International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 06 | June -2017 www.irjet.net p-ISSN: 2395-0072

Machine Learning Techniques used for the Detection and Analysis of Modern Types of DDoS Attacks

Irfan Sofi[1], Amit Mahajan[2], Vibhakar Mansotra[3]

[1]Student, Department Of Computer Science & IT, University Of Jammu, J&K, India.
[2]System Analyst, Department Of Computer Science & IT, University Of Jammu, J&K, India.
[3] Professor, Department Of Computer Science & IT, University Of Jammu, J&K, India.

Abstract - *Distributed Denial of service (DDoS) attacks is the most devastating attack which halts the normal functionality of critical services provided by the various organizations in the internet community. These attacks have become more sophisticated and continue to increase in number day by day, thus making it difficult to detect and counter such attacks. Therefore there is a need of intelligent intrusion detection system (IDS) to detect and classify any anomalous behavior of the network traffic. In this paper, the work is carried out on the new dataset which contains the modern type of DDoS attacks such as (HTTP flood, SIDDoS). This work incorporates various machine learning techniques for classification: Naïve Bayes, MLP, SVM, Decision trees*

Key Words: **DDoS Attacks, IDS, Naïve Bayes, Decision trees, MLP, SVM, ARFF File, and WEKA**

1. INTRODUCTION

With the proliferation of computer networks, especially the internet, come many kinds of network attacks. Recently global ransom ware virus named as Wannacry have halted network services in about 156 countries. According to reports of Kaspersky Lab in the fourth quarter of 2015, resources in almost 69 countries were targeted by Botnet assisted attacks. Also fourth quarter witnessed the longest Botnet based DDoS attack which lasted for 371 hours i.e. 15.5 days approximately. Crackers or black hackers are continually generating new types of DDoS attacks which is multilayered but mostly occur at the network and the application layer of OSI model. These attacks make use of the spoofed IP addresses to elude the source identification and to carry out the attack at the large scale. Such attacks are very immense that the available bandwidth at the bottleneck is completely utilized by the attack traffic thereby dropping the legitimate packets. The victims are surprisingly government agencies, financial corporations, defense agencies and military departments. Popular websites like Facebook, twitter, wikileaks, paypal, ebay became victims of DDoS which experienced interruption in normal operations leading to financial losses, service degradation and lack of availability [2].

The detection is quite difficult as the illegitimate packets are indistinguishable from the legitimate packets. Moreover, the cracker or black hacker quickly leaves the zombies after it executes the command; therefore detection of the cracker is extremely difficult. Thus there is a need of the intelligent intrusion detection system (IDS) to defend the network services. To develop the system we utilized the various machine learning techniques for detection and analysis of the behavior of DDoS packets using anomaly-based approach.

This paper outlines the various machine learning classification techniques like Naïve Bayes, MLP, SVM and decision trees for the detection and analysis of various types of DDoS attacks such as SIDDoS, HTTP flood, Smurf, UDP flood. In this paper the work is carried out on the novel dataset which contains the modern types of DDoS attacks because there were no common data sets that contains the modern DDoS attacks in different layers, such as (SIDDoS, HTTP flood)[1].The comparative analysis of different classification techniques is done and from the experimental results it is clear that MPL achieved the highest accuracy rate.

2. RELATED WORK

In recent literature, many methods have been introduced to detect and analyze DDoS attacks. The majority of current detection projects depend upon feature selection from the ip packets captured. Mouhammd Alkasassbeh et al. has taken all the 27 features into consideration in a novel dataset that contains the modern DDoS attacks in the different network layers, such as (SIDDoS, HTTP Flood). This paper mainly focused on the comparative analysis of various classifiers used in classification and determine the confusion matrix of each technique used. The method incorporates the well-known machine learning techniques like Naïve Bayes, Multilayer Perceptron (MLP), and Random Forest. Among these techniques it is shown that MLP achieved the highest accuracy rate (98.63) [1].

Sanguk Noh et al. works on all the flags within the TCP header and they analyze the relationship between the flags and the TCP packets. To analyze the features of the DDoS attacks, therefore, this paper presents the network traffic analysis mechanism which computes the ratio of the number of TCP flags to the total number of TCP packets. Based upon the calculation of TCP flag rates, they compile a pair of the TCP flag rates and the presence (or absence) of the DDoS attack into state-action rules using machine

learning algorithms. The alarming agent then detects the network flooding against a Web server [2].

Sherif Saad et al. Proposed a new approach for characterizing and detecting botnets using network traffic behaviors. This approach focuses on detecting newest and most challenging types of botnets before they launch their attack. Different machine learning techniques have been used to meet the online botnet detection requirements, namely adaptability, novelty detection, and early detection. The results of the experimental evaluation based on dataset taken showed that it is possible to detect effectively botnets during the botnet Command-and Control (C&C) phase and before they launch their attacks using traffic behaviors only [3].

Niharika Sharma, Amit Mahajan, Vibhakar Mansotra in their paper studied captured PCAP file and analyse the DoS attack using the decision tree data mining tool.They have used classifier model in the WEKA tool for intrusion detection method. For Decision tree algorithms it shows a set of rules that determine whether or not SYN flooding exists. The decision tree in the output states that if no. of SYN packets from same source to same destination is greater than one than it is considered as threat otherwise it is normal i.e. Tcp.flags.syn <=0: Normal and Tcp.flags.syn >0: Threat.[4]

Rough Set Theory (RST) and Support VectorMachine (SVM) to detect network intrusions. First, packets are captured from the network, RST is used to pre-process the data and reduce the dimensions. The features selected by RST is sent to SVM model to learn and test respectively. The method is effective to decrease the space density of data. The experiments compare the results with Principal Component Analysis (PCA) and show RST and SVM schema could reduce the false positive rate and increase the accuracy. The three main approaches we are considering is Paxson's Bro, Leckie et al's probabilistic approach and Jung et al's sequential hypothesis testing for scan detection.[5]

Carl Livadas et al author use machine learning techniques to identify the command and control traffic of IRC-based botnets (compromised hosts that are collectively commanded using Internet Relay Chat (IRC)). The author split this task into two stages: (I) distinguish between IRC and non-IRC traffic, and (II) distinguishing between botnet IRC traffic and real IRC traffic. For Stage I, He compare the performance of J48, naive Bayes, and Bayesian network classifiers, identify the features that achieve good overall classification accuracy, and determine the classification sensitivity to the training set size. A naive Bayes classifier performs best, achieving both low false negative (2.49%) and false positive (15.04%) rates for real-life IRC/non-IRC flows and low false negative (7.89%) rates for our botnet testbed IRC flows. While some J48 and Bayesian network classifiers perform better for real-life IRC/non-IRC flows, they classified botnet testbed IRC flows poorly. For the feature sets and the traces considered, it was observed that training sets of 10K flows are sufficient and that the benefit of using larger sets is minimal.[6]

Research paper by Martin J Reed et al. presents an introduction to intrusion detection systems (IDS) and survey of different DoS/DDoS detection techniques. An overview and broad classification IDS are presented. The difficulties and characteristics of DoS/DDoS attacks are discussed in the DoS detection section. Furthermore, a classification of DoS attacks is explained. Three different classifications have been chosen and divided in two groups: general DoS classification and network flooding DoS-based. In each classification, many different proposed techniques are introduced and reviewed to point out the limitations. The key observation of this survey paper is that a CUSUM-based detection technique has many advantages over other statistical instruments in that it is nonparametric; consequently, it does not require training and is more robust to variations in the attack profile. [7]

Research paper by Prajakta Solankar et al shows various techniques for classification of attack. K-Nearest Neighbor (KNN), support vector machine (SVM), decision tree and naïve Bayes are described and experimental results by using weka tools are determined. In this paper various denial of service attack types and review of various classification techniques like support vector machine, k-NN, Naïve Bayes and decision tree are given. From weka tool, the author analyzed that support vector machine and k-NN having more accuracy than all other however k-NN requires more time.[8]

In this paper, the authors Bayu Adhi Tama et al attempts to classify papers concerning DoS/DDoS attack detection using data mining techniques. 35 papers were selected and carefully reviewed by authors from two online journal databases. Each of selected paper was classified based on the function of data mining such as association, classification, clustering, and hybrid methods. The findings of this work indicate that classification and hybrid techniques received a great deal of attention from researchers. Our literature review provides a state of the art analysis concerning DoS/DDoS attack detection using data mining techniques. [9]

Research paper by V. Hema et al presents a traffic classification scheme to improve classification performance when few training data are available is used. The traffic flows are described using the statistical features and traffic flow information is extracted. A traffic classification method is proposed to aggregate the Naïve Bayes predictions of the traffic flows. Since classification scheme is based on the posterior conditional probabilities, it can identify attacks occurring in an uncertain situation. The experimental results show that the proposed scheme can efficiently classify packets than existing traffic classification methods and achieved 92.34% accuracy. [10]

3. UNDERSTANDING DDoS ATTACK

In "distributed denial-of-service" attack an attacker attempt to prevent legitimate users of a service from using that service. DDOS is a distributed denial of service attack carried out from many sources simultaneously, so there's

not just one or two IP addresses to block. The attack may exploit a vulnerability in a third party's service, e.g. NTP or DNS, so you are actually seeing packets from legitimate sites like businesses or universities which cannot be closed down, though there are ongoing projects to locate and advise these sites of the problem and get them to patch their service. We outline the details of such type of attacks for clarity. If 'A' an attacker has IP address 1.2.3.4 and 'B' victim has IP address 5.6.7.8, 'A' can send a packet with 'B' IP address 5.6.7.8 as the source to xyz.com and say "tell me all about X". So xyz.com sends a bunch of data to attacker 'A' that he didn't ask for. If 'A' do that to abc.com, def.com etc. all asking them to send data to 5.6.7.8, that's a DDOS attack. As a result connection buffer of the victim will be filled up with pending connections which will never be completed, and thus prevent it from answering new requests that may be valid.

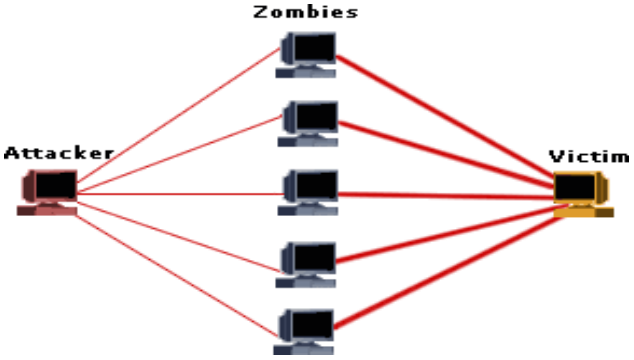


Fig: DDoS attack

DDoS attacks can be implemented on different layers of the OSI model. In this paper, we describe the testing of the most recent attacks on the network and the application layer.

3.1. Network Layer Attacks

A Smurf attack and User Datagram Protocol (UDP) flood attack are part of the network layer attacks. In a Smurf attack, the victim is flooded with Internet Control Message Protocol (ICMP) echo-reply packets. This attack uses IP broadcasting in which when a packet is sent to an IP broadcast address from a machine on the local network, that packet is delivered to all machines on that network. Under these circumstances, the attacker broadcasts packets with the spoofed source IP address targeted to the victim. Since the packets are sent at broadcast address, it is received by all the nodes within the network. Each node responds back to the victim machine since the source IP address is spoofed as that of the victim's address

In UDP Flood attack attacker sends large number of UDP packets to random ports of their target server, which results in saturation of the network and the depletion of available bandwidth for legitimate service requests to the victim system. On receiving a UDP packet, a victims system will try to determine the waiting application on the destination port. When there is no application waiting on the port, it will generate an ICMP packet of —destination unreachable to the forged source address.

3.2. Application Layer Attacks

HTTP flood and UDP flood are the modern types of application layer attacks. . In HTTP flood the attacker exploits the HTTP GET or POST requests to attack a web server or application. These attacks are also significantly harder to detect and block. An HTTP client is like a web browser —talks‖ to an application or server by sending an HTTP request either of GET or POST type. .In incomplete HTTP Flood attack using the GET method, the client sends HTTP requests to the web server but in a different way. The Client never sends the complete HTTP header but sends just a part of it. Client continues to send subsequent headers at regular intervals to keep socket alive. By sending multiple incomplete requests the server's resources get exhausted. These request consume all the available resources on the server, thereby denying the legitimate users' requests.

Moreover, most modern DDoS application layer attacks are SQL Injection Distributed Denial of Service (SIDDOS), where Attackers start from the client side, for example the browsers, by inserting a malicious code, and forwarding it to the server side[1].

4. DATASET COLLECTION

In this research a new dataset is collected because there is no existing data sets that contain a modern DDoS attack such as (SIDDOS, HTTP Flood) and furthermore, other available data sets may include a great deal of duplicate and redundant records, and that may result in an ultimate unrealistic outcome. Our collected dataset contains four types of DDoS attack as follows: (HTTP Flood, SIDDOS, UDP Flood, and Smurf) without redundant and duplicate records. Table 1 lists number of records of these types of attack. Table 2 shows the dataset features we dealt with.

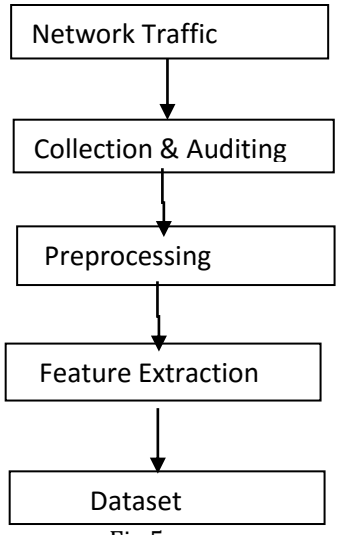


Fig.5

The proposed system that is used to collect data is shown in the figure 5.The various steps in the process are briefly described as:

- Collection and auditing: in this step, all network traffic is collected and audited from NIDS.
- Preprocessing file format: redundant and duplicate records are removed.
- Feature extraction: Extracts features parameters from the collected network traffic and assigns each feature to the first column; these will be used as a vector in the new dataset.

Table 1

Attack Name	*No. of Records*
NormaL	67752
UDP Flood	7020
Smurf	421
SIDDOS	221
HTTP Flood	138

5. CLASSIFICATION

In this work we investigated and tested various classifiers such as Naïve Bayes, MLP, Decision trees and SVM. By using these classifiers we perform the comparison and analysis of accuracy, precision, recall rates etc. of the DDoS packets and normal packets.

5.1. Decision Tree

Decision Trees are one of the most widely use data mining tool for classification purposes. A decision tree is used as a classifier for determining an appropriate action (among a predetermined set of actions) for a given case [4]. They are a non-parametric supervised learning method used for classification and regression purposes. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Structure of Decision Tree
A decision tree has three types of nodes:
Root node: Root node is the top most node. It has no incoming edge but zero or more outgoing edge.
Internal node: internal node has exactly one incoming edge and two or more outgoing edges.
Leaf node: Leaf node has exactly one incoming node and no outgoing edge.
Decision trees play an important role in the process of intrusion detection. From an intrusion detection perspective, decision trees can classify incoming packet as malicious, normal or any other category using information like source port, destination port, no. of SYN flags from a particular source to destination port (in case of SYN Flood) etc.

Table –2

Variable No	*Description*
1	SRC ADD
2	DES ADD
3	PKT ID
4	FROM NODE
5	TO NODE
6	PKT TYPE
7	PKT SIZE
8	FLAGS
9	FID
10	SEQ NUMBER
11	NUMBER OF PKT
12	NUMBER OF BYTE
13	NODE NAME FROM
14	NODE NAME TO
15	PKT IN
16	PKTOUT
17	PKTR
18	PKT DELAY NODE
19	PKTRATE
20	BYTE RATE
21	PKT AVG SIZE
22	UTILIZATION
23	PKT DELAY
24	PKT SEND TIME
25	PKT RESEVED TIME
26	FIRST PKT SENT
27	LAST PKT RESEVED

5.2. Naïve Bayes

Naïve Bayes is a simple probabilistic classifier that returns p (y|x), Naïve Bayes calculates probabilities for each class in a dataset and determines discriminative learning to predict values of the new class. Given a set of variables, X = {x1, x2, x3..., xd}, we want to construct the posterior probability for the event Cj among a set of possible outcomes C = {c1, c2, c3..., cd}. In a more familiar language, X is the predictors and C is the set of categorical levels present in the dependent variable.

Using Bayes' rule:

$$p(Cj|x1, x2, x3, , xd) \propto p(x1, x2, x3 \dots xd|p(Cj)).P(Cj)) \quad \dots\dots (1)$$

Where p(Cj | x1, x2, x3..., xd) is the posterior probability of class membership, i.e., the probability that X belongs to Cj.

Since Naive Bayes assumes that the conditional probabilities of the independent variables are statistically independent we can decompose the likelihood of a product of terms:

$$p(X|Cj) \propto \prod_{k=1}^{d} p(x_k|Cj) \quad (2)$$

And rewrite the posterior as:

$$p(Cj|X) \propto p(Cj) \prod_{k=1}^{d} p(x_k|Cj) \quad (3)$$

Using Bayes' rule above, we label a new case X with a class level Cj that achieves the highest posterior probability.

5.3. Decision Tree

Decision Trees are one of the most widely use data mining tool for classification purposes. A decision tree is used as a classifier for determining an appropriate action (among a predetermined set of actions) for a given case [4]. They are a non-parametric supervised learning method used for classification and regression purposes. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Structure of Decision Tree

A decision tree has three types of nodes:

Root node: Root node is the top most node. It has no incoming edge but zero or more outgoing edge.

Internal node: internal node has exactly one incoming edge and two or more outgoing edges.

Leaf node: Leaf node has exactly one incoming node and no outgoing edge.

Decision trees play an important role in the process of intrusion detection. From an intrusion detection perspective, decision trees can classify incoming packet as malicious, normal or any other category using information like source port, destination port, no. of SYN flags from a particular source to destination port(in case of SYN Flood) etc.

5.4. MLP -ANN

The most common and well-known Feedforward Neural Network (FFNN) model is called MLP. MLP has been successfully applied in a number of applications, including regression, classification, or time series prediction problems using simple auto-regressive models. It allows the data to flow in one direction from input nodes to output node. There is no feedback; it tends to be straight-forward networks that companion inputs with outputs. The MLP architecture can be clarified as the pattern of connections between the neurons in different layers: input layer, hidden layers, and output layers as shown in figure 3 below. Further the MLP architecture uses the training algorithm and transfer function for classification process. Sigmoid function is one of the most commonly used transfer function.

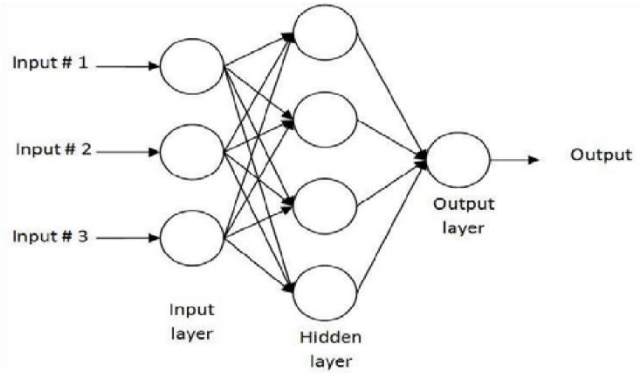


Fig. 3. MLP Structure

In order to understand the algorithm of the learning process on MLP, suppose that a given MLP has N neurons in the input layer and M neurons in the hidden layers, and one output neuron.

5.5. Support Vector Machine

SVM is one of the most popular machine learning algorithms for many applications, such as intrusion detection, spam filtering, and pattern recognition. There are various SVM formulations for classification, regression, and distribution estimation. As we mentioned above, in this study, the main goal is to classify each IP packet belong to the different classes.Therefore, we selected c-support vector classification (C-SVC) for training and testing datasets.

Let xi ∈ Rn, i =1, 2, … l, where l is the number of training examples, and an indicator vector y ∈ Rl where yi ∈ {-1, 1} be given training vectors. In the experiments, the dimension parameter n ranges from three to five, because we have prepared three datasets with different numbers of features. We use -1 to represent "Normal" for IP addresses from the victim pool and +1 as "DDoS *attack*" for IP addresses from the attacker pool, and we use C-SVC to solve the following the optimization problem .

$$\begin{aligned} \underset{w,b,\xi}{\text{minimize}} \quad & \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + C\sum_{i=1}^{l}\xi_i \\ \text{subject to} \quad & y_i(\boldsymbol{w}^{\mathrm{T}}\phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0,\ i = 1, 2 \ldots, l. \end{aligned} \quad (3)$$

Where ⍰(xi) maps xi into a higher dimensional space, and C is the regularization parameter, which must be greater than zero.

6. RESULTS

In this work, the dataset were reduced to 30% of the actual data and the experiments were performed on Ubuntu 16.04 LTS platform. A machine learning tool called WEKA version 3.8 was used for the application of the

classification techniques. To measure the efficiency of the algorithms each algorithm was trained on our dataset using 66% of the collected data and the 34% were used as a test data. We also used ten-folds cross validation technique but our previous partitioning works better.

6.1. Evaluation Metrics

The dataset is loaded into the WEKA tool in Ubuntu and various preprocessing techniques is applied selecting the data for training and testing phases. Here we eliminate the class label on the testing data by converting the ARFF file into the CSV file format. We have a total of 28 attributes including the class label. We evaluate the performance based on confusion matrix generated by these algorithms as shown in table a, b, c, d.

		Predicted	
		Positive	***Negative***
True	***Positive***	TP	FN
	Negative	FP	TN

Confusion Matrix

- Accuracy - measures the rate of the correctly classified attack instances of both classes.

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

- Precision: It is the ratio of the number of relevant attacks retrieved to the total number of irrelevant and relevant attacks retrieved. It is also called positive predictive, which can be calculated by the following equation.

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

- Recall: It is the ratio of the number of relevant attacks retrieved to the total number of relevant attacks. It is also called positive sensitivity value, which can be calculated by the following equation.

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

Table a

	Normal	UDP Flood	Smurf	SIDDOS	HTTP-FLOOD
Normal	6750	0	0	7	2
UDPFlood	695	6328	0	0	0
Smurf	264	0	136	21	0
SIDDOS	7	0	0	214	0
Http Flood	0	0	0	7	131

Confusion matrix for Naïve Bayes

Table b

	Normal	UDP-Flood	Smurf	SIDDOS	HTTP-Flood
Normal	66535	0	1208	7	9
UDP-Flood	692	6328	3	0	0
Smurf	255	0	9	21	136
SIDDOS	115	0	16	2120	0
Http Flood	0	0	0	86	1352

Confusion matrix for decision trees

Table c

	Normal	UDP-Flood	Smurf	SIDDOS	HTTP FLOOD
Normal	67756	0	0	3	1
UDP Flood	712	6400	0	0	0
Smurf	264	0	140	19	1
SIDDOS	8	0	0	215	0
Http Flood	0	0	0	80	1342

Confusion matrix for MLP-ANN

	Normal	UDP-Flood	Smurf	SIDDOS	HTTP-FLOOD
Normal	66530	0	1213	7	9
UDP Flood	690	6330	3	2	0
Smurf	250	0	140	53	136
SIDDOS	8	0	16	2120	0
Http Flood	0	0	0	90	1348

Table d

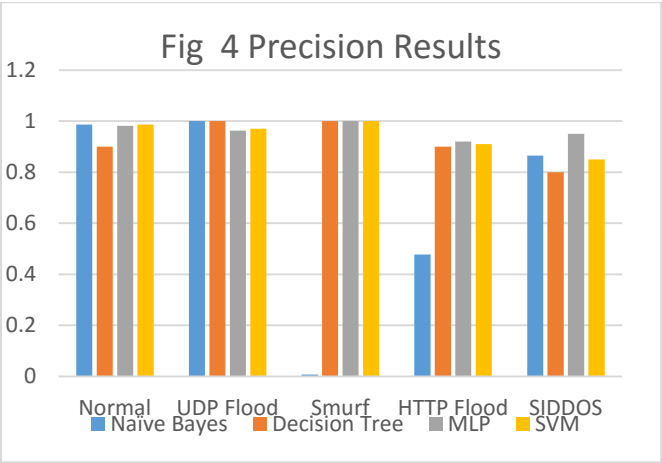
Confusion Matrix for SVM

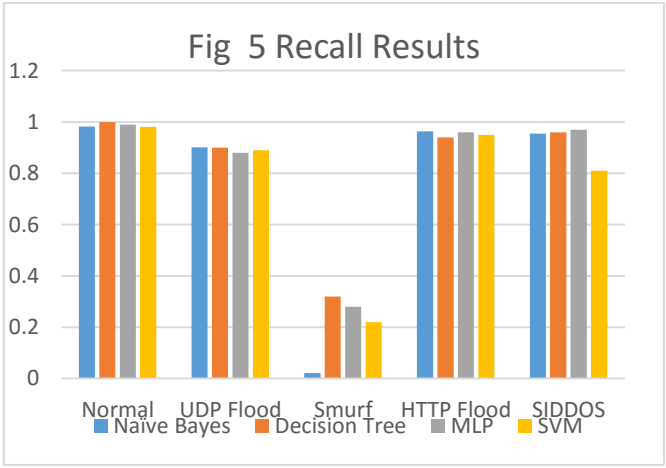
6.2. Result Discussion

The classifiers were evaluated and assessed using the confusion matrix based on the evaluation metrics listed in section VII (A). The resultant confusion matrices for Naïve Bayes, Decision Trees, MLP-ANN and SVM are shown in Tables a, b, c and d respectively. . From these confusion matrices we calculated the accuracy, precision and recall of the models. The overall accuracy was 96.89%, 98.89% and 98.91%, 92.31% for Naïve Bayes, Decision Trees, MLP-ANN and SVM correspondingly.

However, taking into consideration only the accuracy rate is not sufficient, especially when the data are imbalanced as in our case, where the number of instances in the normal class was much higher than the other classes. Therefore, the precision and recall were calculated for each class: Normal, UDP-Flood, Smurf, SIDDOS and the HTTP-FLOOD as shown in figure below.

	Naïve Bayes		Decision Trees		MLP		SVM	
Packets	**Precision**	**Recall**	**Precision**	**Recall**	**Precision**	**Recall**	**Precision**	**Recall**
Normal	0.986	0.982	0.9	1	0.981	0.99	0.98	0.981
UDP flood	1	0.901	1	0.90	0.963	0.88	0.97	0.89
Smurf	0.007	0.021	1	0.32	1	0.28	1	0.22
HTTP Flood	0.478	0.964	0.9	0.94	0.92	0.96	0.91	0.95
SIDDOS	0.865	0.955	0.8	0.96	0.95	0.97	0.85	0.81





7. CONCLUSION

In this paper, we collected a new dataset that includes modern types of attack, which were not been used in previous research. The dataset contains 27 features and five classes. The collected data has been recorded for different types of attack that target the Application and network layers. Four machine learning algorithms (Naïve Bayes, Decision Trees, MLP, and SVM) were applied on the collected dataset to classify the DDoS types of attack namely: Smurf, UDP-Flood, HTTP-Flood and SIDDOS. The MLP classifier achieved the highest accuracy rate.

The future work is to examine the different features for feature selection technique and include the more types of modern attacks in different OSI layers, such as the transport layer.

8. REFERENCES

[1] Mouhammd Alkasassbeh et al," Detecting Distributed Denial of Service Attacks Using Data Mining Techniques," International Journal of Advanced Computer Science and application, Vol. 7, Issue 1, pp. 436-445, January 2016.

[2] Sanguk Noh et al," Detecting Distributed Denial of Service (DDoS) Attacks through Inductive Learning", pp. 286–295, Springer-Verlag Berlin Heidelberg 2003.

[3] Sherif Saad, Issa Traore,et al." Detecting P2P Botnets through Network Behavior Analysis and Machine Learning" , Ninth Annual International Conference on Privacy, Security and Trust , 2011.

[4] Niharika Sharma, Amit Mahajan, Vibhakar Mansotra, "Identification and analysis of DoS attack Using Data Analysis tools," International Journal of Innovative Research in Computer and Communication Engineering," Vol. 4, Issue 6,pp. 11368-11375, June 2016.

[5] Vipin Das , Vijaya Pathak, Sattvik Sharma, Sreevathsan, MVVNS.Srikanth, Gireesh Kumar T," NETWORK INTRUSION DETECTION SYSTEM BASED ON MACHINE LEARNING ALGORITHMS" International Journal of Computer Science & Information

Technology (IJCSIT), Vol 2, No 6, pp 138-150, December 2010.

[6] Carl Livadas, Bob Walsh, David Lapsley, Tim Strayer," Using Machine Learning Techniques to Identify Botnet Traffic," Internetwork Research Department BBN Technologies

[7] Martin J Reed, Mohammed Alenezi," Methodologies for detecting DoS/DDoS attacks against network servers," The Seventh International Conference on Systems and Networks Communications, pp 92-98, 2012.

[8] Prajakta Solankar1, Prof. Subhash Pingale2, Prof. Ranjeetsingh Parihar3,"Denial of Service Attack and Classification Techniques for Attack Detection," (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (2) , pp 1096-1099, 2015.

[9] Bayu Adhi Tama, Kyung-Hyune Rhee," Data Mining Techniques in DoS/DDoS Attack Detection: A Literature Review," The 3rd International Conference on Computer Applications and Information Processing Technology (CAIPT 2015), Yangon, Myanmar, June 23-24, 2015.

[10] V. Hema and C. Emilin Shyni," DoS Attack Detection Based on Naive Bayes Classifier, "Middle-East Journal of Scientific Research 23 (Sensing, Signal Processing and Security), pp 398-40