Analysis and Detection of DDoS Attacks on Cloud Computing Environment using Machine Learning Techniques

Abdul Raoof Wani[1], Q.P. Rana[2], U. Saxena[3], Nitin Pandey[4]

[1,4]Amity University Noida, [2]Jamia Hamdard University, [3]CCFIS,
[1]wanirauf@gmail.com, [2]qprana@jamiahamdard.ac.in, [3]Usaxena@akcds.in, [4]Npandeyg@gmail.com

Abstract: ***The primary benefit of the cloud is that it elastically scales to meet variable demand and it provides the environment which scales up and scales down instantly according to the demand, so it needs great protection from DDoS attacks to tackle downtime effects of DDoS Attacks. Distribute Denial of Service attacks fall on the category of critical attacks that compromise the availability of the network. These attacks have become sophisticated and continue to grow at a rapid pace so to detect and counter these attacks have become a challenging task. This work was carried out on the owncloud environment using Tor Hammer as an attacking tool and a new dataset was generated with Intrusion Detection System. This work incorporates various machine learning algorithms: Support Vector Machine, Naïve Bayes, and Random Forest for classification and overall accuracy was 99.7%, 97.6% and 98.0% of Support Vector Machine, Random Forest and Naïve Bayes respectively***

Keywords: Machine learning, SVM, Naïve Bayes, Random Forest, DDoS, Cloud Computing, Weka

I. INTRODUCTION

Cloud computing is the environment which provides people platform to share resource, service, and information. It provides companies with a flexible architecture which in turn provides an efficient framework for computing. Due to the adoption of the cloud computing environment by a large number of organizations, there are a lot of risks and challenges which have come with it. Cloud computing servers as a utility available on the internet so the issues like user privacy, data leakage, and authentication remain one of the biggest challenges to cloud computing environment [1]. The novel architecture of cloud computing a lot of traditional issues have been effectively countered but its infrastructure and resources sharing has introduced the number of distinctive challenges. Cloud computing security includes the number of issues primarily in network and access control data, cloud infrastructure and it's not easy to enforce all security measures because of different security demands of different users [2].

Cloud computing services are often delivered by HTTP protocol providing access and reducing cost at the same time but these services are vulnerable to HTTP DDoS attacks. DDoS attack is intentionally performed by the malicious user to disrupt and degrade the service and resources of the legitimate user. In this type of attack, several negotiated computers are used to targeting network resources and servers leading to flooding of messages, malformed packets and connection requests resulting in the denial of service to the legitimate user. DDoS attacks on the cloud are highly sophisticated and request methods are created to find vulnerabilities and perform flooding attacks. The DDoS attacks are classified on the basis of network protocol and application. The network or transport level attacks are launched using the UDP, ICMP and TCP protocols. The HTTP DDoS attacks are performed on high and low rates scenarios each of them having a significant impact on the victim. The high rate attack floods the victim with a large number of requests while in the low rate scenario the victim is compromised by slow and compromised requests which results in the exhaustion of resources [3].

The DDoS attacks on the cloud computing environment are mainly application layer which sends out requests following the communication protocol which are then hard to distinguish in the network layer because their pattern matches the legitimate requests thus making the traditional defence systems not applicable. DDoS flooding attacks on cloud can be of various categories like session and request flooding attacks, slow response and asymmetric attack. All these flooding attacks generate traffic which resembles that of a legitimate user which becomes tougher for the target to distinguish between attack and legitimate traffic thus blocking the services for the legitimate user.

II. RELATED WORK

The recent increase of DDoS attacks has enticed a substantial interest by the researchers because they have avoided traditional network-based detection mechanisms. There is at least one DDoS attack reported in 20% of enterprises in the world [4]. DDoS attacks in the first quarter of 2018 were registered against targets in 79 countries and 84 in the previous quarter and the vast majority around 95% happened in the top 10 countries [5]. Various techniques have been developed in order to prevent DDoS attacks on the cloud like challenge answer, unseen servers, preventive access, and resource bounds. Other techniques which are being used to counter DDoS attacks are puzzle based methods and like CAPTCHA puzzle which offers a simple approach for attack mitigation but

978-1-5386-9346-9/19/$31.00 ©2019 IEEE

Authorized licensed use limited to: MULTIMEDIA UNIVERSITY. Downloaded on July 24,2024 at 13:26:17 UTC from IEEE Xplore. Restrictions apply.

its ineffective shown by recent studies[6]. The IP address monitoring is one of the techniques which is being constantly used to counter DDoS attacks [7]. A digital signature for network flow investigation using meta-heuristic methods was created to investigate the abnormal traffic which showed the improved accuracy in the DDoS detection but the model failed to detect normal DoS attacks [8]. Other techniques like IP traceback defence mechanism is used which comprises of packet marking, reconstruction procedure and attack source identification using entropy [9].

Cloud computing environment faces with a lot of HTTP DDoS attacks and a lot of research has already been going in this field and new techniques have been developed in order to counter such kind of attacks. Dantas Y et al. [10] developed a technique called SeVen based on the Adaptive Selective Verification which is used to counter network layer DDoS attacks. The technique works on the concept of the notion of a state but the application layer DDoS attacks do not possess a notion of state. This mechanism automatically fails against HTTP Post Flooding attack because the enormous amount of reflectors are used to send payloads. A technique named Protocol Free Detection was developed which analyses the network flow of the cloud services by studying the basic traffic co-relation. The packed rate is sampled using an upstream router and flow correlation coefficient is used to test the flow of the traffic which gives the current FCC value as well as previous information. The deploying of Protocol Free Detection in cloud makes it vulnerable to HTTP DDoS attacks [11].

Various machine learning and data mining approaches have been used recently in the prevention and detection of DDoS attacks. Alkasassbeh et al.[12] has used the modern database of DDoS attacks which is collected at different network layers like SIDDoS and HTTP flood. The work focuses on a comparative analysis where 27 features are taken into consideration and machine learning techniques like Multilayer Perceptron, Random forest, and Naïve Bayes are used to calculate the accuracy. Carl Livadas et al [13] used machine learning techniques to identify traffic at command and control center. The author distinguished the traffic into IRC traffic and real IRC traffic by using machine learning techniques Naive Bayes, J48 and Bayesian network classifiers. The performance of Naive Bayes is better than the other two techniques but J48 performs better for real life IRC and non-IRC flows. Bayu Adhi Tama et al [14] provided a detailed literature survey of 35 papers where he classified the papers concerning DoS and DDoS attacks which used data mining techniques. The literature review provided deep analysis of DDoS attacks which used the data mining functions to classify data the basis of association, clustering, and hybrid methods

III. EXPERIMENTAL SETUP

The DDoS attacks were performed on the owncloud environment which is a free and open source platform for cloud users. The server edition of Owncloud Platform is open-source which works as a private server and allows operate, modify and install without any charge. Owncloud platform was created on Ubuntu 16.04 operating system with MySQL database, Apache web server and PHP 7.0.

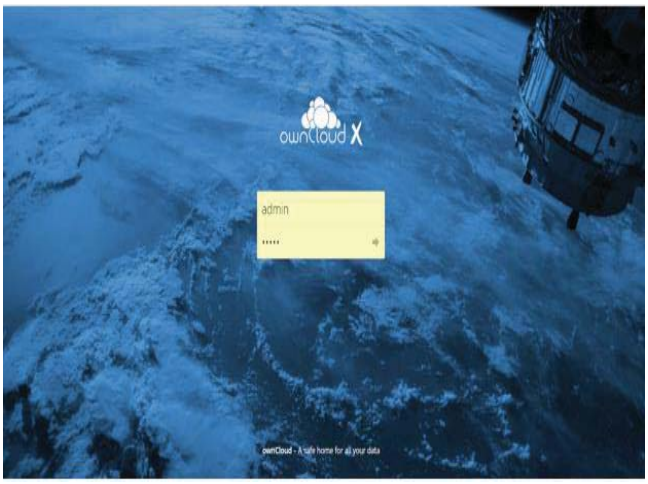

Fig. 1. Shows the Interface of Owncloud Environment.

Attack Generation

The DDoS attack was generated in a secure environment using Tor Hammer tool. The attacking machine comprised of operating system **Kali 2018.2 with** Kernel 4.15.0, GNOME 3.28.0. During the execution of DDoS attack the specified program generated traffic at sink nodes and during this process tshark, traffic protocol analyser recorded both the normal and suspicious traffic. The recorded traffic was then transferred to the server.

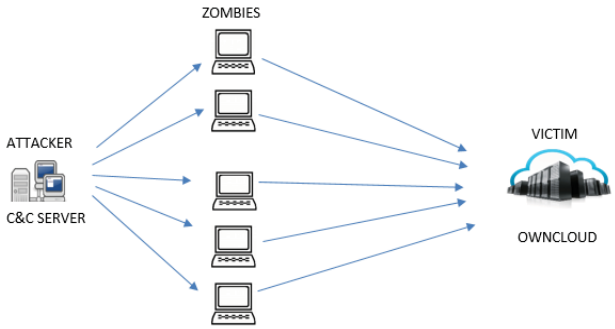

Fig. 2. Depicts the DDoS Attack on Owncloud

B. Attack Detection and Dataset Collection

Intrusion Detection System SNORT was given an input of files which were taken from the server. This open source rule-based tool is used to detect all these attacks but the default rules were changed to identify the DDoS attacks. The output from the SNORT was directed by defining the required tuples.

Authorized licensed use limited to: MULTIMEDIA UNIVERSITY. Downloaded on July 24,2024 at 13:26:17 UTC from IEEE Xplore. Restrictions apply.

1. Rules for Attack Detection

TABLE I: Rules Define for Attack Detection

Step1	Weight (Testing interval time)< (Normal classifier weight) ≤(weight of Attacker)	A(w)< T(w)≤ N(w)	Normal
Step 2	2.1 Normal classifier Similarity of testing record is more than 99 percent 2.2 Suspicious classifier Similarity of testing record is more than 99 percent	Test: If the Similarity of normal ≥ 99% Test: If the Similarity of attack ≥99%	Normal Suspicious
Step 3	Normal classifier Similarity is more than the Similarity of Attack classifier.	Test: Similarity (Normal attack) >(test Attack)	Normal
Step 4	All above conditions don't match		Unknown

Output alert csv: date, duration, proto, source ip, dest ip, src pt, dst pt, packets, class, bytes, The alert generated from SNORT consist of 10 Tuples
"Proto","date,""duration","source ip","dest ip","src pt","dst pt","packets", "class" ,"bytes".

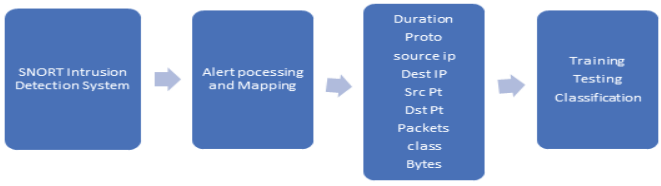

Fig. 3. Process of Attack detection

The sample of the output is as following

Date first	Duration	Proto	source ip	Src Pt	Dest IP	Dst Pt	Packets	Bytes	class
43:57.2	81412.7	HTTP	14.139.238.86	8082	14.139.238.89	56978	3057	2000	normal
43:57.2	81412.7	HTTP	14.139.238.86	56978	14.139.238.89	8082	4748	2500	normal
43:26.1	81504.79	HTTP	14.139.238.86	8082	14.139.238.89	56979	8639	9000	normal
43:26.1	81504.79	HTTP	14.139.238.86	56979	14.139.238.89	8082	12024	10000	normal
17:09.0	82100.69	HTTP	14.139.238.86	8082	14.139.238.89	51649	11012	27200	normal
17:09.0	82100.69	HTTP	14.139.238.86	51649	14.139.238.89	8082	14186	12000	normal
46:37.3	83640.14	HTTP	14.139.238.86	8082	14.139.238.89	37039	9974	31900	normal
46:37.3	83640.14	HTTP	14.139.238.86	37039	14.139.238.89	8082	16476	1700	normal
02:42.7	86303.3	HTTP	14.139.238.86	8082	14.139.238.89	48380	10178	25200	normal
02:42.7	86303.3	HTTP	14.139.238.86	48380	14.139.238.89	8082	14552	2600	normal
17:17.0	87709.79	HTTP	14.139.238.86	8082	14.139.238.89	50807	17031	52500	normal
17:17.0	87709.79	HTTP	14.139.238.86	50807	14.139.238.89	8082	24706	3500	normal
44:10.0	183356.4	HTTP	14.139.238.86	8082	14.139.238.89	60803	24161	64800	normal
44:10.0	183356.4	HTTP	14.139.238.86	60803	14.139.238.89	8082	34136	98000	normal
43:39.0	183418.5	HTTP	14.139.238.86	8082	14.139.238.89	60802	13266	33000	normal
43:39.0	183418.5	HTTP	14.139.238.86	60802	14.139.238.89	8082	20751	5800	normal
00:10.7	0	HTTP	14.139.238.86	8830	14.139.238.89	23	1	46	suspicious
00:10.7	0	HTTP	14.139.238.86	23	14.139.238.89	8830	1	40	suspicious
00:09.2	15.42	HTTP	14.139.238.86	22	14.139.238.89	4589	19	3093	suspicious
00:09.2	15.42	HTTP	14.139.238.86	4589	14.139.238.89	22	13	2843	suspicious
00:27.4	0	HTTP	14.139.238.86	18816	14.139.238.89	23	1	46	suspicious
00:27.4	0	HTTP	14.139.238.86	23	14.139.238.89	18816	1	40	suspicious

Fig. 4. Sample Output of Dataset Generated.

2. Dataset Features

The data set collected contains 9 attributes which are tabulated in Table II.

TABLE II: Dataset features

Feature	Description
Duration	Duration of the flow
Proto	Type of Protocol
source Ip	Internet Protocol Address (Source)
Dest IP	Internet Protocol Address (Destination)
Src Pt	Port (Source)
Dst Pt	Port (Destination)
Packets	Transmitted packets
class	Attack classification Labels
Bytes	Number of transmitted bytes

IV. CLASSIFICATION

We investigated and tested three Machine learning algorithms Random, Forest, Naïve Bayes and Support Vector Machine for the classification of data. These algorithms were chosen on the basis of their efficient performance and application in the field of network security. We performed a comparison and Analysis of Accuracy, Precision, Recall on DDoS packets and Normal packets.

The dataset is loaded into the WEKA tool in Ubuntu and various pre-processing techniques were 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 9 attributes including the class label. We evaluate the performance based on confusion matrix generated by these algorithms.

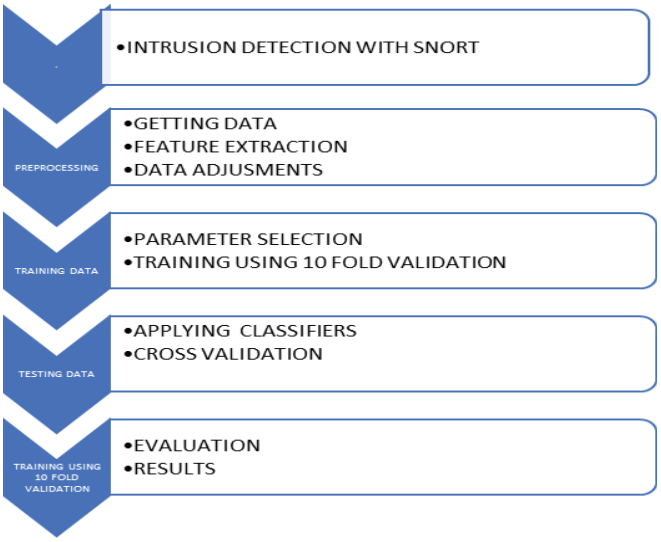

Fig. 5. Classification of Attacks Using Machine learning

Authorized licensed use limited to: MULTIMEDIA UNIVERSITY. Downloaded on July 24,2024 at 13:26:17 UTC from IEEE Xplore. Restrictions apply.

1). Support Vector Machine (SVM)

Support Vector Machine (SVM) is one of the standard machine learning algorithms used in pattern recognition, spam filtering and anomaly network intrusion detection. SVM can learn the pattern in gives accurate classification by using class labels. The accurate classification is achieved by training machine to classify unknown samples with the training dataset model. SVM has the capability to find the global optimal solution by performing the linear separation finding an optimal hyperplane that separates two classes. The closest data to the hyperplane are support vectors and by getting the features the predicted class is declared.

Given the training dataset of n points of the form

$$(\overrightarrow{x1}, y1), \ldots, (\overrightarrow{xn}, yn)$$

Where "yi" are either 1 or -1 each which indicates the class to the point $\overrightarrow{xi}$ belongs.

The hyper lane can be written as the set of points $\overrightarrow{xi}$ satisfying

$$\vec{w}.\vec{x} - b = 0,$$

Where $\vec{w}$ is not necessarily normalized

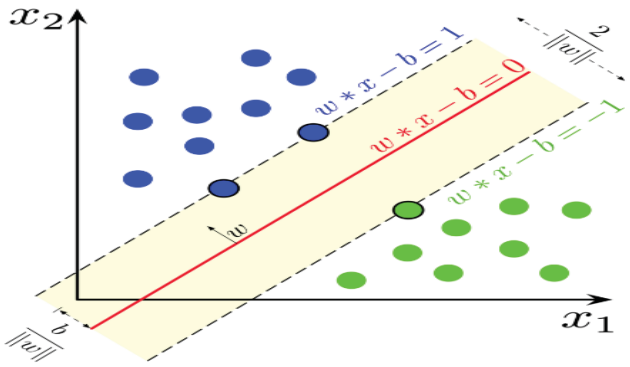

Fig. 6. Maximum Hyperplane Margin [15].

$\vec{w}.\vec{x} - b = 1$ ("Anything on or above this boundary is of one class, with label 1")
$\vec{w}.\vec{x} - b = -1$("Anything on or below this boundary is of the other class, with label -1")

2). Naïve Bayes

Naive Bayes is one of the famous probabilistic models that calculate probabilities for each class in and determines classifying and learning to predict the values of the new class. Given a problem instance to be classified represented by vector x= $(x1 \ldots . xn)$ representing n independent variables which are assigned to instance probabilities
$p(Ck|x1 \ldots \ldots, xn)$

For each of $\boldsymbol{K}$ possible outcomes or classes$\boldsymbol{Ck}$.

The conditional probability can be further decomposed as

$$p(Ck|X) = \frac{p(Ck)p(X|Ck)}{p(X)}$$

Or it can simply be written as

$$posterior = \frac{prior * likelihood}{evidence}$$

The joint model can be written as

$$p(Ck|x1, \ldots . xn)\alpha\, p(Ck, x1 \ldots \ldots, xn) =$$

$$= p(Ck)p(x1|Ck)p(x2|Ck)p(x3|Ck) \ldots .$$

$$p(Ck)\prod_{i=1}^{n} p\,(xi|Ck),$$

3). Random Forest

Random Forest machine learning algorithm which is flexible and easy to use and also produces great results most of the times. It is widely used because of its simplicity and the fact that it can be both used in classification as well as regression. It functions by building a multitude of decision trees during the training process resulting in the output in the form of classification of individual trees.

In the training algorithm of Random Forest, bootstrap aggregation technique is used. Given a training set X=$(x1, \ldots \ldots xn)$ with Y=$(y1 \ldots . . yn)$ bagging B times by selecting a random sample and applies tress to these samples.
For $b = (1, \ldots \ldots, B)$
Training Sample Replacement X, Y=(Xb)(Yb)

Training classification of regression tree of $fb\ on\ Xb, Yb$
Predictions (average) from all individual regression trees on x'.

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} fb(x')$$

Estimate of uncertainty can be made as the standard deviation of the prediction of all individual regression trees on x'.

$$\sigma = \sqrt{\frac{\sum_{b=1}^{B}(fb(x') - \hat{f})^2}{B-1}}$$

V. PERFORMANCE PARAMETER CALCULATION

The purpose of this approach is to classify the traffic data whether it is suspicious, normal or unknown and the results are obtained by using the following performance metrics.

TABLE III: Performance parameter calculations

'TP'(True Positive)	"The total sum of suspicious transactions identified, which are truly suspicious"
'FP' (False Positive)	"The total sum of normal transactions identified, which are truly suspicious"
'TN'(True Negative)	"The total sum of normal transactions identified, which are truly normal"
'FN'(False Negative)	"The total sum of suspicious transactions identified, which are actually normal."

Authorized licensed use limited to: MULTIMEDIA UNIVERSITY. Downloaded on July 24,2024 at 13:26:17 UTC from IEEE Xplore. Restrictions apply.

- Precision: Positive labels on data are shown by the precision of class agreement specified via classifier
- Recall: The efficiency of positive labels identified by a classifier.
- Specificity: Effectiveness of a classifier to identify negative labels.
- Accuracy: Overall efficiency of a classifier is shown by accuracy.
- F-measure: Relationship of data's positive labels with those given by a classifier

TABLE IV: CONFUSION MATRIX

Recall	("TP")/("TP+FN")
Precision	("TP")/("TP+FP")
Accuracy	("TP+TN")/("TP+TN")+("FP+FN")
Specificity	("TN")/("TN+FP")
F measure	("2TP")/("2TP+FP")=("FN")

VI. RESULT AND DISCUSSION

The database generated by Snort was classified by machine learning algorithms Support Vector Machine, Random Forest and Naïve Bayes in weka tool. The confusion matrix was used to evaluate the classifiers and the results are shown in Table V. The overall accuracy was 99.7%, 97.6% and 98.0% of Support Vector Machine, Random Forest and Naïve Bayes respectively. The precision, recall and specificity are equally important because of the imbalanced data and should be put into consideration. Out of the three algorithms used SVM shows the better results in terms of accuracy, recall .precision, specificity and f measure closely followed by Random Forest.

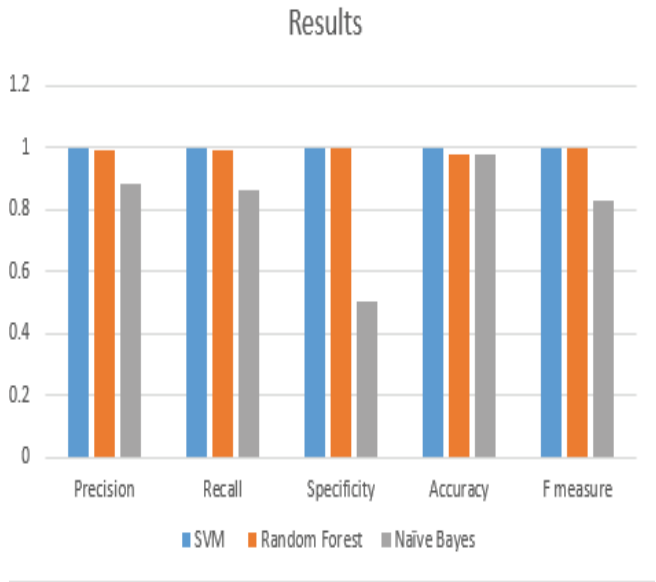

Fig. 7. Shows the results of SVM, Naïve Bayes, Random Forest

TABLE V: RESULTS

	SVM	Random Forest	Naïve Bayes
Recall	0.998	0.993	0.860
Precision	0.998	0.992	0.881
Accuracy	0.997	0.976	0.980
Specificity	0.996	0.995	0.505
F measure	0.998	0.996	0.826

VII. CONCLUSION

The dataset contains 9 feature and 4 classes. Machine learning algorithm were applied to the data set namely Support Vector Machine, Naïve Bayes and Random Forest. SVM algorithm showed greater accuracy and precision from Naive Bayes and Random Forest. The SVM model's high performance and can be used in Intrusion detection Purposes. Future work will include more type of attacks and different feature selection techniques.

REFERENCES

[1] Wani, Abdul Raoof, Q. P. Rana, and Nitin Pandey. "Cloud security architecture based on user authentication and symmetric key cryptographic techniques." *Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), 2017 6th International Conference on*. IEEE, 2017.

[2] Wani, Abdul Raoof, Q. P. Rana, and Nitin Pandey. "Analysis and Countermeasures for Security and Privacy Issues in Cloud Computing." *System Performance and Management Analytics*. Springer, Singapore, 2019. 47-54.

[3] E.Cambiaso, G. Papaleo, and M. Aiello, "Taxonomy of Slow DoSAttacks to Web Applications," in *Recent Trends in Computer Networks and Distributed Systems Security*, vol. 335 of *Communications in Computer and Information Science*, pp. 195–204, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[4] Kaspersky Labs, Global IT security risks survey 2014 - distributed denial of service (DDoS) attacks, 2014, (http://media.kaspersky.com/en/ B2B- International- 2014- Survey- DDoS- Summary- Report.pdf).

[5] https://securelist.com/ddos-report-in-q1-2018/85373/. 2018

[6] S.Y. Nam, T. Lee, Memory-efficient IP filtering for countering DDoS attacks, in: Proceedings of the 12th Asia-Pacific Network Operations and Management Conference on Management Enabling the Future Internet for Changing Business and New Computing Services, APNOMS'09, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 301–310.

[7] Maciá-Fernández, Gabriel, Rafael A. Rodríguez-Gómez, and Jesús E. Díaz-Verdejo. "Defense techniques for low-rate DoS attacks against application servers." *Computer Networks*54.15 (2010): 2711-2727.

[8] Salmen, Fadir, et al. "Using Firefly and Genetic Metaheuristics for Anomaly Detection based on Network Flows." *AICT: The Eleventh Advanced International Conference on Telecommunications*. 2015.

Authorized licensed use limited to: MULTIMEDIA UNIVERSITY. Downloaded on July 24,2024 at 13:26:17 UTC from IEEE Xplore. Restrictions apply.

[9] Vijayalakshmi, M., S. Mercy Shalinie, and A. Arun Pragash. "IP traceback system for network and application layer attacks." *Recent Trends In Information Technology (ICRTIT), 2012 International Conference on*. IEEE, 2012.

[10] Dantas, Yuri Gil, Vivek Nigam, and Iguatemi E. Fonseca. "A selective defense for application layer ddos attacks." *Intelligence and Security Informatics Conference (JISIC), 2014 IEEE Joint*. IEEE, 2014.

[11] Xiao, Le, et al. "A protocol-free detection against cloud oriented reflection DoS attacks." *Soft Computing* 21.13 (2017): 3713-3721.

[12] Alkasassbeh, Mouhammd, et al. "Detecting distributed denial of service attacks using data mining techniques." *International Journal of Advanced Computer Science and Applications* 7.1 (2016).

[13] Livadas, Carl, et al. "Usilng machine learning technliques to identify botnet traffic." *Local Computer Networks, Proceedings 2006 31st IEEE Conference on*. IEEE, 2006.

[14] Tama, Bayu Adhi, and Kyung-Hyune Rhee. "Data mining techniques in DoS/DDoS attack detection: A literature review." *Information (Japan)* 18.8 (2015): 3739.

[15] Anjum Zameer Bhat; Dalal Khalfan Al Shuaibi; Ajay Vikram Singh, "Virtual private network as a service — A need for discrete cloud architecture", 2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO) at AUUP, NOIDA, India, September 07-09, Year: 2016 Pages: 526 – 532.

[16] Ajay Vikram Singh, Moushumi Chattopadhyaya, "Mitigation of DoS Attacks by Using Multiple Encryptions in MANET", 2015 4th IEEE International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), 2015 at AUUP, NOIDA, India, September 02-04, 2015

[17] Abhishek Bhardwaj, Subhranil Som, S. K. Muttoo, (2018) "HS1-RIV: Improved Efficiency for Authenticated Encryption", International Journal of Engineering and Technology (UAE), Scopus Indexed, DOI: 10.14419/ijet.v7i2.7.10871, Vol. 7, Issue 2.7, Page 502-506, https://www.sciencepubco.com/index.php/ijet/article/view/10871 March 2018.

[18] Iti Burman, Subhranil Som, Syed Akhter Hossain, (2018) "Meta-Analysis of Psychometric Measures and Prediction of Student's Learning Behavior using Regression Analysis and SVM", Journal of Advance Research in Dynamical & Control Systems, Scopus Indexed, Vol. 10, 02-Special Issue, ISSN 1943-023X, Page 291-298.

[19] https://en.wikipedia.org/wiki/Support_vector_machine#/media/File:SVM_margin.png

Authorized licensed use limited to: MULTIMEDIA UNIVERSITY. Downloaded on July 24,2024 at 13:26:17 UTC from IEEE Xplore. Restrictions apply.