INTRUSION DETECTION SYSTEM USING MACHINE LEARNING TECHNIQUES IN CLOUD COMPUTING

A Synopsis submitted to Gujarat Technological University

For the Award of

Doctor of Philosophy

in

Computer Engineering

By

Pinal J. Patel
[129990907010]

Under supervision of
Dr. J. S. Shah

GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD

[October - 2020]

Table of Content

I. Title of the thesis and abstract

Title of Thesis

Intrusion Detection System using Machine Learning Techniques in Cloud Computing

ABSTRACT

Over the last decades, computer and internet are being used everywhere. The People's dependencies on internet are increasing day by day. People need services more than the mass data from internet. Cloud computing has developed as a new distributed computing model for providing services on demand with "pay as you use" basis. It aims to provide convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services). The security is a primary concern in cloud computing due to its distributed and open architecture. An intrusion detection system (IDS) plays an important role to protect or secure a computer system and its main goal is to monitor activities automatically and identify unusual access or attacks to the system. Detection of anomaly in data is a fundamental task of machine learning method. Machine learning techniques are capable to identify known as well as unknown attacks.

We proposed IDS framework based on unsupervised and supervised machine learning methods on cloud platform. The unsupervised machine learning method is used to make clusters while the supervised machine learning is used for classifying attack types. Proposed IDS give flexibility to the user on cloud for IDS deployment. It can handle problem of single point of failure. We have conducted three experiments. In the first experiment, same labeled clusters of different cloud users are merged, while in the second experiment, all clusters of same cloud user are combined before applying supervised learning method. In the third experiment, we have simulated attack from on vm to another vm of cloud and executed proposed framework on it. The experimental result shows that the proposed model improves the ability of intrusion detection.

II. Brief description of the state of the art of the research topic

Cloud computing mainly focuses on providing more convenient, on-demand, network access to shared resources of configurable computing resources (e.g. networks, servers, storage, applications, and services), which can be quickly provisioned and sent with minimum effort or service provider interactions [36][109]. Cloud delivers different

types of services. 1) Software as a Service (SaaS) e.g. Microsoft Office 365, Google Apps [37] 2) Platform as a Service (PaaS) e.g. AWS Elastic Beanstalk, Google App Engine, Apache Stratos, Microsoft's Azure [38] and 3) Infrastructure as Service (IaaS) e.g. Amazon EC2, Google Compute Engine. Cloud can be deployed as a private, public, hybrid, or community cloud [43].

Security and privacy of Cloud services are major problems to be addressed as they are provided by cloud service provider over the internet. International Data Corporation (IDC) carried out a survey on the challenges of cloud and concluded that security is the biggest challenge of Cloud computing [39]. Lockheed Martin Cyber Security division shows that the major security concern after data security is intrusion detection and its prevention in Cloud infrastructures [40]. Cloud infrastructure makes use of virtualization techniques, integrated technologies, and runs through standard Internet Protocols. These may draw attention of intruders to penetrate it. Cloud computing also experiences different types traditional attacks such as Denial of Service (DoS), Distributed Denial of Service (DDoS), IP spoofing, ARP spoofing, DNS poisoning, flooding, etc. E.g. DoS attack on the underlying Amazon Cloud infrastructure caused BitBucket.org, a site hosted on AWS to stay inaccessible for few hours [41]. Cryptographic methods are not suitable to be used for cloud because to its computing cost [42]. A firewall can be considered as a good option in order to prevent outside attacks but does not work for insider attacks [39]. Efficient intrusion detection systems (IDS) should be incorporated into cloud infrastructure to detect these attacks [48]. An intrusion detection system (IDS) monitors network traffic and also monitors any kind of suspicious activity and alerts the system or network administrator. Snort can be used as IDS but it uses a rule dataset and is not able to identify unknown attacks. Machine learning methods are used to detect known and unknown attacks. Back propagation neural network has a high detection rate and can also classify unstructured packet efficiently [43]. Snort can be used as IDS but it uses rule dataset and cannot identify unknown attacks. Machine learning methods are used to detect known and unknown attacks. Back propagation neural network has high detection rate and can also classify unstructured packet efficiently [59].

III. Definition of the problem

Design and implement intrusion detection system using supervised and unsupervised machine learning methods on cloud computing.

IV. Objective and Scope of work

1. To study security issues related to cloud computing.
2. To study machine learning techniques for intrusion detection.
3. To find suitable machine learning method to increase accuracy, to reduce false alarm and to reduce training time for intrusion detection.
4. To design and implement intrusion detection system framework using machine learning techniques on cloud platform.
5. To measure performance of implemented intrusion detection system.

V. Original Contribution by Thesis

As per the literature survey and experiments which are being carried out on intrusion detection system (IDS), the supervised machine learning method - Back Propagation Neural Network (BPNN) has high detection accuracy and it works for large dataset. It is also suitable for real rime traffic classification. But the peak in the response time of BPNN is a major issue. Due to high response time, the packet dropping rate will be increased and the false alarm rate of intrusion detection system will also be increased. This issue should be addressed before applying it to the cloud platform. To mitigate the problem of BPNN, we used the K-means clustering method. The K-means clustering method is first used to partition the training instances into k clusters using euclidean distance similarity. We have used the K-means clustering method to have disjointed smaller sub-spaces.

Initially, we have divided training instances into two clusters, labeled with cluster attack and normal and dimensionally reduced dataset in form of clusters are assigned to the virtual machine of the cloud platform and finally we used BPNN classification method to classify attack types. We optimized the learning parameters of the Back Propagation Neural Network (BPNN). The model BPNN consists of architecture and parameters. For a given architecture, the values of the parameters determine how accurately the model performs the task. The goal is to minimize the loss and thereby to find parameter values that match predictions with reality. This is the essence of training. Most of the researchers have presented IDS framework for cloud computing handled by either cloud provider or cloud user and IDS is generally placed at either cloud provider site or cloud user site. We have proposed an IDS framework wherein, the IDS activities have been carried out by both cloud provider and cloud user.

VI. Methodology of Research

We used a quantitative and exploratory approach for this research work. During the literature review, we referred to various research papers, patents, thesis, annual reports of market-leading companies like Cisco [44], and Symantec [26]. In addition to this, we installed snort [14], and Bro which are an open-source IDS and observed result of Snort IDS and Bro IDS. During this initial phase of the literature review, we found researchers had done work on snort IDS and Bro IDS but they are used for a known attack. Also we found research works on IDS using machine learning methods for detecting unknown attacks [43].

Therefore, our second phase of the literature review was mainly focused on IDS using machine learning techniques. We have implemented various machine learning methods on the intrusion dataset - KDD Cup 1999 to find suitable machine learning (ML) method as ML results highly depend on the dataset. After collecting results, we have decided to use the back propagation neural network as it gives better performance in detecting attacks. We have optimized the parameters of the back propagation neural network (BPNN) after doing various experiments.

BPNN takes much training time and testing time, and we need to run it on the cloud platform, so we have decided to make clusters of a dataset and assigned clusters on the virtual machine (VM) of the cloud. Finally we have executed BPNN to collect the results of intrusion detection using two different scenarios.

In the main stage, model uses KDD CUP 1999 dataset [26] [43] which is openly accessible Model uses disarray grid as the presentation measures. Table I indicates different kinds of assaults from KDD'99 dataset

Table I: Attack Types [26]

1	Denial Of Service Attacks	back, land, neptune, pod,smurf, teardrop
2	User to Root Attacks	buffer overflow,loadmodule, perl, rootkit
3	Remote to Local Attacks	ftp write, guess passwd,imap, multihop, phf, spy
4	Probes	satan, ipsweep,nmap, portsweep

We started with different types of machine learning method to train KDD’99 dataset. Confusion matrix is useful for measuring the performance of intrusion detection classification problem. Table II shows confusion matrix. Table III represents results gathered from this experiment. From the experiments, Neural Network has good detection rate and suitable for real time traffic classification. We started with different

types of neural network. Confusion matrix is useful for measuring the performance of intrusion detection classification problem. Table IV represents results gathered from this experiment. Table V shows result gathered for considering different no. of neurons in BPNN architecture.

Table II. Confusion Matrix [41]

Actual	**Predicated**	
	Predicated	**Normal**
Attack	TP (True Positive)	FN(False Negative)
Normal	FP (false Positive)	TN (True Negative)

Table III. Machine learning Methods for intrusion detection

Sr. No	**Methods**	**Time Taken to Build Model (sec)**	**Time to test data(Sec)**	**Correctly Classified %**	**Incorrectly classified %**	**Accuracy**	**Error**
1	Decision Tree	85.99	0.23	99.9	0.0417	0.999583	0.0006
2	Naive Bayes	3.76	3.42	98.1	1.873	0.98127	0.00187
3	Neural Network	4368.02	2.59	99.9	0.072	0.99928	0.0009
4	SVM	2461.34	1.3	99.7	0.2929	0.997071	0.0029
5	KNN	0.16	185000.3	99.9	0.0667	0.999333	0.0007

Table IV. Performance of Neural Network Types

Type of Neural Network	**MSE**	**FNR**	**FPR**	**TPR**	**TNR**
Feed forward	0.0210	0.0024	0.010	0.9900	0.9976
Back Propagation	0.0123	0.0022	0.016	0.9984	0.9978
Recurrent	0.00604	0.0447	0.005	0.9948	0.9553
Radical Basis function	0.3015	0.8480	0.00	0.1520	1.00
Probabilistic	0.2702	0.5000	1.00	0.00	0.5000
Generalize	0.6650	0.7836	0.00	0.2164	1.00
Linear Vector Quantization	0.218	0.1551	0	0	0.8449

The neuron is the fundamental data preparing unit of Neural Network. It comprises of :

1. A lot of connections, depicting the neuron contributions, with loads W1, W2, … , Wm.
2. An adder function (direct combiner) is utilized for processing the weighted entirety of the

data sources (genuine numbers).

3. Initiation capacities are utilized for constraining the sufficiency of the neuron yield.

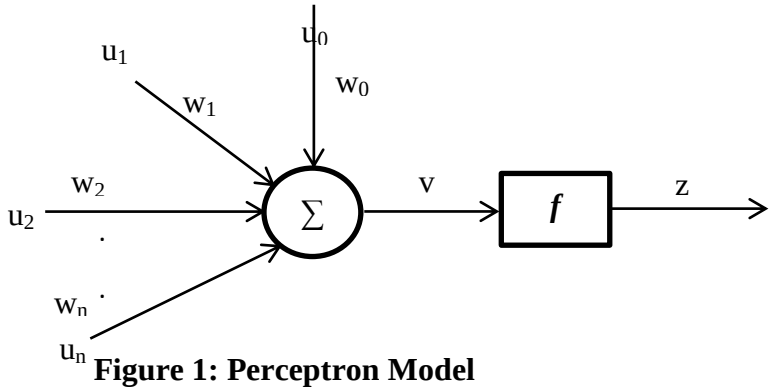


Figure 1: Perceptron Model

TPR = TP / (TP+FN)

TNR = TN / (FP+TN)

FPR = FP / (FP+TN)

FNR = FN / (FN+TP)

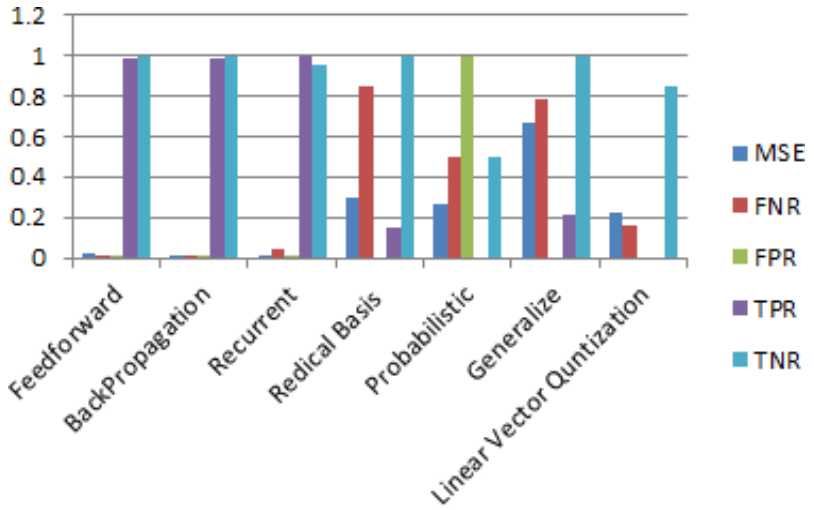


Figure 2: Comparison of Neural Network Types

Table V. Single layer neural network

No. of Neuron	MSE	Time	FNR	FPR	TPR	TNR
10	0.0123	5.03	0.0022	0.0168	0.9832	0.9978
20	0.0213	5.06	0.0006	0.1094	0.8906	0.9994
30	0.0345	5.57	0.0007	0.2954	0.7046	0.9993
40	0.3480	18.52	0.0170	0.0417	0.9583	0.9830

In the next examination, the system structure of BPNN is set to 10-10- 1, i.e. 10 neurons in input layer, 10 neurons in hidden layer and one neuron in output layer. Table VI speaks to results accumulated from this investigation.

Table VI. Performance of network with one hidden layer

Network	MSE	Time	FNR	FPR	TPR	TNR
10-10-1	0.0456	0:28:33	0.0316	0.0523	0.9477	0.9684
20-10-1	0.0274	0:35:16	0.0130	0.0308	0.9692	0.9870
30-10-1	0.0388	0:42:58	0.0055	0.0409	0.9591	0.9945
41-10-1	0.2284	1:13:59	0.0204	0.4724	0.5276	0.9796
41-20-1	0.2102	1:21:51	0.0214	0.4738	0.5262	0.9786
41-30-1	0.0664	1:28.33	0.0069	0.2058	0.7942	0.9931

As we can see network with 20 input nodes and 10 hidden nodes performed better than others. There was significant change in performance when we increased number of hidden nodes from 20 to 30 in network with 41 input nodes. In next experiment, we added another layer of hidden layer. Along these lines, the system structure of back proliferation neural system is set to 10-10-10-1. Table VII represents results gathered from this experiment.

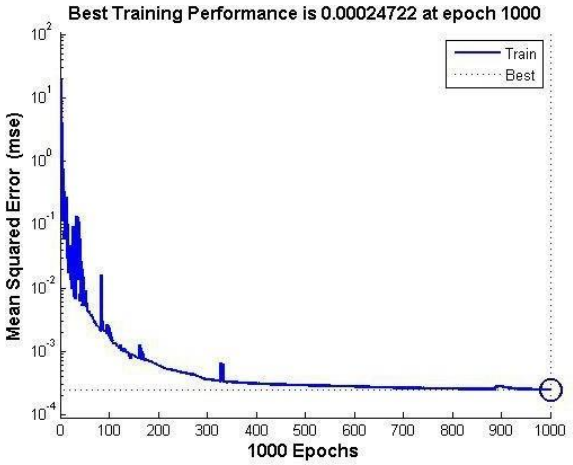


Figure 3: Performance of BPNN over 1000 epochs

TABLE VII. Performance of network with two hidden layers

Network	MSE	Time	FNR	FPR	TPR	TNR
10_10_10_1	0.0228	1:13:25	0.0108	0.0135	0.9865	0.9892
20_10_10_1	0.0669	1:20:21	0.0106	0.4652	0.5348	0.9894
30_10_10_1	0.1048	1:25:13	0.0134	0.0206	0.9794	0.9866
40_10_10_1	0.0366	1:30:23	0.0200	0.2247	0.7753	0.9800
41_10_10_1	0.0325	1:31:21	0.0086	0.0228	0.9772	0.9914
41_20_20_1	0.0316	1:32:33	0.0123	0.0384	0.9616	0.9877
41_20_30_1	0.1048	1:31:15	0.0107	0.0472	0.9528	0.9893

As you can see, network with 10 inputs performed far better than others. On the other hand, network with 30-10-10-1 and 41-20-30-1 performed worst.

Weight Initialization of BPNN:

a. Initialize all loads to a similar worth (for example zero or one)

For this situation, each concealed unit will get the very same sign. In the event that all loads are instated to 1, every unit gets sign equivalent to entirety of information sources. On the off chance that all loads are zeros, each shrouded unit will get zero sign. Regardless of what was the info - if all loads are the equivalent, all units in concealed layer will be the equivalent as well.

b. Initialize random weight

It makes neural network learn faster but every time values of weights will be changed, for same dataset, different run gives different output.

c. Good initial weight values:

Choose initial weights of a neural network from the range (-1 /sqrt(d) , 1/sqrt(d)) Where d= no. of inputs to the given neuron of layer.

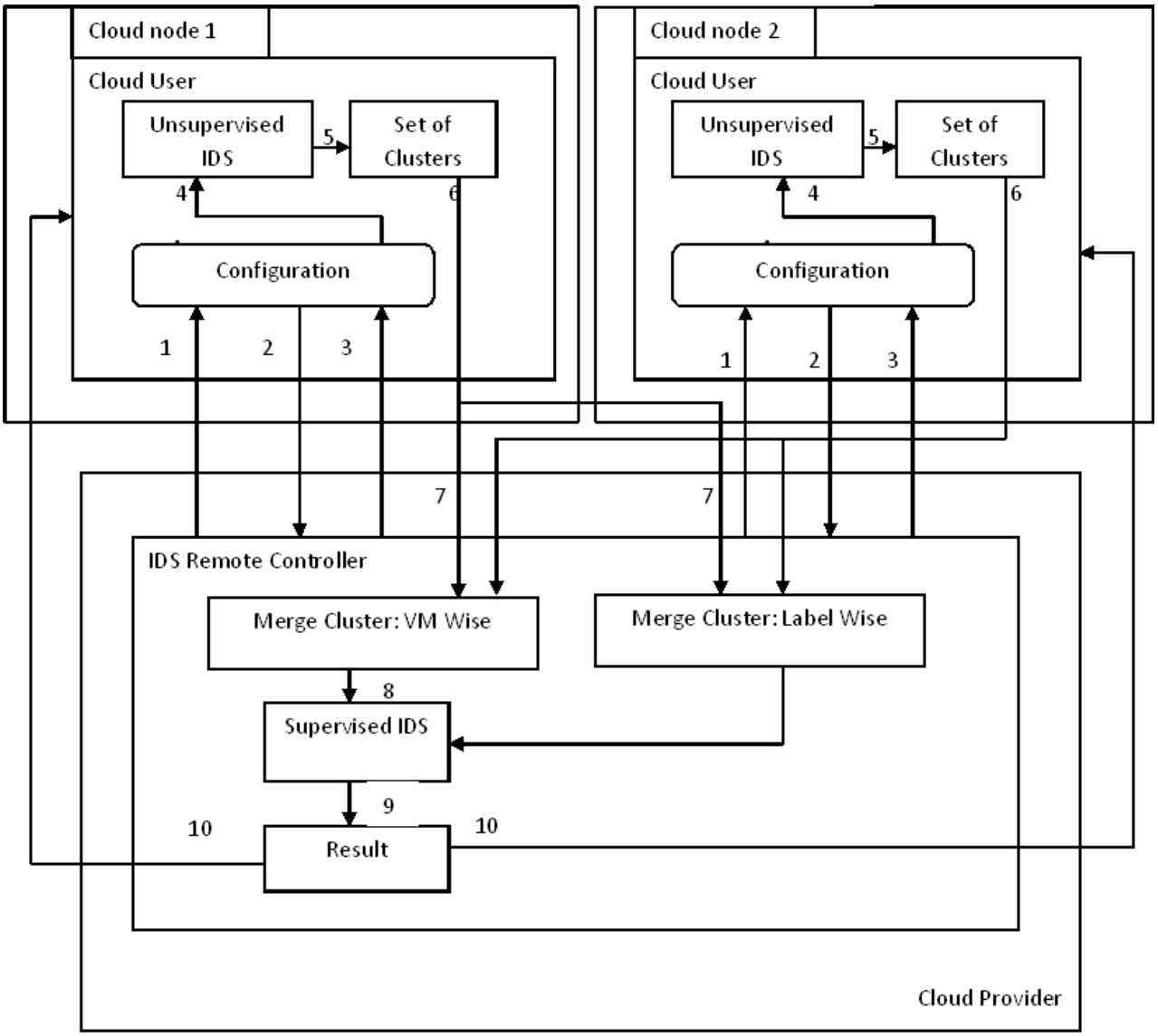
Table VIII. Performance of BPNN with different weight assignment methods

Weight initialize method	Epoch	Time to learn	Mean Square Error (MSE)	TPR	TNR	FPR	FNR
With all 0	1000	00:04:48	0.0694	0.9967	0.9237	0.0033	0.0763
With all 1	1000	00:04:49	0.0693	0.9966	0.9237	0.0033	0.0764
With 0.5	1000	00:03:10	0.0433	0.9965	0.9239	0.0035	0.0761
Random	1000	00:05:06	0.0433	0.9477	0.9684	0.0523	0.0316
Range (-1/sqrt(d), 1/sqrt(d))	1000	00:04:58	0.0455	0.9478	0.9683	0.0524	0.0315

Cloud Setup

Open nebula is used to setup real cloud. Open nebula server and sunstone server are installed on server machine. Three physical nodes are added to cloud server and VMs are dynamically created on nodes using open nebula server.

Proposed System



1: Initialize IDS

2: Response with Configuration

3: Launch IDS

4: Perform Unsupervised IDS

5: Generate Clusters

6: send Cluster to IDS Remote Controller

7: 1) Merger Cluster data VM wise 2) Merger Cluster Data Label Wise

8: Perform Supervised IDS

9: Evaluate

10: Send Alert to Cloud User VM

Fig. Proposed IDS Based on Machine Learning On Cloud Computing

Figure 4: Proposed IDS framework

1) IDS remote controller resides at Cloud Provider. It initiates IDS module service and sends IDS require message to all active cloud user VMs and wait for timeout period to receive positive or negative response from cloud user VM.
2) Active cloud user VMs receive IDS require message and check their own capacity of physical resources as IDS requires physical resources to run. If Cloud User VM requires IDS to run on their machine and physical requirement is also satisfied then he sends positive reply to IDS Remote Controller. If Cloud User VM is not interested to run IDS or lack of physical resources, then he gives negative response to IDS Remote Controller.
3) Based on Positive responses from Cloud User VMs, IDS remote controller stores their details and maintain counter of cloud users of IDS and launch IDS instance to them and wait for timeout period to receive clusters. If None of the Cloud User VM responds with clusters then IDS remote Controller uses unsupervised machine learning method and generate k set of clusters.
4) Cloud User VM receives IDS instance and use unsupervised machine learning method to predict intrusion activity and sends k set of cluster data to IDS remote controller. If Cloud User VM does not sends cluster data within timeout period then IDS Remote controller delete his information and decrease counter.
5) After receiving k set of clusters, IDS Remote Controller merges them based on Cloud user VM wise and labeled cluster wise. i.e. if Cloud User Vm1 and Cloud user VM2 both send 2 clusters – cluster 0 and 1 to IDS Remote Controller then IDS Remote Controller merge data of 1) Cluster 0 and Cluster 1 of VM1, and Cluster 0 and Cluster 1 of VM2 (Merger Cluster - VM wise) , 2) merger data of Cluster 0 of Cloud User VM1 and data of Cluster 0 of Cloud User VM2, and data of Cluster 1 of Cloud User VM1 and data of Cluster 1 of Cloud User VM2.
6) IDS Remote Controller performs supervised machine learning to classify data as intrusion data or normal data.
7) Based on results, IDS Remote Controller sends alert to all Cloud User VM of IDS module.

Unsupervised IDS

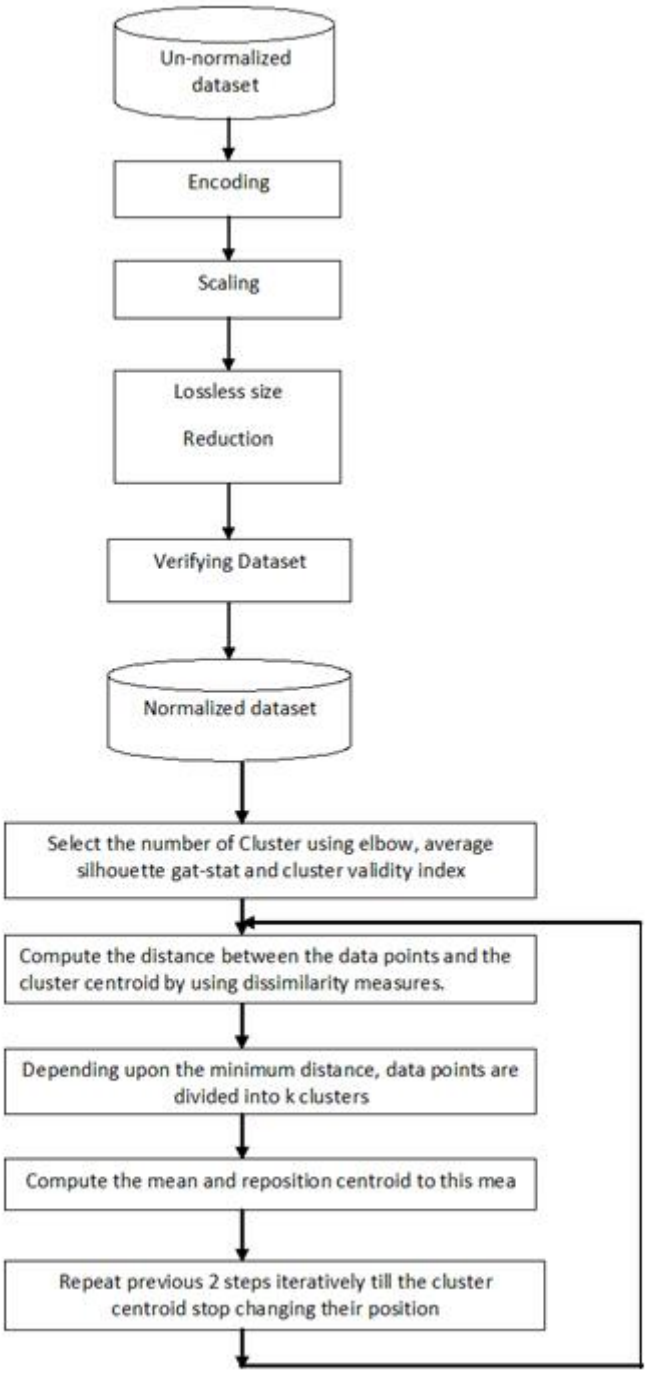


Figure 5: Unsupervised IDS

Cloud User VM uses Unsupervised IDS for generating clusters. We have used modified K – means algorithm to generate cluster.

Kmeans Clustering Algorithm

Algorithm :Kmeans of Cluster k

Input : Dataset of size M x N, and No. of Cluster k

Output: k clusters of n objects

Step 1: Initialize Cluster centroid randomly.

Step 2: Calculate the distance between the data points and the cluster centroid by using dissimilarity measures. Depending upon the minimum distance, data points are partitioned into k clusters.

Step 3: Compute the new centroid for each and every cluster.

Step 4: Go to Step 2 and continue this procedure until cluster label does not change anymore.

Advantages:

1) Easy to implement and robust.
2) Relatively adaptable and productive in preparing enormous informational indexes with direct time multifaceted nature.
3) Produce more tightly groups than various leveled bunching Produce tighter clusters than hierarchical clustering

Disadvantages:

1) Applied just when the mean of a bunch is characterized.
2) Cannot be applied on categorical attributes
3) Sensitive to the selection of number of a clusters k and initial cluster center.
4) Sensitive to noise and outlier data points.

We have gone through literature for finding no. of optimal cluster. Researchers suggest to use Elbow method, Average Silhouette index and gat stat index to find value of k cluster

So, we have applied these three methods on our intrusion dataset.

Find optimal k no. of clusters

1) Elbow Method

The main purpose of the elbow method is to run k-means clustering on the dataset for a range of values of *k* (say, *k* from 1 to 10), and for each value of *k* compute the sum of squared errors (SSE). Our goal is to select a small value of *k* that still has a low SSE.

Step 1: Calculate clustering algorithm (e.g., k-means clustering) for different values of

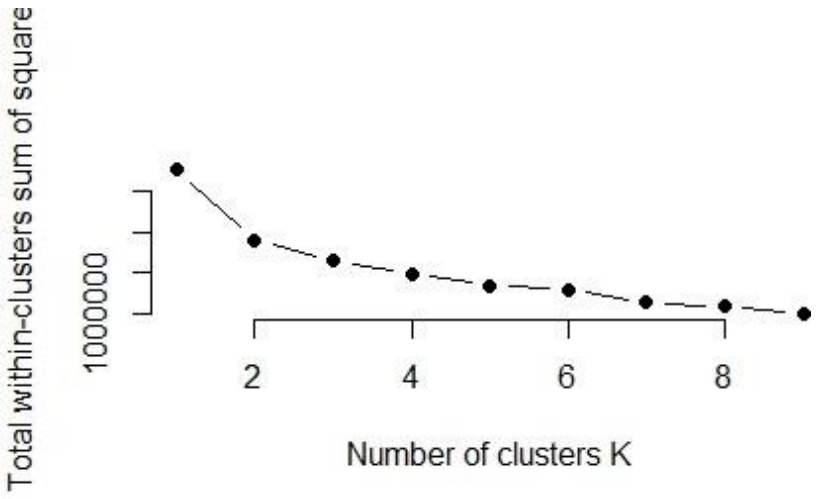
k. For instance, by varying k from 1 to 10 clusters.

Step 2: For every k, calculate the total within-cluster sum of square (wss).

Step 3: Plot the curve that is according to the number of clusters k.

Step 4: The location of a bend (knee) in the plot is generally considered as an indicator of the appropriate number of clusters.

We have applied elbow method on our dataset. Output is shown below.



2) Silhouette Method

The silhouette value is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation). The silhouette ranges from −1 to +1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.

Step 1: For each data point i, let a(i) be the average dissimilarity of i with all data points within the same cluster.

Step 2: Let b(i) be a lowest average dissimilarity of i to any other cluster.

Step 3: The cluster with this lowest average dissimilarity is said to be the neighboring cluster of i because it is the next best fit cluster for point.

3) Gap Statistic

The gap statistic compares the total within intra-cluster variation for different values of k with their expected values under null reference distribution of the data. The estimate of the optimal clusters will be value that maximizes the gap statistic (i.e, that yields the largest gap statistic). This means that the clustering structure is far away from the random uniform distribution of points.

4) Dunn Index

These cluster validity indices have been introduced in paper [7]. If a data set contains

well- separated clusters, the distances among the clusters are usually large and the diameters of the clusters are expected to be small [3]. Therefore larger value means better cluster configuration.

Implementation of Unsupervised IDS

We have gone through all steps of data normalization and run K-means clustering algorithm with k finding methods.

Sr No	Method	No. of clusters	Value_index	Value _index for k=2 to 9
1	Dunn	2	0.208	[0.2080 ,0.0600 ,0.0440 , 0.0406, 0.167 , 0.0233 ,0.0227 ,0.0285]
2	Silhouette	2	0.463	[0.4630 ,0.4564 ,0.3630 ,0.4194, 0.4341 ,0.4059,0.4461 ,0.4264]
3	Gap-stat	2	-0.304	[-0.3040, -0.9225, -1.4652, -1.6343 -1.8044, -2.3216, -2.2199,-2.0993]

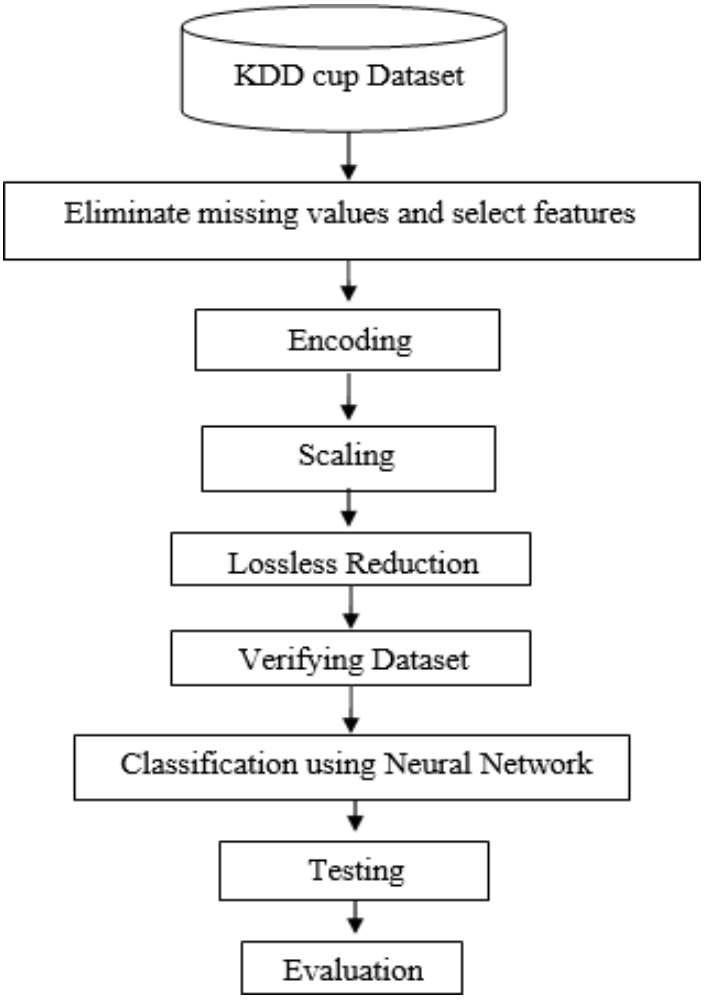
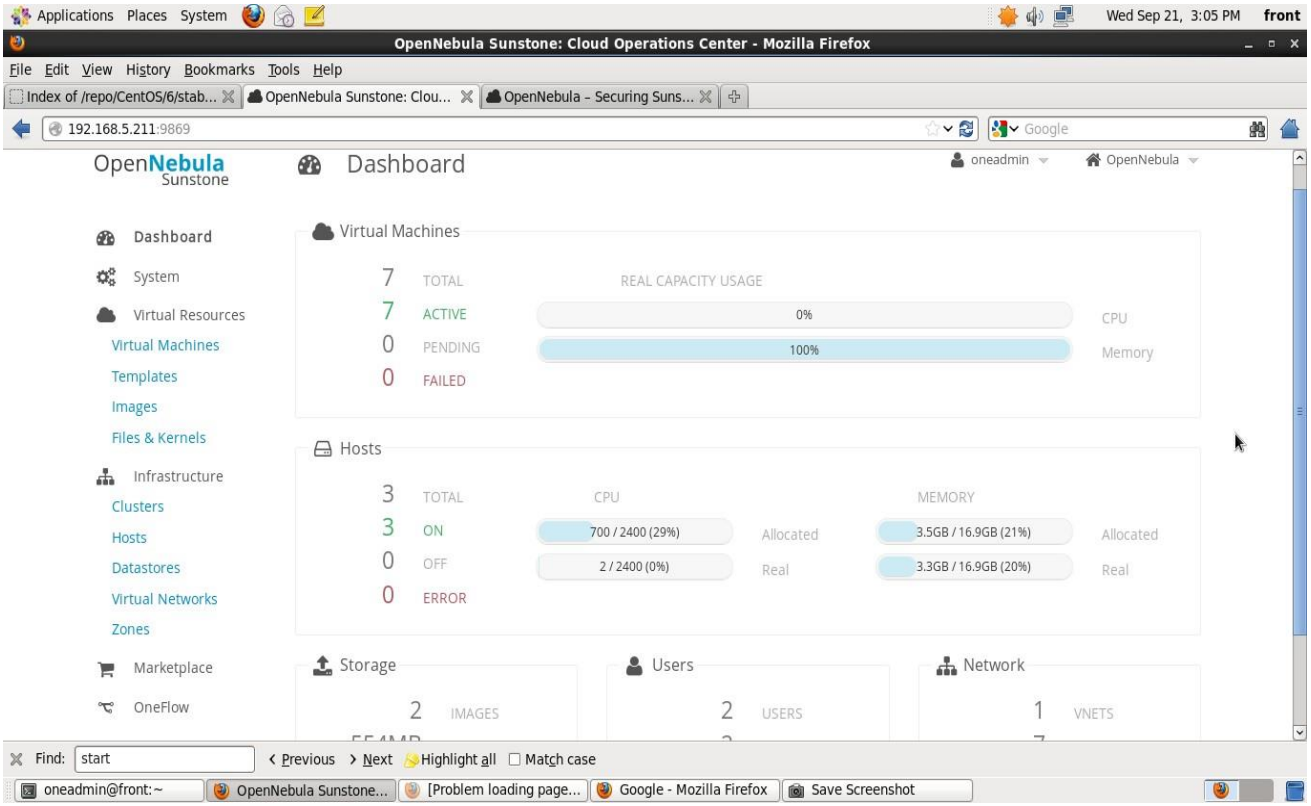


Figure 6: Supervised IDS [43]

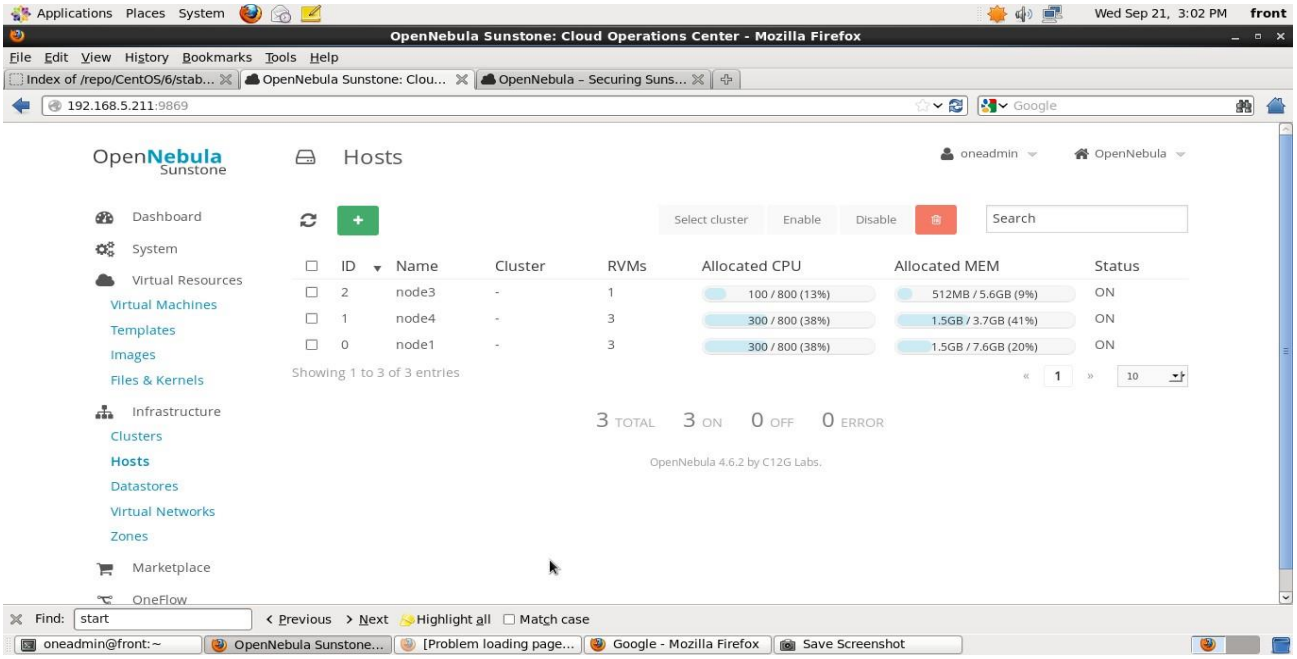
Implementation

We have setup private cloud using open nebula 4.12 on centos machines. We have added three physical nodes to open nebula server and created VMs on it. Open nebula sunstone is used for GUI. Screenshot of Private Cloud is shown below.

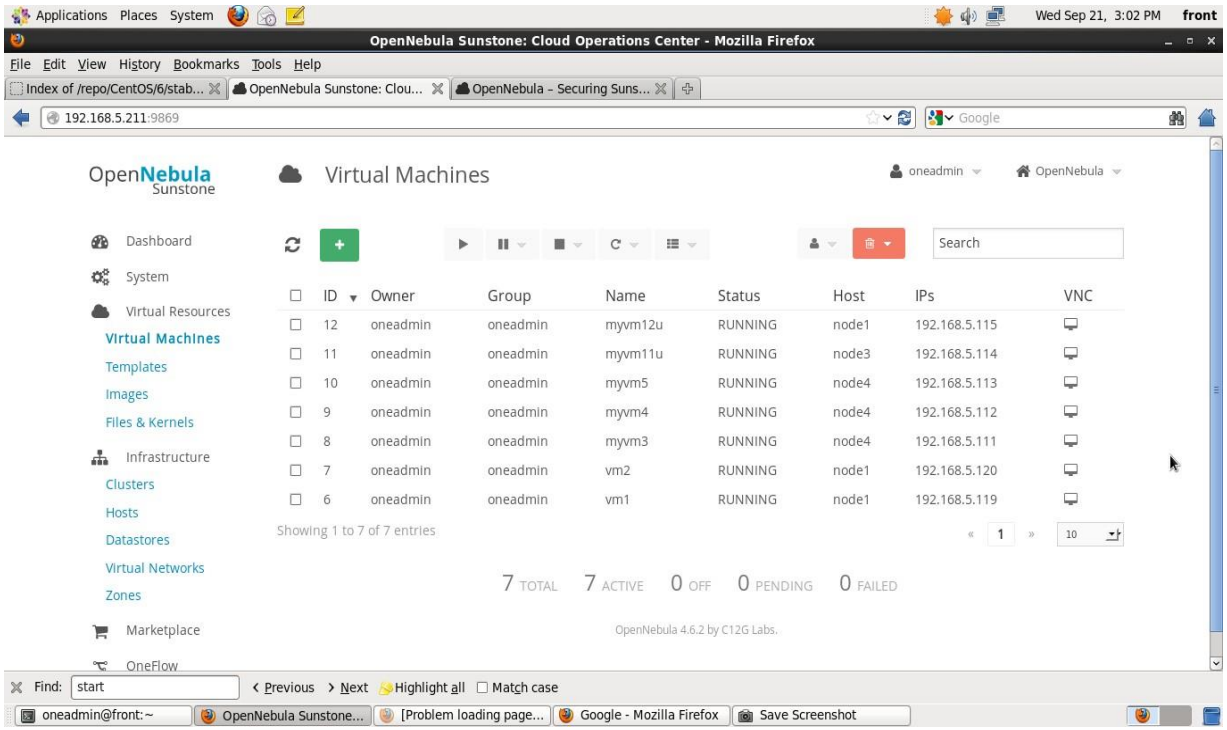
1) Dashboard of open nebula sunstone



2) Nodes in Cloud



3) VMs in Cloud

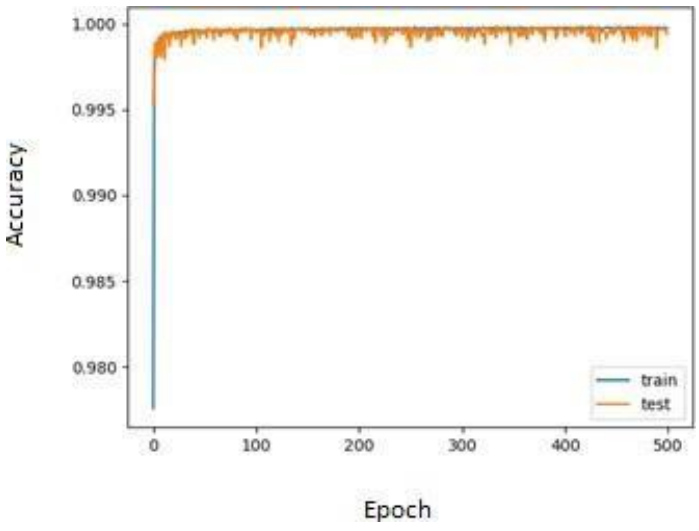


We have used Java API for open nebula and java JSchto and Pyone to extract information of active VM and deploy unsupervised IDS on it

VII. Results

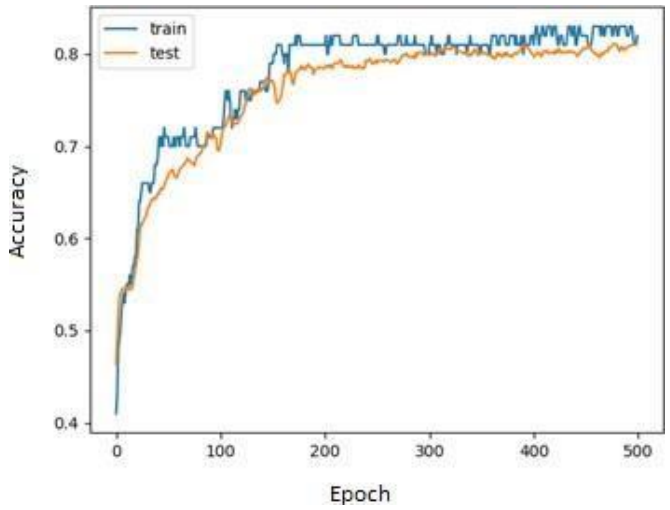
Experiment 1)

In First Experiment, labeled clusters of each Cloud User VM has been merged, and supervised IDS are run on it. Figure shows training accuracy v/s testing accuracy over iterations



Experiment 2)

In Second Experiment, a same cluster of each Cloud User VM has been merged and supervised IDS are run on it. Figure shows training accuracy v/s testing accuracy over iterations



VIII. Achievements with respect to objectives

Using supervised and unsupervised machine learning methods on cloud for intrusion detection gives better performance in terms of accuracy and reduces the false alarms.

IX. Conclusion

The major contributions of this research work are the proposal of a hybrid framework of unsupervised and supervised machine learning methods for effective intrusion detection on the cloud platform, the detection techniques for a network and/or host intrusion detection system that use clustering and classification methods to enhance the performance of IDS. The performance of the proposed IDS framework has been evaluated in terms of detection rate, precision, F1 score, recall, and false-positive rate. The KDD CUP 1999 dataset has been used to test the proposed IDS framework.

This model combines supervised and unsupervised learning methods for intrusion detection. Thus it reduces false alarm generated by a single machine learning method. Two stages sequentially deployment of clustering and classification of the IDS model is time and resource-conserving. The model is implemented in two phases, in the first phase the model uses a clustering method for reducing the size of the data and response time while in the second phase. The model is implemented for bifurcating the attack types and detecting the multiclass attack. Making clusters from a dataset reduces the amount of training time for the supervised learning method. Cloud providers and cloud

users participate in proposed IDS so, it reduces the risk of a single point of failure. In case of failure of VM, IDS remote controller can detect intrusive event. We verified the applicability of the proposed IDS framework using the KDD CUP 1999 dataset. We have implemented three different scenarios to detect intrusion on cloud platforms. In which the second scenario detects low frequent attacks more accurately than the first scenario, while the first scenario detects DoS and Probe attacks more accurately than the second scenario. Both scenarios detect normal data above 97%.

X. Paper publications and a list of all publications

Sr No	Title of Paper	Conference/ Journal name
1	Intrusion Detection System using Back propagation Neural Network	1st International conference on smart computing and informatics (SCI 2017), 3-4th March, ANITS- Vishakhapatnam
2	Performance Analysis of Neural Networks for Intrusion Detection System	International Journal of Computer Technology & Applications, vol 8(2),88-93, ISSN:2229-6093
3	Comprehensive study on Machine Learning Techniques for IDS in Cloud Computing	International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, vol. 3 Issue 4, April – 2014

XI. Patents: NIL

XII. References

1. Prezi (2016) History of the internet [Online]. Available https://prezi.com/q3k1voj9l5xz/history-of-the internet/ [Accessed in December 2016]

2. Mayo K. and Newcomb P (2008) The birth of the world wide web: An oral history of the Internet [Online]. Available http://www.vanityfair.com/news/2008/07/internet200807 [Accessed in December 2011]

3. Prezi (2016) The Internet [Online]. Available https://prezi.com/nt0kdsbj7ijg/the-internet/ [Accessed in December 2016]

4. Palermo E. (2015) 10 worst data breaches of all time [Online]. Available http://www.tomsguide.com/us/ biggest-data-breachesnews-19083.html [Accessed in May 2015]

5. Honan B. (2015) DDoS attacks take down RBS, Ulster bank, and NatWest online systems [Online]. Available http://www.csoonline.com/article/2955693/cyber-attacks-espionage/ ddosattacks-take-down-rbs- ulster-bank-and-natwest-online-systems.html [Accessed in May 2015]

6. Pegram, B. (2016) 10 surprising Cyber security facts that may affect your online safety Heimdal security Blog [Online]. Available https://heimdalsecurity.com/blog/10-surprisingcyber-security-facts-that-may- affect-your-online-safety/ [Accessed in August 2016]

7. Statista (2018) [Online]. Available:https://www.statista.com/statistics/241272/cyber-crime-forensics-types/ [Accessed in September 2018]

8. K. Ilgun (1993) „USTAT: A real-time intrusion detection system for UNIX, *Proceedings of IEEE Symposium on Security and Privacy* pp 16–28

9. Sreenivas Sremath Tirumala, Hira Sathu, Abdolhossein Sarrafzadeh,(2015) Free and open source intrusion detection systems: A study, *International Conference on Machine Learning and Cybernetics (ICMLC)* Guangzhou China

10. J. Peng, C. Feng, and J. Rozenblit (2006) A hybrid intrusion detection and visualization system, *Proceedings of 13th Annual IEEE International Symposium and Workshop on Engineering of Computer Based Systems* pp 505–506

11. O. Depren, M. Topallar, E. Anarim, and M.K. Ciliz (2005) An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks, Expert systems with Applications 29 (4) pp 713–722

12. M. Sebring, E. Shellhouse, M. Hanna, and R. Whitehurst (1988) Expert systems in intrusion detection: A case study, *Proceedings of 11th National Computer Security Conference*, pp 74–81

13. K. Ilgun, R.A. Kemmerer, and P.A. Porras (1995) State transition analysis: A rule-based intrusion detection approach, IEEE transactions on software engineering pp 181–199

14. Sandeep Kumar , Eugene Spafford (1994) An application of pattern matching in intrusion detection, Technical Report 94-013, Purdue University, Department of Computer Sciences

15. Seo, Jeongseok, and Sungdeok Cha (2007) Masquerade detection based on SVM and sequence-based user commands profile, *Proceedings of 2nd ACM symposium on Information, computer and communications security*

16. S. Coull,Joel BranchCoull,B. Szymanski,Eric Breimer (2003) Intrusion detection: A bioinformatics approach, 19th annual *IEEE Proceeding on Computer Security Applications Conference NW Washington DC*

17. Sho Ohtahara,Takayuki Kamiyama (2009) Anomaly-based Intrusion Detection System Sharing Normal Behavior Databases among Different Machines, *9th IEEE International Conference on Computer and Information Technology* Xiamen China Volume 1

18. Karim Ali,Raouf Boutaba (2009) Applying Kernel Methods to Anomaly Based Intrusion Detection Systems, IEEE Global Information Infrastructure Symposium Hammemet Tunisia

19. DamianoBolzoni, SandroEtalle, Pieter Hartel (2006) POSEIDON: a 2-tier Anomaly-based Network Intrusion Detection System", *Proceedings of the Fourth IEEE International Workshop on Information Assurance* pp144-156

20. D. Anderson, T.F. Lunt, H. Javitz, A. Tamaru, and A. Valdes (1995) Detecting unusual program behavior using the statistical component of Next-generation Intrusion Detection Expert System (NIDES) SRI International Computer Science Laboratory

21. Christopher Kruegel, Giovanni Vigna (2003) Anomaly detection of web-based attacks, *Proceedings of 10th ACM conference on Computer and communication security (Washington D.C., USA), ACM Press* pp 251–261

22. HS Teng, K. Chen, SC Lu (1990) Adaptive real-time anomaly detection using inductively generated sequential patterns, *Proceedings of Symposium on Research in Security and Privacy (Oakland, CA)* pp 278–284

23. Wang, Ke, Gabriela Cretu, Salvatore J. Stolfo (2006) Anomalous payload-based worm detection and signature generation, Recent Advances in Intrusion Detection Springer Berlin Heidelberg

24. Krugel, Christopher, Thomas Toth (2002) Flexible, mobile agent based intrusion detection for dynamic networks European Wireless

25. S. Chakrabarti (2010) Study of snort-based IDS, Proceedings of the ICWET '10 *International Conference & Workshop on Emerging Trends in Technology* Mumbai Maharashtra India

26. Vern Paxson (1999) Bro: a system for detecting network intruders in real-time, The International Journal of Computer and Telecommunications Networking Volume I pp 23-24

27. Huiming Yu, N. Powell, D. Stembridge, X. Yuan (2012) Cloud computing and security challenges, *Proceeding of the Annual Southeast Conference* pp 298-302

28. Andrew Hay, Daniel Cid (2008) Chapter Getting Started with OSSEC, book OSSEC Host-Based Intrusion Detection Guide pp 1-27

29. Mazzariello, C., Bifulco, R. Canonico, R. (2010) Integrating a network IDS into an open source cloud computing environment, *6th International Conference on Information Assurance and Security* pp 265-270

30. Lo, C.C., Huang, C.C. , Ku, J.(2010) A cooperative intrusion detection system framework for cloud computing networks, *39th International Conference on Parallel Processing Workshops* pp.280-284

31. Khan, Nabeel, Younus, Bilal Rauf, and Kabeer Ahmed (2010) Comparative study of intrusion detection system and its Recovery mechanism, *The 2nd International Conference on Computer and Automation Engineering (ICCAE)* Volume 5

32. E. H. Spafford (2000) Intrusion detection using autonomous agent, Journal of Computer Networks 3(4) pp 547-570

33. Zhang Guo-rong, Lu Song-nian (2006) Research of distributed intrusion detection system based on mobile agent, Computer engineering and design Volume 27 pp. 3328-3331

34. Paul Innella (2014) An Introduction to IDS [Online]. Available http://www.symantec.com/connect/articles/ introduction-ids [Accessed in December 2014]

35. Mark Crosbie and Gene Spafford (1995) Active defense of a computer system using autonomous agents, Technical Report 95-008, COAST Group Department of Computer Sciences Purdue University West Lafayette IN 47907-1398

36. Wayne Jansen, Wayne Jansen, Timothy Grance, Rebecca M. Blank ((2011) NIST- Guidelines on Security and Privacy [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary [Accessed in September 2012]

37. Oktaya, U. And Sahingoz (2013) „Attacks types and intrusion detection systems in cloud computing",
Proceedings of 6th International Information Security & Cryptography Conference (ISC) pp 71-76

38. Zarrabi, A. and Zarrabi, A. (2012) Internet intrusion detection system service in a cloud, International Journal of Computer Science Issues Volume 9 pp 694-814

39. Peeyush Mathur, Nikhil Nishchal (2010) Cloud Computing: New challenge to the entire computer industry, *1st International Conference on Parallel, Distributed and Grid Computing* pp 223-228

40. Loubna Dali,Ahmed Bentajer,Elmoutaoukkil Abdelmajid, Karim Abouelmehdi, Hoda Elsayed, Eladnani Fatiha, Benihssane Abderahima (2015) A survey of intrusion detection system ,IEEE 2nd World Symposium on Web Applications and Networking (WSWAN)

41. Ch. Cachin and M. Schunter (2011) A Cloud You Can Trust, IEEE Spectrum 48(12) pp 28-51

42. Jun-jie, W. And Sen (2011) Security issues and countermeasures in cloud computing, *IEEE International Conference on Grey Systems and Intelligent Services (GSIS)* Nanjing pp 843-846

43. Bhavin Shah (2015) improving anomaly detection process in computer networks having existing ids using additional behavioral layer, optimized back propagation neural network and mobile agents (multi class attack detection) Ph.D. Thesis GTU

44. Li, C., Song, Q., & Zhang (2004) MA-IDS: Architecture for distributed intrusion detection using mobile agents", *Proceedings of 2nd International Conference on Information Technology for Application (ICITA)*

45. Dastjerdi, A.V., Bakar, K.A. Tabatabaei, S.G.H (2009) Distributed intrusion detection in clouds using mobile agents, *Third International Conference on Advanced Engineering Computing and Applications in Sciences* pp 175-180

46. Chen Y, Sion R, (2010) On securing untrusted clouds with cryptography, *Proceedings of the*

9th annual ACM workshop on Privacy in the electronic society pp 109–114

47. TakakoPatrícia, Patricia Takako Endo, GlaucoEstacioGonçalves, DjamelSadok (2013) A Survey on Open-source Cloud Computing Solutions 8th Workshop on Clouds and Grids

48. Chirag Modi, Dhiren Patel, Bhavesh Borisaniya, Hiren Patel, Avi Patel, Muttukrishnan Rajarajan (2013) A survey of intrusion detection techniques in Cloud, Journal of Network and Computer Applications pp 42–57

49. Bhavin Shah, Bhushan H. Trivedi (2013) Data Set Normalization: For Anomaly Detection Using Back Propagation Neural Network, *IEEE International Conference on Research and Development Prospectus on Engineering and Technology (ICRDPET)*

50. Tesfahun, Abebe, and D. Lalitha Bhaskari (2013) Intrusion Detection Using Random Forests Classifier with SMOTE and Feature Reduction, *IEEE International Conference on Cloud & Ubiquitous Computing & Emerging Technologies (CUBE)*

51. Liu, Hongyu; Lang, Bo (2019) Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey, Applied Sciences (2076-3417) Volume 9 Issue 20 pp 43-9645.

52. D. J. Brown, B. Suckow, and T. Wang (2002) A Survey of Intrusion Detection Systems, Department of Computer Science University of California San Diego

53. H. Q. WangZ. Q. WangQ. ZhaoG. F. WangR. J. ZhengD. X. Liu (2006) Mobile Agents for Network Intrusion Resistance, Advanced Web Network Technologies and Applications Springer Berlin Heidelberg pp 965-970

54. Smaha, Stephen E (1988) Haystack: An intrusion detection system, Fourth Aerospace Computer Security. Applications Conference pp 37–44

55. Porras, Phillip A.,Alfonso Valdes (1998) Live traffic analysis of TCP/IP Gateways, Internet Society's Networks and Distributed Systems Security Symposium

56. Liu, Jianxiao, and Li Lijuan (2008) A Distributed Intrusion Detection System Based on Agents, *Proceedings of IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application* Volume I pp 553–557

57. Tao Ji, Yongzhong Li, Jing Xu (2013) Immune Mobile Agent and Its Application in IDS, *Proceedings of 2013 Chinese Intelligent Automation Conference* Electrical Engineering Volume 254 pp 127-136

58. Sodiya, Adesina Simon (2006) Multi-level and secured agent-based intrusion detection system, Journal of Computing and Information Technology -CIT 14(3) pp 217–223

59. Christopher Krügel, Thomas Toth (2001) Applying Mobile Agent Technology to Intrusion

Detection, CSE Workshop on Software Engineering and Mobility

60. Saidat Adebukola Onashoga, Adebayo D. Akinde, Adesina Simon Sodiya (2009) A Strategic Review of Existing Mobile Agent Based Intrusion Detection Systems, Issues in Informing Science and Information Technology Volume 6

61. Bhavin Shah, Bhushan Trivedi (2015) Improving Performance of Mobile Agent Based Intrusion Detection System, *Proceedings of the Fifth International Conference on Advanced Computing & Communication Technologies* pp 425–430

62. L. Yu and H. Liu (2004) Efficient Feature Selection via Analysis of Relevance and Redundancy, Journal of Machine Learning Research Volume 5 pp 1205 -1224

63. M.RAMAKRISHNA MURTY (2012), Data mining and soft computing techniques [online]. Available http://dataminingzone.weebly.com/uploads/6/5/9/4/6594749/ch_21major_clustering_methods.pdf [Accessed in October 2014]

64. ICSA Labs (2010) A book on Intrusion detection systems buyer's guide, [Online]. Available https://www.ipa.go.jp/security/fy11/report/contents/intrusion/ids-meeting/idsbg.pdf [Accessed in December 2014]

65. D. stiawan, A. H. Abdullah, M. Y. Idris (2010) The Trends of Intrusion Prevention System Network ,2nd International Conference on Education Technology and Computer (ICETC), Volume 4 pp 217- 221

66. S. Roschke, C. Feng, C. Meinel (2009) An Extensible and Virtualization Compatible IDS Management Architecture, *5th International Conference on Information Assurance and Security* Volume 2 pp 130-134

67. A.bakshi, B. Yogesh (2010) Securing Cloud from DDOS Attacks Using Intrusion Detection System in Virtual Machine, *Second International Conference on Communication Software and Networks* pp 260-264

68. C. C. Lo, C. C. Huang, J. Ku (2008) Cooperative Intrusion Detection System Framework for Cloud Computing Networks, *First IEEE International Conference on Ubi-Media Computing* pp 280-284

69. C. Mazzariello, R. Bifulco, R. Canonoco (2010) Integrating a network IDS into an Open source Cloud computing, *Sixth International conference on Information Assurance and Security (IAS)* pp 265-270

70. T. Dutkevyach, A. Piskozub, N. Tymoshyk (2007) Real-Time Intrusion Prevention and Anomaly Analyze System for Corporate Networks, 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications IDAACS pp 599-602

71. H. Zhengbing, S. Jun, V. P. Shirochin (2007) An Intelligent Lightweight Intrusion Detection System

with Forensic Technique, 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications IDAACS pp 647-651

72. T. Garfinkel, M. Rosenblum (2003) A Virtual Machine Introspection Based Architecture for Intrusion Detection, *Proceeding of Network and Distributed Systems Security Symposium* pp 191-206

73. Vieira, A. Schulter (2010) Intrusion detection techniques in grid and cloud computing environment, IEEE IT Professional Magazine

74. V. Dastjerdi, H. Tabatabaei (2009) Distributed intrusion detection in clouds using mobile agents, 3rd International Conference on Advanced Engineering Computing and Applications in Sciences ADVCOMP ¨09 pp 175 – 180

75. Y. Guan, J. Bao (2009) A CP Intrusion Detection Strategy on Cloud Computing, International Symposium on Web Information Systems and Applications (WISA) pp 84–87

76. M. Moradi, M. Zulkernine (2004) A Neural Network Based System for Intrusion Detection and Classification of Attacks, *Proceedings of the IEEE International Conference on Advances in Intelligent Systems - Theory and Applications*

77. J. Han, M. Kamberm (2006) A book on Data Mining Concepts and Techniques 2nd edition Morgan Kaufmann Publishers

78. L. M. Ibrahim (2010) Anomaly Network Intrusion Detection System Based on Distributed Time-Delay Neural Network, Journal of Engineering Science and Technology Volume 5 pp 457 – 471

79. J. Cannady (1998) Artificial Neural Networks for Misuse Detection, National Information Systems Security Conference

80. Grediaga, F. Ibarra, F. García, B. Ledesma, F. Brotons (2006) Application of neural networks in network control and information security LNCS pp 208–213,

81. P. Tillapart, T. Thumthawatworn,, P. Santiprabhob (2002) Fuzzy intrusion detection system Assump University J Technology (A.U. J.T.), Volume 6 pp 109–114

82. S. Chavan, K. Shah, N. Dave, S. Mukherjee (2004) Adaptive neuro-fuzzy intrusion detection systems, *IEEE international conference on information technology: coding and computing (ITCC"04)* pp 70– 74

83. M-Y. Su, G-J. Yu, C-Y. Lin (2009) A real-time network intrusion detection system for large-scale attacks based on an incremental mining approach, Computer Security Volume 28 pp 301–309

84. H. Li, D. Liu (2010) Research on Intelligent Intrusion Prevention System Based on Snort, *International Conference on Computer, Mechatronics* Control and Electronic Engineering (CMCE) Volume 1 pp 251- 253

85. Y. Dhanalakshmi, I. Ramesh Babu (2008) Intrusion detection using data mining along fuzzy logic

and genetic algorithms, International Journal of Computer Science & Security Volume 8 pp 27–32

86. M. Botha, R. Solms, K. Perry, E. Loubser, G. Yamoyany (2002) The utilization of Artificial Intelligence in a Hybrid Intrusion Detection System, SAICSIT pp 149-155

87. C. Katar (2006) Combining multiple techniques for intrusion detection, International Journal of Computer Science & Network Security Volume 6 pp 208–218

88. S. Beg, U. Naru1, M. Ashraf, S. Mohsin (2010) Feasibility of Intrusion Detection System with High Performance Computing: A Survey, International Journal for Advances in Computer Science Volume 1

89. Phil cox (2010) Intrusion detection in a cloud computing environment [Online]. Available: http://searchcloudcomputing.techtarget.com/tip/Intrusion-detection-in-acloud-computing-environment [Accessed in September 2013]

90. Pinal J. Patel, Dr. J. S. Shah, Jinul Patel (2017) Performance Analysis of Neural Networks for Intrusion Detection System, International Journal of Computer Technology & Applications Vol 8(2) 88-93