or trues) as 1´s, and all 0´s (or false) as 0´s. So we would have FN = FP = 0.

3.1.2. Impact of threshold values:

1. *Higher threshold value* – Choosing higher threshold (e.g. P(y=1) > 0.7) will make model more restrictive when classifying as 1´s, and therefore we get more False Negative errors.
2. *Lower threshold value* – Choosing lower threshold (e.g. P(y=1) > 0.3) will make model less strict and will classify more examples as class 1, making more false positives errors.

3.2. Confusion Matrix

A confusion matrix (also known as error matrix) is a predictor of model performance on a classification problem. The number of correct and incorrect predictions is summarized with count values and broken down by each class. The confusion matrix shows the ways in which the classification model is confused when it makes predictions on observations, it helps us to measure the type of error our model is making while classifying the observation into different classes.

		Actual	
		Positive	Negative
Predicted	Positive	**True Positive**	**False Positive**
	Negative	**False Negative**	**True Negative**

Figure 2: Confusion Matrix

3.2.1. Elements of Confusion Matrix:

1. *True Positive (TP):* This refers to the cases in which we predicted "YES" and our prediction was actually TRUE
2. *True Negative (TN):* This refers to the cases in which we predicted "NO" and our prediction was actually TRUE
3. *False Positive (FP):* This refers to the cases in which we predicted "YES", but our prediction turned out FALSE
4. *False Negative (FN):* This refers to the cases in which we predicted "NO" but our prediction turned out FALSE

3.2.2. Learning Metrics From Confusion Matrix:

1. *Accuracy* – It answers how much the classifier is correct and is defined by following formula.

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

2. *Precision* – It answers how often the model correctly predicts positive values when it predicts positive and is defined by the relation

$$Precision = \frac{TP}{TP + FP}$$

3. *Recall* – It answers how often the model correctly predicts actually positive result and is defined by the relation. It is also known as sensitivity

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

4. *F1-score* – It is the harmonic mean of precision and recall. So, it is defined as:

$$F1\,score = \frac{2 * Precision * Recall}{(Precision + Recall)}$$

3.3. Implementing Logistic Regression Classifier:

Following steps are covered when we implement logistic regression

1. Reading Data
2. Analyzing Data (Basic EDA/Descriptive analysis)
3. Train and Test (Breaking the sample data into two sets)
4. Accuracy Report (Measuring the model performance using confusion matrix discussed in section 3.2.)

Details about implementing logistic regression is presented in section 4.

4. IMPLEMENTATION

4.1 Dataset Description

The dataset we used for the implementation is the CICDDoS2019 [9] obtained from University of New Brunswick (Canadian Institute of Cybersecurity, website: https://unb.ca). This dataset includes the result of abstract behaviour of 25 users based on the HTTP, HTTPS, FTP, SSH, and email protocols. This dataset consist of modern reflective DDoS attacks such as PortMap, NetBIOS, LDAP, MSSQL, UDP, UDP-Lag, SYN, NTP, DNS, and SNMP. We studied and implemented logistic regression on PortMap and LDAP variant of DDoS attack.

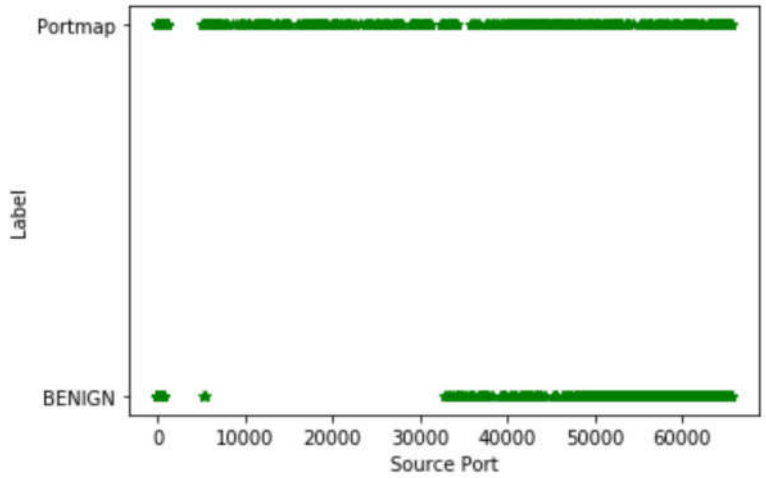

Figure 3: Source Port vs Label Plot for Portmap dataset

The dataset of Portmap attack consist of 88 columns and 191694 records. Out of 88, 87 columns are various dependent variables (features) and last one is the deciding class for the attack. The 'Label' column consist of two values: Benign and Portmap. Figure 3 shows the data plot of Source Port vs Label of Portmap DDoS attack

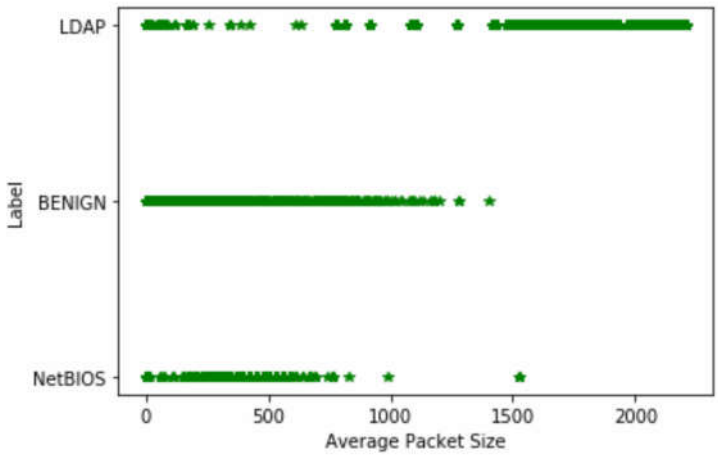

Figure 4: Plot of Average Packet Size vs Label for LDAP dataset

Similarly, another dataset of LDAP attack consist of 88 columns and 2113234 records. Like Portmap dataset, this dataset also has 87 features and one deciding class at the end called 'Label'. Unlike Portmap dataset, the deciding class has three values: Benign, NetBIOS, and LDAP. So, for this dataset, we need to use multinomial logistic regression. Figure 4 shows the plot between Average Packet Size vs Label for LDAP dataset

4.2. Feature Selection

Since, there are a lot of features in the dataset and it is not possible to include all the features for predicting the kind of attack, we employed correlation technique to find out the relationship between these 87 features with "label" class. And, we choose some of the top features which are highly correlated with 'Label'.

The features taken for Portmap dataset with its correlation coefficient are given below:

```
{'Protocol': 0.705635574606102,
 'Fwd_Packet_Length_Min': 0.7291026803636192,
 'Min_Packet_Length': 0.7291679201346289,
 'Destination_IP': 0.7884252762633482,
 'Source_Port': 0.8189050406122815,
 'Inbound': 0.8600933612454168,
 'Label': 1.0}
```

And the features taken for LDAP dataset with its correlation coefficient are given below:

```
{' Subflow Fwd Bytes': 0.9074102023178582,
 'Total Length of Fwd Packets': 0.9074102023178582,
 ' Max Packet Length': 0.9834324294254687,
 ' Min Packet Length': 0.9848409090943522,
 ' Fwd Packet Length Min': 0.9848622293276422,
 ' Fwd Packet Length Max': 0.9850204261207723,
 ' Average Packet Size': 0.9850804491267217,
 ' Fwd Packet Length Mean': 0.9852247505387776,
 ' Avg Fwd Segment Size': 0.9852247505387776,
 ' Packet Length Mean': 0.9854861398026699,
 ' Label': 1.0}
```

4.3. Regression Analysis

After selecting the features, the dataset was divided into train and test set in the ratio 4:1 and the regression model was trained using train set and tested on test set. This was done for both

datasets (Portmap and LDAP). Using the predicted test result from trained model and actual test set, confusion matrix was calculated for the performance evaluation. The confusion matrix for Portmap dataset is shown in figure 5 and confusion matrix for LDAP dataset is shown in figure 6.

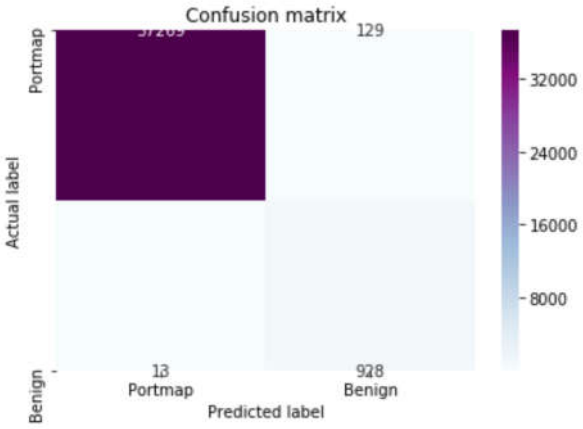

Figure 5: Confusion Matrix for Portmap Analysis

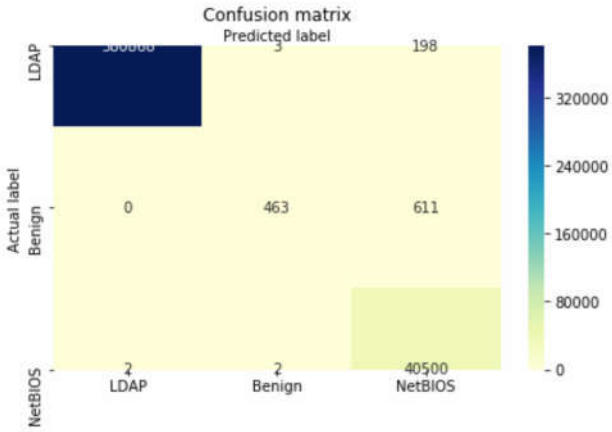

Figure 6: Confusion Matrix for LDAP and NetBIOS Analysis

Finally from these confusion matrix, accuracy, precision, recall and f1 score was calculated for both datasets whose result is given here. Following result were obtained on Portmap dataset

Accuracy: 0.9962961996922194
Precision: 0.9962961996922194
Recall: 0.9962961996922194
F1 Score: 0.9962961996922194

And following results were obtained on LDAP dataset

Accuracy: 0.998069310796007
Precision: 0.998069310796007
Recall: 0.998069310796007
F1 Score: 0.998069310796007

5. DISCUSSION AND CONCLUSION

In this study, a logistic regression classifier was used to predict the detection of three variants of DDoS attacks using the dataset obtained from University of New Brunswick (CICDDoS2019). Among different features in the dataset, a correlation test was done for each of the feature with the detection label to check the relevancy and few top features that are highly correlated with the label were used in the logistic regression. The Portmap attack detection used seven features in the regression classifier and has an accuracy of 99.6 % detection with f1 score of 0.996 while the LDAP and NetBIOS attack detection used ten features in the regression classifier and has an accuracy of 99.8 % detection with f1 score of 0.998. Hence, we can conclude that the logistic regression classifier is suitable for detecting DDoS attacks and its variants.

REFERENCES

[1] A. Sahi, D. Lal, Y. Li, and M. Diykh, "An Efficient DDoS TCP Flood Attack Detection and Prevention System in a Cloud Environment", in *IEEE Access*, vol. 5, pp. 6036-6048, 2017.

[2] A. Saied, R. E. Overhill, T. Radzik, "Detection of known and unknown DDoS attacks using Artificial Neural Networks", in *Neurocomputing 172*, pp. 385-393, 2016.

[3] B. Cashell, W. D. Jackson, M. Jickling, and B. Webel, The Economic Impact of Cyber-Attacks, document CRS RL32331, *Congressional Research Service Documents*, Washington, DC, USA, 2004.

[4] C. Bedon, A. Saied, Snort-AI (Version 2.4.3), "Open Source Project", 2009. Available from: http://snort-ai.sourceforge.net/index.php/

[5] D. M. Divakaran, K. W. Fok, I. Nevat, and V. L. L. Thing, "Evidence gathering for network security and forensics", in *Digital Investigation 20*, pp. S56-S65, 2017.

[6] J. Gera, and B. P. Battula, "Detection of spoofed and non-spoofed DDoS attacks and discriminating them from flash crowds", in *EURASIP Journal on Information Security,* doi:10.1186/s13635-018-0079-6, 2018.

[7] M. Roesch, Snort (Version 2.9), "Open Source Project", 1998. Available from: http://www.snort.org/.

[8] R. Russell, Iptables (Version 1.4.21), "Open Source Project", 1998. Available from: http://ipset.netfilter,org/iptables.man.html/.

[9] *DDoS Evaluation Dataset "CICDDoS2019" Dataset*, [online] Available: http://205.174.165.80/CICDataset/CICDDoS2019/Dataset/CSVs/CSV-03-11.zip

[10] *"Logistic Regression For Dummies: A Detailed Explanation".* Accessed on: Dec. 9, 2019. [Online]. Available: https://towardsdatascience.com/logistic-regression-for-dummies-a-detailed-explanation-9597f76edf46