Machine Learning Solutions for Analysis and Detection of DDoS Attacks in Cloud Computing Environment

Abdul Raoof Wani, Q. P. Rana, Nitin Pandey

Abstract— ***Distributed denial of service is a critical threat that is responsible for halting the normal functionality of services in cloud computing environments. Distributing Denial of Service attacks is categorized in the level of crucial attacks that undermine the network's functionality. These attacks have become sophisticated and continue to grow rapidly, and it has become a challenging task to detect and address these attacks. There is a need for Intelligent Intrusion detection systems that can classify and detect anomalous behavior in network traffic. This research was performed on the cloudstack environment using Tor Hammer as an attacking mechanism, and the Intrusion Detection System produced a new dataset. This analysis incorporates numerous algorithms of machine learning: k-means, decision tree, Random Forest, Naïve Bayes, Support Vector Machine and C4.5***

Keywords: Machine learning, K-Means, Decision Tree, C4.5, SVM, Naïve Bayes, Random Forest, DDoS, Cloud Computing;

I. INTRODUCTION

Cloud computing provides a flexible and on-demand virtual pool of configurable computing resources with limited management effort or service provider interactions at all times and anywhere.Cloud computing platform faces a lot of challenges and security still remains one of the biggest challenges. There are many security-related attacks that are well mitigated in non-cloud infrastructures and these solutions are now being applied in cloud computing environments [1]. The risks and challenges of adopting a cloud computing environment are very high due to the migration of a lot of companies to the platform. Many traditional issues have been effectively addressed with cloud computing's novel architecture, but its infrastructure and resource sharing has brought with it a number of distinctive challenges. Networking, access control data, and cloud infrastructure have a number of issues, and security solutions are needed at each cloud infrastructure level[2].

Out of the attacks, DDoS attacks have been much visible in the cloud computing environment. Cloud attacks are mainly application layers that send communication protocol requests that are difficult to identify in the network layer as their pattern corresponds to legitimate requests so that traditional defense systems cannot identify them. These attacks aim to overload the victim which results in flooding of packets making it incompetent of performing normal services for legitimate users. In a distinctive DDoS attack, the relay host gets compromised by the attacker which then uses machines called zombies that spread attack packets to the victim [3].

II. RELATED WORK

Many techniques for detecting and analyzing DDoS attacks have been implemented in recent times. Most of those detection initiatives rely on the choice of feature selection from the captured IP packet. The latest escalation of DDoS attacks on the application layer has drawn considerable interest from a research community[4]. These research approaches can be commonly divided into several groups: approaches based on applications, approaches based on puzzles and approaches based on network traffic characteristics. This work is based on machine learning methods to enhance the precision of identification of DDoS attacks, false-positive rates. Many prior works have been dedicated to enhancing DDoS attack detection efficiency. We summarize some of the recent work on detecting DDoS attacks in this section.[5]

Fadir Salmen[6] et al. designed the community level digital signature for flow evaluation with the aid of the usage of two meta-heuristic techniques. In order to investigate the behavior of designed processes, they injected odd site visitors and showed progressed accuracy inside the detection of DDoS attacks, however the important model can't detect DoS attacks. Liu et al.[7] implemented two coordinated defenders Egress Filter and Behavior Analyzer in defending systems against DDoS attacks, The counterattack mechanism provides separate services for each use, depending on their degree of deviation. A single classifier based on SVM is used to detect anomalies in [8] where training data was mapped into a unique feature space. Several techniques have been introduced to extract valuable features from this dataset and then some classifiers, such as statistical, machine learning, pattern recognition, are trained with portions of this dataset. Dantas Y et al.[9] introduced an Adaptive Selective Verification (ASV)-based protection mechanism against the HTTP POST Flooding attack. Since ASV has been meant to mitigate attacks on network layer DDoS, it assumes that communications are regular amongst customer-server stateless syn-ack interactions. However, this is not sufficient as the protocol to mitigate Application Layer DDoS attacks.

Recently various approaches of data mining and machine learning were used to prevent DDoS attacks. 27 characteristics have been taken into account by Alkasassbeh et al[5 in a new dataset with current DDoS attacks in network layers including (SIDDoS,

Revised Manuscript Received on January22, 2020.

Abdul Raoof Wani, Scholar, Amity Institute of Information Technology, Noida, Uttar Pradesh, India.

Q. P. Rana, Assistant Professor, Jamia Hamdard University New Delhi, India.

Nitin Pandey, Assistant Professor, University of Uttar Pradesh, Amity Institute of Information Technology, Noida, Uttar Pradesh, India.

HTTP FloodThis paper focuses especially at the comparative evaluation of the unique classifiers used within the class and the determination of the uncertainty matrix of every approach used. The approach involves common techniques in machine learning, including Naïve Bayes, Multilayer Perceptron (MLP) and Random Forest. Among these approaches is MLP's highest accuracy rate (98.63 per cent)[6].

III. EXPERIMENTAL SETUP

DDoS attacks have been carried out on the cloudstack. Cloludstack software is free and open-source software that is used as a service to build and manage infrastructure in cloud computing. The VMware VSphere and VMware ESXi hypervisors were used in creating host and Vcenter server. The cloud stack infrastructure consisted of management node of Intel Xeon 3.0 GHz with 16 GB of RAM, 2 host nodes consisting of Intel Xeon 2.6 GHz each with ESXI 6.5 installed on them. The VSphere server consists of Intel Xeon 3.0 GHz with 24 GB of RAM and VSphere Version 6.5.0 installed on it accessed via vSphere Web Client. The primary storage was installed in a cloud stack management server while as a separate NFS server was installed on a different machine for secondary storage using openfiler operating system.

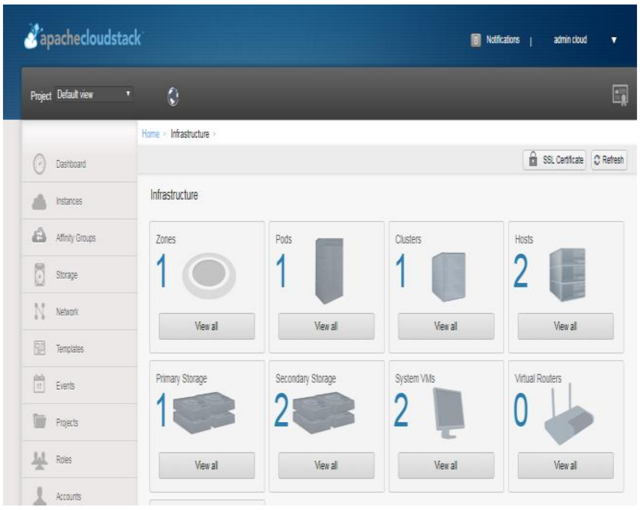

Figure. 1 Shows Cloudstack Environment

A. Attack Generation

The DDoS attack was performed using the Tor Hammer tool in a secure environment. The attacking platform included Kali operating system 2018.2 with Kernel 4.15.0, GNOME 3.28.0. The specified program generated traffic on sink nodes during the execution of the DDoS attack, and the traffic protocol analyzer collected both normal and suspicious traffic during this tshark process and subsequently sent the traffic collected to the server.

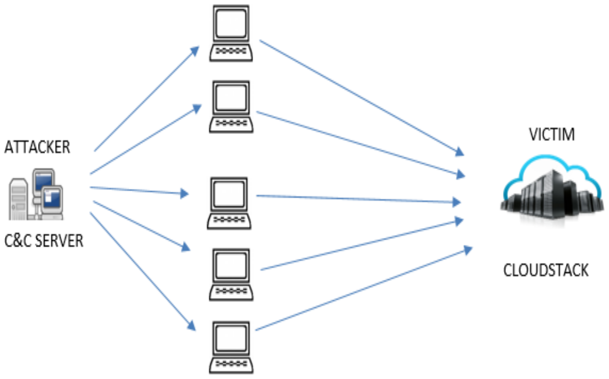

Figure 2. DDoS Attack on CloudStack

B. Attack Detection and Dataset Collection

Intrusion Detection System SNORT obtained an input of server records This open-source intrusion detection tool was used to detect the attack by analyzing real-time traffic in order to identify DDoS attacks by changing and modifying rules. The SNORT output was managed by setting the necessary tuple.

Table1:Rules for Attack Detection

1.Weight (Test time) < (Normal weight of the classifier) ≤ (Attacker weight)	Normal
2.1 Normal classifier Testing record similarity is more than 99% 2.2 Suspicious classifier Testing record similarity is more than 99%	Normal Suspicious
3. Normal classification Similarity is more than the Attack Classification Similarity.	Normal
4. None of the above conditions match.	Unknown

The input of the snort will be the dump file which will generate alerts. Recorded alerted will be separated by a comma and this csv will be stored for further processing. The alerts generated from snort consists of 21 tuples "Duration","count","Proto","sourceIp","DestIP","SrcPt","DstPt","Packets","class","Bytes","Flags","AttackType","AttackID","Attack_Description","wrong_fragment","Urgent","Hot","Error-rate","Rerror-rate","Srv_error_rate" "Srv_Rerror rate"

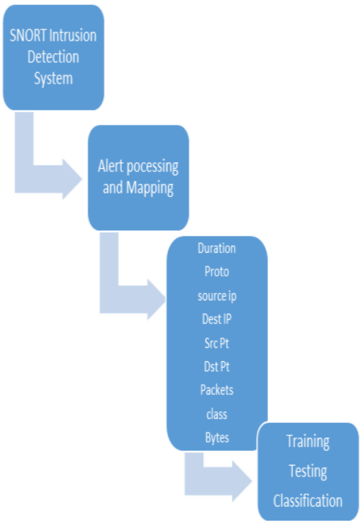

Figure. 3 Attack detection Process

C. Dataset Features

Table 2 The collected data set contains 21 attributes that are shown below.

Feature	Description
Duration	Duration of the flow
Count	Start time flow first seen
Flags	TCP Flags concatenation
AttackType	Type of Attack
AttackID	Unique attack id
Attack_Description	Additional information about the set attack parameters
wrong_fragment	Wrong Fragments
Urgent	Urgent packets
Hot	Hot Indicators
Protoc	Type of Protocol
Source Ip	Source Ip
Destt IP	Destination Ip
Srce Pt	SourcePort
Dstt Pt	DestinationPort
Packet	Packets Transmitted
Class	Classification Labels
Byte	Transmitted bytes
Error-rate	Percentage of SYN error connections
Rerror-rate	Percentage of REJ error connections
Srv-error-rate	Percentage SYN errors connections with same service
Srv-Rerror-rate	Percentage of REJ error with same connections

IV. CLASSIFICATION

Six Machine learning algorithms k-means, Decision tree, Naïve Bayes, C4.5, Support Vector Machine and Random Forest for data classification were investigated and tested. These algorithms were selected based on their effective performance and implementation of network security. The precision, accuracy, recall of DDoS packets and regular packets were compared and analyzed

The dataset generated was then preprocessed in weka. The data set was divided into training and testing phases. Here the ARFF file is converted to CSV file format, we remove the class label on the test information. We have 21 attributes in total, including the class label. We assess the efficiency of these algorithms based on the confusion matrix generated

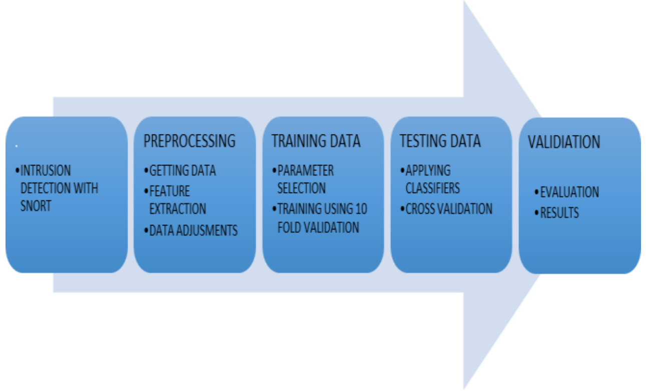

Figure. 4 Classification of attacks using machine learning

1. K-Means

K-Means is a machine learning algorithm in which a large number of observations are taken to form a small number of clusters. This technique divides N observations with P dimensions into K clusters in order to minimize the total of squares in the cluster and assign the number of clusters to be identified. The process then divides the data into a set of cluster centers using spherical clusters which in turn allocates each observation to a cluster thus forming cluster centers that keep the process repeating.

Let us suppose we have N observations (rows) separated into K groups. The cluster at the kth point will contain observations as nk.P variables are designated in each row. A value missing in the variable (ith) of the row (jth) of the group (kth) is labeled by δijk.

The standardized data elements are represented as (zij) and the data is standardized by variable mean which is then divided by the standard deviation.

The within-cluster sum of squares is used to get the goodness of fit criterion in order to compare various cluster configurations.

2. Decision Tree

Decision tree is a very powerful tool for classification and prediction. Due to its nature, it has been widely used to represent the classification of models because of the advantage of creating a coherent classification and accomplish the accuracy level. The goal of the decision tree is to find the optimal decision by minimizing the generalization error by inducing algorithms that are automatically constructed for a given dataset.

Due to their non-parametric nature decision trees can be applied either to classification or regression tasks. Partitioning the training data of pre-classified instances improve homogeneity by partitioning into smaller fragments or child partitions. The decision tree uses the splitting criteria and various induction algorithms to calculate the variants of impurity the entropy of splitting its child partitions. A new instance is classified with initialization at the root of a decision tree after that the attribute to that specific node is tested. The outcome of this test helps down the tree through the branch comparative to the attribute value of the given instance repeating the process until a leaf is met

3. Support Vector Machine (SVM)

A Support Vector Machine (SVM) is a discriminative classifier formally described by using a setting apart hyperplane. Support vector system technique creates a hyperplane in boundless dimensional area, which is type and regression. Using class labels, SVM can learn the pattern and classify it accurately. Through training machine to identify unknown samples with the training data set template, the correct classification is achieved. SVM can find the optimum solution by finding the ideal hyperplane separating the two classes. Support vectors are the hyperplane's closest data and the predicted class of features is declared.

Based on the training data set of n points

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

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

DOI: 10.35940/ijeat.B3402.029320

4. Naïve Bayes

Naive Bayes is one of the common probabilistic models that measures probabilities in each class and determines how new class values can be predicted. Problem instance to be listed representing the vector x= (x1..... xn) representing n independent variables assigned to the probabilities of example
$p(Ck|x1\ldots..,xn)$
$p(Ck|X)=p(Ck)p(X|Ck)p(X)$
Or it can simply be written as
$posterior=prior*likelihoodevidence$
The joint model can be written as
$p(Ck|x1,\ldots.xn)\alpha\ p(Ck,x1\ldots..,xn)=$
$=p(Ck)p(x1|Ck)p(x2|Ck)p(x3|Ck)\ldots.\ p(Ck)\prod pni=1(xi|Ck),$

5. Random Forest

Random Forest technique is versatile, user-friendly and most of the time also generates excellent results. It is commonly used for its simplicity and ability to work in both classification and regression problems. It works by constructing a number of decision trees during the training phase leading to output in the form of individual trees classification and using the Bootstrap aggregation method during training.

Given X=$(x1,\ldots\ldots xn)$ with Y=$(y1\ldots..yn)$
For $b=(1,\ldots\ldots,B)$
Training Sample Replacement X, Y=(Xb)(Yb)
$fb\ on\ Xb, Yb$

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

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

6. C4.5

C4.5 is the suite of decision tree algorithms that is used in problem and data classifying in machine learning. Supervised learning is the main goal of C4.5.The mapping of the attributes values to classes is done by C4.5 by learning and then are applied to classify new and hidden instances. The induction methods begin with a root node that represents the dataset which then splits data into subsets and each attribute is tested of a node. The partitions are denoted by subtrees of the original dataset specifying test attribute values. Till all instances, in the subset fall in the same class, the process keeps running and after that, the tree stops growing and is terminated.

Decision trees are generated by this algorithm which is used to classify data instances for analysis and detection of valid results.C4.5 made a lot of improvements to its ID3 algorithm such as management of both continuous and discrete attributes, management of training data with missing attributes, management of attributes with differing costs and replacing of leaf nodes. So it is the well-suited machine learning algorithm used in network security purposes.

V. PERFORMANCE PARAMETER CALCULATIONS

The objective of this strategy is to classify traffic data whether it is suspicious, normal or unknown and to obtain outputs using the following performance metrics.

Table 3: Performance parameter calculations

'TP'(True Positive)	"The overall quantity of suspicious transactions discovered which can be certainly suspicious"
'FP' (False Positive)	"The overall quantity of normal transactions observed, which can be actually suspect"
'TN'(True Negative)	"Identified total of normal transactions which are truly normal
'FN'(False Negative)	"The total sum of suspicious transactions identified, which are actually normal."

Table 4. Performance Matrix

Recall	$\frac{TP}{TP+FN}$
Precision	$\frac{TP}{TP+FP}$
Accuracy	$\frac{TP+TN}{FP+FN+TP+TN}$
Specificity	$\frac{TN}{TN+FP}$
F measure	$\frac{2TP}{2TP+FP+FN}$

VI. RESULT AND DISCUSSION

Weka classification tool was used to classify the database which was previously generated by the Snort intrusion detection system. Different algorithms like Support Vector Machine, k-means, Decision tree Random Forest, Naïve Bayes and C4.5 were used in training and testing purposes on the dataset. To evaluate the classifiers, the confusion matrix was used and the results are tabulated in Table 5. The overall accuracy was 95.8%, 94.2%, 99.7%, 97.6%, 98.0%and 98.7% of k-means, Decision tree Random Forest, Support Vector Machine, Naïve Bayes and C4.5 respectively. Precision, recall, and specificity are equally essential due to the imbalanced data and should be taken into account. Comparing these algorithms. SVM demonstrates better results in terms of precision, recall, f-measure specificity and f measure followed closely by C4.5 and Random Forest.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table 5.Results

	K-Means	Decision Tree	SVM	Random Forest	Naïve Bayes	C4.5
Recall	1.0	0.929	0.998	0.993	0.860	0.983
Precision	0.922	0.992	0.998	0.992	0.881	0.988
Accuracy	0.958	0.942	0.997	0.976	0.980	0.987
Specificity	0.916	0.923	0.996	0.995	0.505	0.992
F measure	0.959	0.960	0.998	0.996	0.826	0.988

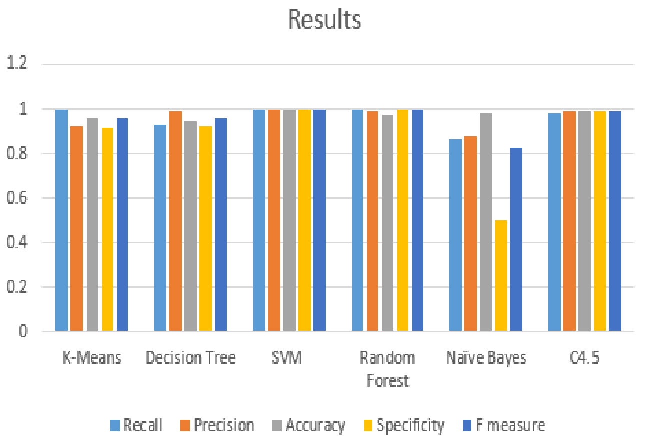

Figure. 4 Results

VII. CONCLUSION

The generated dataset has four classes with 21 features. The algorithm, which was applied to the data set, are k-means, Decision tree Random Forest, Support Vector Machine, Naïve Bayes and C4.5. The results SVM algorithm showed that the SVM algorithm has greater accuracy from k-means, Decision tree Random Forest, Naïve Bayes, and C4.5. The results shown by C4.5 are very close to that of SVM and due to their performance and accuracy, these two algorithms can be used in intrusion detection purposes. Future work will include more types of attacks and distinct methods for selecting features

REFERENCES

1. Wani, A.R., Rana, Q.P. and Pandey, N., 2017, September. Cloud security architecture based on user authentication and symmetric key cryptographic techniques. In 2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO) (pp. 529-534). IEEE.
2. Wani, A.R., Rana, Q.P. and Pandey, N., 2019. Analysis and Countermeasures for Security and Privacy Issues in Cloud Computing. In System Performance and Management Analytics (pp. 47-54). Springer, Singapore..
3. Cambiaso, E., Papaleo, G. and Aiello, M., 2012, October. Taxonomy of slow DoS attacks to web applications. In International Conference on Security in Computer Networks and Distributed Systems (pp. 195-204). Springer, Berlin, Heidelberg..
4. Somani, G., Gaur, M.S., Sanghi, D., Conti, M. and Buyya, R., 2017. DDoS attacks in cloud computing: Issues, taxonomy, and future directions. Computer Communications, 107, pp.30-48. [5] https://securelist.com/ddos-report-in-q1-2018/85373/. 2018
5. Salmen, F., Hernandes, P., Carvalho, L. and Proenca, M., 2015. Using firefly and genetic metaheuristics for anomaly detection based on network flows. In Proceedings of the 11th Advanced International Conference on Telecommunications (pp. 113-118)..
6. Liu, H.I. and Chang, K.C., 2011, October. Defending systems against tilt DDoS attacks. In 2011 6th International Conference on Telecommunication Systems, Services, and Applications (TSSA) (pp. 22-27). IEEE.
7. Eskin, E., Arnold, A., Prerau, M., Portnoy, L. and Stolfo, S., 2002. A geometric framework for unsupervised anomaly detection. In Applications of data mining in computer security (pp. 77-101). Springer, Boston, MA.
8. Dantas, Y.G., Nigam, V. and Fonseca, I.E., 2014, September. A selective defense for application layer ddos attacks. In 2014 IEEE Joint Intelligence and Security Informatics Conference (pp. 75-82). IEEE.
9. Vijayalakshmi, M., Shalinie, S.M. and Pragash, A.A., 2012, April. IP traceback system for network and application layer attacks. In 2012 International Conference on Recent Trends in Information Technology (pp. 439-444). IEEE..

AUTHORS PROFILE

Abdul Raoof Wani is currently a full-time Scholar at Amity University Noida. He has previously done in Maters in Network Technology and Management from AUUP with a CGPA of 9.15 and a Bachelors from the University of Kashmir. He is certified in CCNA and EMC Academic Associate, Information Storage and Management. His research interests include cryptography and network security, and cloud computing security. He has 4 academic papers and articles published in various reputed Conference Proceedings and Journals in the field of Information Technology.

Q.P Rana is an assistant professor at Jamia Hamdard University New Delhi India He has ten years of experience in his field of interest, cryptography, and network security.. He has done his Ph.D. from Jamia Hamdard University and is currently working as He has completed his Ph.D. from Jamia Hamdard University and currently works as the Director of the Jamia Hamdard University Computer Centre. He is the publisher and co-author of more than 18 scientific journals and conferences.

Nitin Pandey is an Assistant Professor at the University of Uttar Pradesh, Amity Institute of Information Technology. He possesses 13 years of experience. Its field of interest is theory of coding, cryptography, data communication and network security. He has done his Ph.D. in “Application of Coding Theory in Cryptography”. He is a Master of Computer Application from Maharishi Dayanand University Rohtak Haryana. He is a B.Sc. and M.Sc. in Mathematics from Deen Dayal Upadhaya University Gorakhpur Uttar Pradesh. He is the editor and co-author of more than 30 professional papers and conference publications. He is a qualified CCNA Instructor trainer and Certified in CCNA Routing and Switching, CCNA Security, IoT fundamentals, CCNA Cyber Operations.