Computers, Materials & Continua
DOI:10.32604/cmc.2022.022351
Article

Tech Science Press

Deep Learning Based Intrusion Detection in Cloud Services for Resilience Management

S. Sreenivasa Chakravarthi[1,*], R. Jagadeesh Kannan[2], V. Anantha Natarajan[3] and Xiao-Zhi Gao[4]

[1]CSE, Center for Intelligent Computing, Sree Vidyanikethan Engineering College & School of Computer Science and Engineering, VIT, Chennai, Tamilnadu, 600127, India
[2]School of Computer Science and Engineering, VIT, Chennai, Tamilnadu, 600127, India
[3]CSE, Center for Intelligent Computing, Sree Vidyanikethan Engineering College, Tirupati, 517102, India
[4]School of Computing, University of Eastern Finland, Kuopio, 02130, Espoo, Finland
*Corresponding Author: S. Sreenivasa Chakravarthi. Email: sreenivasachakravarthi3@gmail.com
Received: 04 August 2021; Accepted: 25 October 2021

Abstract: In the global scenario one of the important goals for sustainable development in industrial field is innovate new technology, and invest in building infrastructure. All the developed and developing countries focus on building resilient infrastructure and promote sustainable developments by fostering innovation. At this juncture the cloud computing has become an important information and communication technologies model influencing sustainable development of the industries in the developing countries. As part of the innovations happening in the industrial sector, a new concept termed as 'smart manufacturing' has emerged, which employs the benefits of emerging technologies like internet of things and cloud computing. Cloud services deliver an on-demand access to computing, storage, and infrastructural platforms for the industrial users through Internet. In the recent era of information technology the number of business and individual users of cloud services have been increased and larger volumes of data is being processed and stored in it. As a consequence, the data breaches in the cloud services are also increasing day by day. Due to various security vulnerabilities in the cloud architecture; as a result the cloud environment has become non-resilient. To restore the normal behavior of the cloud, detect the deviations, and achieve higher resilience, anomaly detection becomes essential. The deep learning architectures-based anomaly detection mechanisms uses various monitoring metrics characterize the normal behavior of cloud services and identify the abnormal events. This paper focuses on designing an intelligent deep learning based approach for detecting cloud anomalies in real time to make it more resilient. The deep learning models are trained using features extracted from the system level and network level performance metrics observed in the Transfer Control Protocol (TCP) traces of the simulation. The experimental results of the proposed approach demonstrate a superior performance in terms of higher detection rate and lower false alarm rate when compared to the Support Vector Machine (SVM).

Keywords: Anomaly detection; network flow data; deep learning; migration; auto-encoder; support vector machine

1 Introduction

Cloud computing has been in the lime light for around two decades and it holds many features for improving business efficiencies, cost-benefiting, and advantages over conventional computing mechanisms. As per a recent survey by the international data group already 69% of the business firms are utilizing cloud services and 18% of the remaining has plans to implement cloud services at any of the point in their business operations in near future. At the same time a report from the Dell Inc., says that the firms who have adapted to modern big data computing, cloud services and security are earning 53% faster revenue than their competing firms. These reports exhibits that the business firms and the leaders are reaping the benefits of the cloud services in their business operations. They use this modern state of art cutting edge technologies to efficiently implement their operations, provide better service to their customers, and in parallel achieve high profit margins. Gartner predicts an exponential growth of cloud services industry by 2022. The worldwide public cloud services market is projected to grow 17.5 percent in 2019 to total $214.3 billion, up from $182.4 billion in 2018, according to Gartner, Inc. The world economy greatly relies on the manufacturing industries for employment and wealth creation. The cloud computing has become the optimal solution for industry to implement their automation processes by adapting to machine-to-machine translation. Also for storing and managing the ever increasing production and other data, use of cloud services become essential. The various services offered to the end users of cloud in manufacturing sector are presented in Fig. 1.

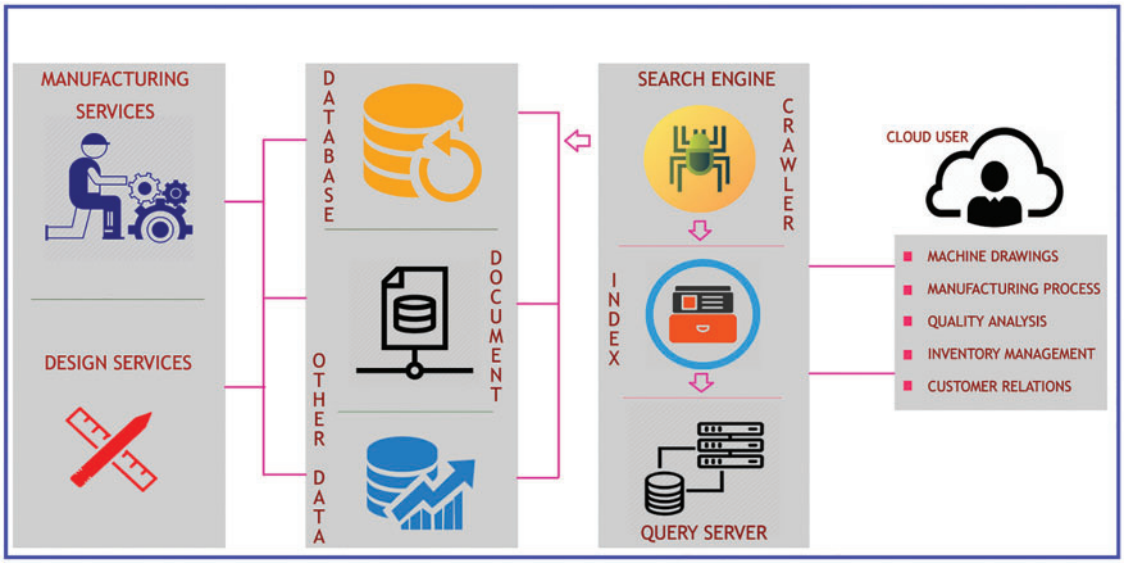

Figure 1: Cloud services for end users of manufacturing sector

The cloud computing environment faces a number of security challenges and most of them can be fixed up to a certain extent using current anomaly based Intrusion Detection Systems (IDS) [1]. IDS in cloud networks have an important role to play in providing security against attacks from both insiders and outsiders. The IDS must be implemented as a part of the cloud services as it is scalable, and efficient in nature. The conventional IDS used to detect attack in internet environment don't have ability to adapt to cloud environment and they are not scalable in nature [2]. Also they are not deterministic in nature and found to be not suitable for cloud environment. Hence a novel and reliable anomaly based intrusion detection system has to be developed and evaluated [3].

Most of the earlier approaches for anomaly detection in cloud environment utilized Machine learning techniques. These techniques have the ability to improve their performance over time by updating its knowledge on the pattern observed in the input data. Whenever a new pattern is observed in the input data the machine learning model parameters are updated to detect the similar anomalies in the future traffic flow [4]. Based on the new information extracted from the previous results the performance of the techniques is improved by changing the execution strategy if required. The various types of machine learning algorithms used for anomaly detection are Bayesian network [5], Genetic algorithm, and neural network [6]. Implementing an anomaly detection mechanism in real time with in the cloud environment involves multiple challenges with respect to performance and scalability of the cloud services. The cloud services are on-demand in nature and hence the anomaly detection must be performed in real-time monitoring. The real time implementation should be highly scalable and must provide support for multiple service providers. This paper focuses on developing an anomaly detection system which has the ability to detect the intrusion accurately, and maintain the resilience of the cloud network. The proposed IDS can be employed to secure the sensitive files, programs, and the ports of the virtual or physical machine in the cloud environment. The network based approach is more efficient when compared to the host based techniques which are unable to detect the attacks in the network and consumes majority of the host computational power and storage resources.

Cloud faces different types of security issues and challenges which affects the growth of cloud services utilization rate. The overall aim is to develop a resilient cloud services for manufacturing sector by implementing an intelligent anomaly detection system as an integrated service within the cloud environment. The objectives of the proposed research work are to identify the service misuse at the client as anomalies and classify the network traffic behavior into two categories as either Normal or Abnormal by deep learning model using network flow data. The efficiency of the proposed method is analyzed based on a dataset created in the cloud simulation. The dataset is comprised of a vector of features extracted from the simulated cloud network with Virtual Machine (VM) migration. The simulation consists of a traffic generator which synthesis both normal and network-based attacks of different categories and intensities. The performance metrics used for studying the efficiency of the anomaly detection in identifying network level attacks are precision, recall, accuracy, and F1-score. For effective implementation of the anomaly detection in the cloud environment, it is designed as a service that can be offered along with infrastructure to the clients. The elasticity of the cloud under various simulated network level attacks and anomaly detection as a service can be evaluated by including VM migration.

The proposed approach includes techniques for efficiently processing, analyzing, and evaluating real time data for detecting anomaly patterns. The challenge is to develop scalable, fault-tolerant anomaly detection method and embed it within a resilient framework for providing warnings promptly in case of adverse conditions during the VM migration. The proposed approach utilizes deep learning architectures to characterize the normal behavior in cloud environment and detect anomalies in the cloud network. The focus is extended to develop methods for capturing and analyzing real time network flow-data in the cloud environment. Further, optimal features are extracted from the captured data and, abnormal traffic patterns are discriminated from the normal traffic patterns using trained Auto Encoder.

2 Literature Survey

In general the architecture of IDS is complex in nature which includes a variety of concepts, and techniques that change with respect to the environment. The working principle of IDS relies on

two main approaches for detecting anomalies and these approaches differ by the analysis method and processing techniques. The first approach utilizes the signature of the abnormal traffics and the second approach tracks for a deviation in the normal traffic. The signature based detection approach is considered to be better in terms of lower false alarm rate but they suffer due to inability in detecting newer type of attacks. But the anomaly based detection methods are able to detect an attack for which no signature is available. This paper focus on the anomaly based detection techniques.

The efficiency of the detection process majorly depends on the quality of the features extracted or engineered from the session/flow in the network. Domain knowledge is essential for a better feature engineering and the deep the learning algorithms have the ability to learn features automatically. The algorithms work in end-to-end nature and at present gaining more importance in the IDS research. These algorithms can analyze the raw data, learns features, and classifies normal from abnormal traffic. Convolutional Neural Network (CNN) based detection algorithms have been proposed in literatures which uses Network Security Laboratory (NSL)-Knowledge Discovery in Databases (KDD) and University of South Wales (UNSW)-NB 15 datasets [7]. Initially the feature vectors are converted in to images for further processing. Using one hot coding the nominal features and the dimension of the features were increased. Each 8 byte of the feature vector is considered as a pixel in image. Each feature vector is thus transformed in to an image of size 8∗8 pixels. A three layer CNN was designed to discriminate normal from abnormal traffic. Performance of the designed network was compared with other deep learning architectures including ResNet50 and GoogleNet. The proposed three layer design yields an accuracy of 91.14% on the NSL-KDD data and 94.9% accuracy on the UNSW-NB 15 dataset. In another work the features were extracted using a sparse auto-encode model and the attacks were classified using an XGBoost model [8]. For experimental analysis NSL-KDD dataset was used. The imbalanced dataset was balanced using synthetic minority over-sampling technique algorithm. The minority classes were oversampled and majority classes are further sub divided in to new classes and thus achieving perfect balancing in the dataset.

The sparse Auto-Encoder (AE) introduces inserts a sparsity constraint on the auto-encoder to enhance the ability of detecting new patterns [9]. At last the data was classified by a XGBoost regression model. The model produces an accuracy of 99.6%, 99.17%, 99.50%, 97.13%, and 89.00%, respectively while detecting Normal, Denial of Service (DOS), Probe and Remote-to-Local (R2L) attacks. Deep learning models have influenced a remarkable growth of Big Data Analytics and they suffer when the data is imbalanced or small. These deep architectures have the ability to model highly complex non-linear data distributions and their performance is better in multiple areas of applications [10]. An important sub-field of deep learning is the generative networks that have shown remarkable performance on capturing data distribution and synthesizing new data samples from the same distribution. As Generative Adversarial Network (GAN) has the capacity to better capture and represent the distribution of the data, many of the recent literatures have used the generative networks for detecting anomalies through a variety of approaches [11].

Adversarial learning methodologies could increase the accuracy of detection in small or imbalanced dataset. One of the recent literatures has used GAN for data augmentation [12]. The KDD99 is considered to be one of the oldest dataset which is both unbalanced and doesn't include new attack patterns. When models are trained on the KDD99 dataset they lack the generalizing ability. When the dataset is augmented with GAN model these issues can be overcome. The experiments included 8 attack categories and adversarial based augmentation increased the accuracy of the 7 attack categories. Using deep learning approach a higher level of features is extracted from low-level ones to achieve a powerful representation. They used recurrent architectures for learning patterns from network traffic sequences and detect network attacks [13]. A two phase approach and improved version of CNN was

used for anomaly detection in cloud datacenter networks. They focused on managing an optimal trade-off between minimized error-rate and reduced feature set by using Grey Wolf Optimization (GWO) techniques for feature selection. The anomaly in the network is classified using an improved CNN architecture [14].

A deep learning framework was developed for anomaly detection in cloud workloads where the usage patterns are analyzed for identifying failures in the cloud because of contention for resources. The resource utilization and performance of the working system are reviewed at regular intervals to model the normal and abnormal behaviors in the cloud network. A hybrid deep neural architecture is used to forecast the near future resource utilization and performance measures of the cloud service in the first stage. In stage two of the analysis the hybrid model is used for classifying the cloud behavior as either normal or abnormal. The proposed anomalous detector is evaluated in the virtual environment using docker containers. The hybrid model is constructed by combining bidirectional long short term memory and long short term memory architectures [15].

Using the auto-scaling characteristics of the cloud an online malware detection approach was proposed in [16]. This approach has the ability to detect the abnormality when the cloud service is running. The performance measures in the process level are used to model a CNN for classifying the abnormal traffic. A 2D-CNN was trained on the samples of VMs running in an auto-scaled scenario. The samples from multiple VM have no correlations. The detection accuracy is enhanced by considering the similarity between different VM through a sample-paring technique. Resilience can be considered as the capacity of a system to offer services in an acceptance level under various challenges. In the case of cloud services the resilience is a fundamental characteristics or feature by which new VM's can be created on an on-demand basis when the load becomes high. Majority of the critical infrastructure services offered in the cloud network are not resilient due to issues in the network infrastructure [17].

Automatic detection and classification of traffic pattern as anomalous is a challenging task that was handled using different approaches and techniques proposed in various literatures. Conventional machine learning techniques and algorithms are sub optimal and are not capable of extracting or uncovering patterns from a high dimensional data. They cannot capture the complex patterns in a high dimensional and voluminous data. This is the reason for engaging deep learning techniques and algorithms to detect anomalies in the cloud network. Based on the taxonomy presented in Fig. 2 it is clear that the deep learning techniques are used at present to classify the traffic pattern as either normal or anomalous.

In a recent work [18] based on heterogeneous data prevailing in the cloud network a hybrid detection model was proposed. The proposed method followed two essential steps for detecting the abnormality; first an optimal set of features were selected from the traffic stream using GWO [19] which is a meta-heuristic algorithm based on evolutionary approach. Then they are classified using CNN either as benign or anomalous traffic.

To train a deep learning model a large volume of training samples are required and these data are preprocessed to resolve the problem of high dimensionality using dimensionality reduction, clustering, and sampling techniques. Later these data are discriminated using a deep classifier. For constructing a deep classifier a training and test data set are essential and additionally a validation data is used to tune the model hyper parameters. Using the training dataset the model parameters are tuned and test data is used for evaluating the performance of the trained model [20]. The process of training and evaluating the deep model performance is presented in Fig. 3. Deep learning algorithms are subset of

machine learning and the performance of the deep learning models are superior to the conventional machine learning or shallow algorithms in majority of application context.

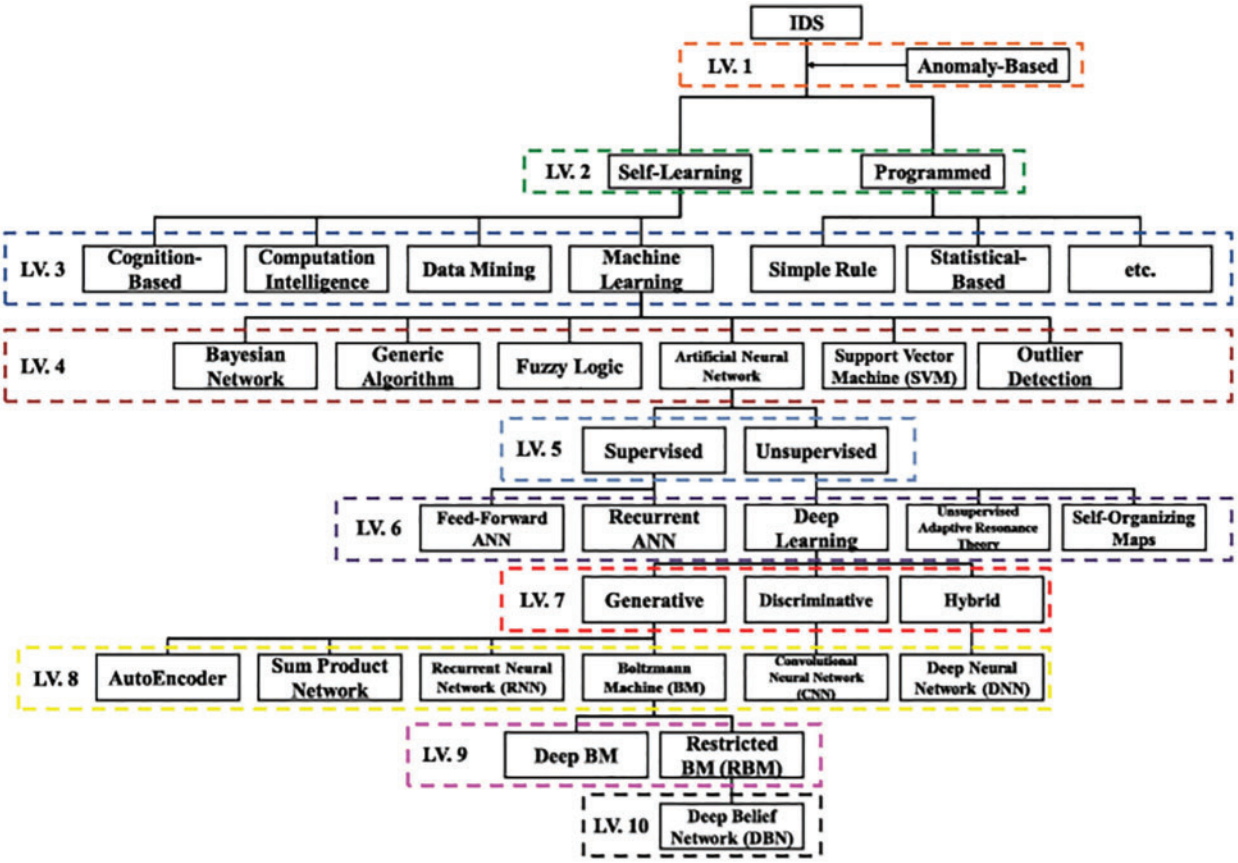

Figure 2: Taxonomy of deep learning based anomaly detection

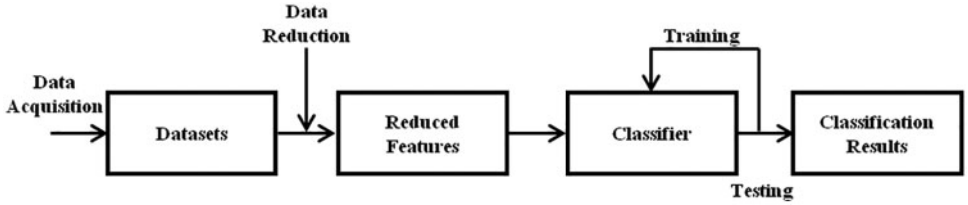

Figure 3: General process flow in classification of traffic patterns

In the cloud network the traffic originates from different heterogeneous sources and it varies immediately due to the elasticity of the services offered to the clients. The large volume of normal traffic and in contrast a low volume of abnormal traffic in the network poses certain challenges. Some of conventional IDS follow signature based approach in detecting the abnormal attacks. In contrast to signature based techniques, anomaly based techniques have been used in cloud computing at various levels which are highly capable of finding new attacks.

The inbound and outbound network traffics are continuously monitored by the IDS and upon further analysis it raises an alarm when there is anomaly being detected. Based on the detection approach used, the IDS can be classified as either signature based or anomaly based intrusion detection. The predefined rules are matched against the pattern extracted from the present network traffic data. Then they are classified as intrusion attack if they vary from the usual network traffic pattern. This approach yield high accuracy in detecting the known category of attacks and generates low false alarm rate. It does not have ability to detect the new category of attacks as the predefined rules don't match the pattern observed in the new attacks. The anomaly based intrusion detection has the ability to detect even new category of attacks. The accuracy of the anomaly based detection is more when compared to the signature based approach as per the theoretical proofs specified in the literatures. The main drawback of the anomaly based detection is high false alarm rate [21]. The main challenges faced while designing anomaly detection are selecting an optimal set of feature from the network traffic and less volume of supervised labeled dataset. The patterns observed from the intrusion attacks are changing from period to period and hence a common set of features chosen for differentiating attacks from normal network traffic flow cannot be suitable in all the cases.

The analysis of network traffic provides more insights on the behavior on the performance of the cloud environment. Due to such revenue growth of the cloud service providers it becomes essential to develop cloud traffic monitoring and analysis methods to increase the availability, and security of the cloud environment. Monitoring and analyzing such a huge volume of network traffic data is a more challenging task. The traditional methods used for monitoring and analyzing network traffic will not suit for cloud environment. The cloud network pattern differs greatly from the patterns observed in a corporate distributed network.

3 Proposed Methodology

Different type of anomaly detection techniques have been implemented so far in various literatures, but majority of them has not focused on the analysis of impact of elasticity of the cloud during VM migration. The proposed methodology focuses on evaluating the performance of the anomaly detection under such challenging conditions. The support for service migration and migration of VM to other physical nodes in the cloud network exploits the elastic property of the cloud for dynamic movement of the cloud resources. For effective management of cloud resources in online and perfect balancing of computing load across physical nodes of the cloud; this real time migration becomes essential. The anomaly detection module might consider the live migration as an anomaly (false positive) or some time an anomalous event occurring during the migration the detection may be masked (false negative).

The network packets are used as source of data for the detection of anomalies and hence the U2Land R2L attacks can be detected effectively. As the packet headers contain the IP addresses the attack source can be detected precisely. Analysis of information extracted from the packets can be done in real time. A single packet does not reveal much information about the context and hence detecting attacks like distributed DoS become difficult. The detection process includes parsing packets, and analysis of the packet payloads. Fig. 4 presents the schematic view of packet based real time anomaly detection.

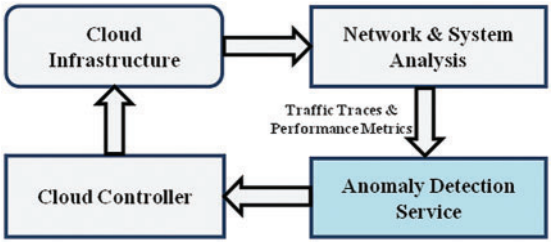

Figure 4: Real time anomaly detection

3.1 Training Data

For efficient training of deep learning models a large volume of balanced anomaly dataset is essential and it is obtained from the traces collected during the simulation of the cloud environment. The traces of TCP streams obtained from the simulated cloud network include both genuine and attack traffic. The effects of VM migration will also be reflected in the traces. The incoming traffic to the physical node becomes different as the VM migrates to a different physical machine. Also the anomalies especially volume based attacks will be best characterized by the traces of the TCP streams as they consume more bandwidth from the normal traffic. The data for training the deep learning model is collected at various level of the cloud including the traffic level, hypervisor and physical machine level. A feature vector is extracted from each one second bin of the traffic traces and the system level performance metrics such as CPU utilization rate, memory utilization, network bandwidth consumption, number of I/O operations, and number of processes waiting for execution in the queue were considered. The network level performance metrics included in the feature vector are number of lost packets, volume of traffic in the network port, and overall load rate of the network.

3.2 Data Augmentation Using GAN

The acquired dataset is highly imbalanced and an efficient model cannot be build using this dataset. Hence the dataset must be balanced before training the model. The aim is to use GAN architectures to discover the pattern in the data that makes them more realistic as uncovering patterns in the data is not possible using other methods. The generative networks help to balance the dataset by synthesizing new samples for minority classes. It is proved that GAN have shown impressive results when compared to other generative architectures including variation AE, and restricted boltzman machines. Synthesizing new samples is a complex task when the dimensionality of the data is high [22]. From the literature review it was obvious that variation AEs [23] and GANs [24] are most successfully used deep architectures for data augmentation. The objective functions for GANs can be chosen among Jensen-Shannon [24] and f-divergences [25]. There are other several methods to define the distance or divergence between the distribution of model generated and real data. The discriminator network in the GAN output the probability that a given data is real or synthesized. The discriminator is given a set of input that includes both real and synthesized data. It generates a probability estimate for each input. The loss of the discriminator network can be measured by cross-entropy function. Cross entropy based loss function is similar to the Jensen-Shannon objective function and they tend to fail in certain cases [26]. To overcome such issues Wasserstein distance based loss function is used. The cross entropy loss function measures the accuracy in the detection of real and synthesized data by the discriminator network. The distribution of each vector in the real and synthesized data is estimated and the distance between them is calculated. It gives the measure of how much mass times distance is required to make the distribution of synthesized data similar to the distribution of real data.

The flow of input data through layers in the GAN network is presented in Fig. 5. Mathematical representation of the min-max loss function used in the GAN network is given in Eq. (1) where the generator network attempts to minimize it while the discriminator network tries to maximize it.

$$E_x[\log(D(x))] + E_z[\log(1 - D(G(z)))] \tag{1}$$

where D(x) is the probability that the generated data is real estimated by the discriminator; E_x is the expected values for all the data samples; $G(z)$ is the output of the generator network when the random noise is given as input; $D(G(z))$ is the probability that the fake instance is real; E_z is the expected value for all the fake instances $G(z)$ synthesized by the generator. The mathematical expression is derived from the cross entropy between the real and generated data distributions. The discriminator network produces output for the generator to improve its performance. The generator learns nothing when the gradient of the generator network diminishes and approaches close to zero.

$$-\nabla_\theta \log(1 - D(G(z))) \to 0 \tag{2}$$

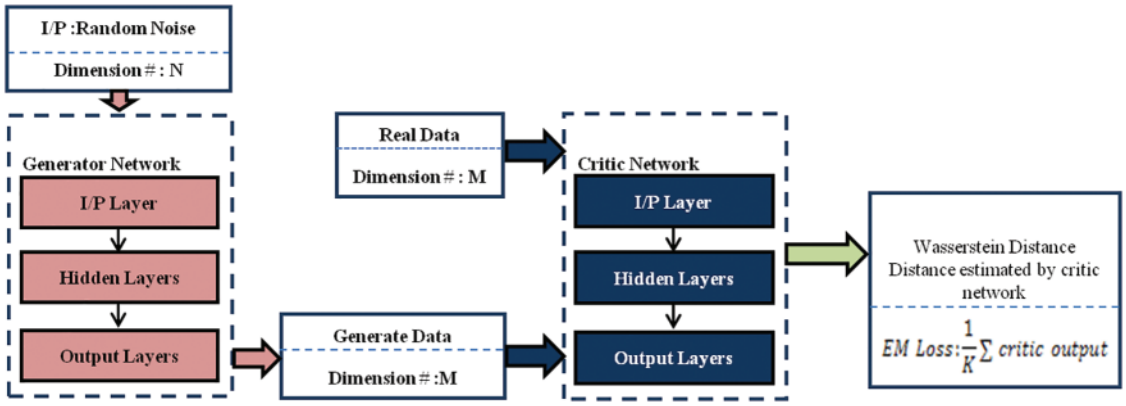

Figure 5: Process flow in wasserstein GAN

To overcome the vanishing gradient problem the following alternate cost function can be used:

$$\nabla_\theta \log(1 - D(G(z^{(i)}))) \tag{3}$$

In [23] the authors have illustrated that the alternate cost function has large variance of gradients which causes the model unstable. Also they suggested that adding noise to the generator output to stabilize the model. The schematic view of the Wasserstein GAN (WGAN) is shown in Fig. 6.

Rather than adding noise, a new cost function based on the Wasserstein distance was proposed in [26]. Based on the Kantorovich-Rubinstein duality [27] the Wasserstein distance can be expressed as given below.

$$W(p_r, p_g) = \sup_{\|f_L\| \leq 1} \mathbb{E}_{x \sim p_r}[f(x)] - \mathbb{E}_{x \sim p}[f(x)] \tag{4}$$

where *sup* represents the least upper bound and f denotes a 1-lipschitz function which obeys the following constraint

$$|f(x_1) - f(x_2)| \leq |x_1 - x_2| \tag{5}$$

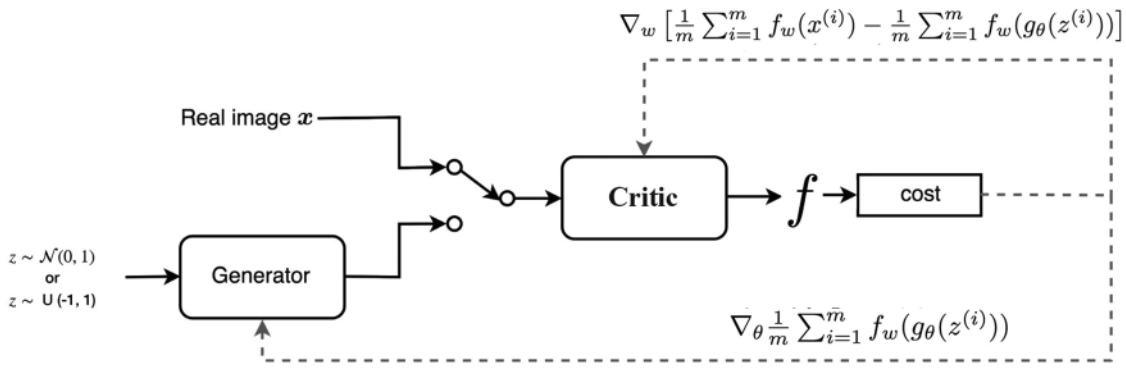

Figure 6: Schematic view of training a WGAN with added noise to the generator output

When the lipschitz function is derived from *K*-lipschitz functions parameterized by w, $\{f_w\}w \in W$. The discriminator network is trained to learn the lipschitz function to compute the Wasserstein distance. When the value of loss function decreases in the training then the Wasserstein distance gets smaller and the model of the generator becomes closer to the distribution of real data.

The loss function can be formulated as to find the Wasserstein distance between p_r and p_g

$$L(p_r,\ p_g) = \ W(p_r,\ p_g) = \ \max_{w \in W} \ \mathbb{E}_{x\sim p_r}[f(x)] - \mathbb{E}_{z\sim p_{r(z)}}[f_w(g_\theta(z))] \tag{6}$$

In order to maintain the ***K***-lipschitz continuity of $f_w$during the training process the weights are clamped after every gradient update to a small window such as [−0.01, 0.01]. Thus the lipschitz's continuity can be preserved by obtaining the lower and upper bound of f_w.

3.3 Anomaly Detection

The anomalous traffic are detected based on the compact representation of the feature vectors obtained using a deep AE. The AE learns to map the given input data to a compact representation with two unsupervised training steps. The AE can be classified as a generative model and it has the ability to learn and extract the similarity and correlation in the input data [28]. An efficient intrusion detection model must be sensitive to the input, yield less reconstruction error, and it should not get overfit for certain input data. The loss function is designed with two terms to achieve the above mentioned constraints.

$$L(x, \hat{x}) + (\alpha * \ regularizer) \tag{7}$$

The first part of the above equation makes the model sensitive to the given input data and the second terms prevent the model from overfitting on the training data. The trade-off between these two different objectives can be achieved by tuning the alpha scaling parameter. Based on the nature and characteristics of the given output the AE model can be considered as non-linear generalization of principal component analysis and the AE model has the ability to learn non-linear relationship between given input and expected output. The AE model helps to separate the normal data from the anomalous data by transforming the given input data on to new axes. The AE consists of two neural networks namely encoder and decoder. The encoder compresses the data points into a lower dimensional representation and the decoder network attempts to reconstruct the original input points from the latent representation generated by the encoder network. The AE parameters are tuned

by minimizing the reconstruction error which is the difference between the input data points and reconstructed data points. The AE is trained in an unsupervised mode with features extracted from normal traffic data both under normal circumstances and VM migration. The AE will be capable of reconstructing the normal traffic data points and fails to reconstruct the anomalous traffic data points. The reconstruction error is used as anomalous score. The performance of the AE trained in an unsupervised learning strategy is compared with a binary classifier, SVM. The SVM model is trained with a Radial Basis Function (RBF) kernel function.

$$\mathrm{K}(\mathrm{X}, \mathrm{X}') = \exp[-\gamma \|x - \mathrm{x}'\|^2] \tag{8}$$

The non-linear kernel represents the similarity between two different vectors as a function of the squared norm of their distance. That is, if the two vectors are close together then, $\|\mathrm{x} - \mathrm{x}'\|$ will be small. Then, so long as $\gamma > 0$, it follows that $-\gamma\|x - \mathrm{x}'\|^2$ will be larger. Thus, closer vectors have a larger RBF kernel value than farther vectors.

4 Experiments and Results

The simulation of the cloud environment is accomplished in CloudSim 5.0 and the metricslisted in Section 3.1 are extracted from the normal network traffic. For creating the VM migration within the simulated cloud environment, initially the list of over utilized host is collected and the backup of over utilized VM to be migrated is done. Then they are mapped to a new suitable host using a migration map. The VM migration is considered to essential to test the resilience of the cloud services under different attacks when they are exposed to a variety of attacks. The following attacks were introduced in the simulated cloud network; Net Scan (NS), and DoS. The network traces and the metrics of the host machine are collected under multiple time instances of the simulation. Then required features are extracted from the traces and labeled as normal or abnormal respectively. Two set of feature set were extracted; one under VM migration and another without migration. Both during migration and normal period anomalous traffic were introduced in the cloud simulating the above mentioned attacks. The model was tested to detect network level attacks as more end users will be accessing the services of the cloud thus increasing the attack surface. More computational power will be allocated in the form of VM in on-demand basis. The anomalous traffic is generated by injecting the attacks into the legitimate traffic at irregular intervals. The VMs are migrated live among the nodes during the simulation may be during normal traffic period or anomalous traffic period. The network traces collected for every 1-second been using a packet analyzer script embedded within the simulated network. The harmonic mean F score and geometric mean provides accurate measure of the performance of a particular anomalous detector keeping all the outcomes to certain degree. The network traffic in the cloud simulation is fixed constant and the cloud environment is simulated with five VMs running web servers. There are two physical host nodes available in the simulation; one with three VMs and another host contains two VMs. The web traffic is generated from a VM to another VM running inside another host node using TRex, an open source traffic generator.

As described earlier the data augmentation task is implemented using WGAN to balance the volume of normal and anomalous traffic. The threshold value c, considered as one of the important hyperparameter is fixed based on Bayesian optimization technique. The *RMSProp* optimization algorithm is utilized instead of momentum based optimization algorithm like Adam. The Adam algorithm causes instability of the model during the training process. The performance of the WGAN model is sensitive to the clipping hyper parameter. The graph shown in Fig. 6 shows the explosion gradient when the value of **c** is varied from 0.001 to 0.1.

The weight clipping method acts as a weight regulator and it reduces the performance of the model thereby limiting the capacity of the model to learn complex function. Hence instead of gradient clipping, Gradient penalty technique was adopted in the experiments. The plot in Fig. 7 illustrates the advantage of using gradient penalty over gradient clipping. When the value of weight clipping threshold c is fixed either low or high values then the gradient either explodes or vanishes. Batch normalization is included in the discriminator network as it creates correlation between samples in the same batch. The experimental evaluations showed that it impacts the effectiveness of the gradient penalty.

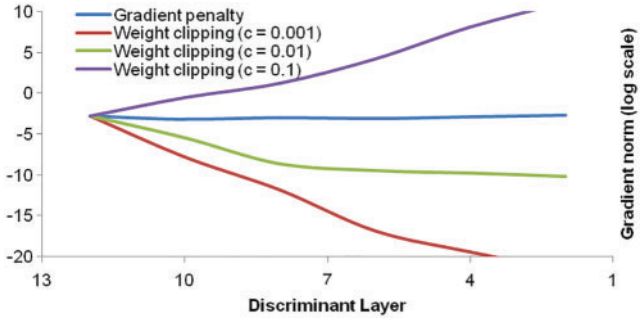

Figure 7: Plot of gradient norm in log scale

Gradient penalty was calculated based on the following procedure;

1. Estimate the gradient values with respect to the input data.
   a) Create a combined data by weighing the synthetic and real data using epsilon and fusing them together.
   b) Compute the discriminator's output for the fused data.
2. Calculate the gradient penalty for the estimated gradient.
   a. Consider the magnitude the gradient
   b. Find the penalty

The performance of the AE and SVM based classification was analyzed using Receiver operating Characteristics (RoC) curves shown in Figs. 8 and 9. Based on the experimental results under VM migration the performance of the anomaly detection under both volume based attack (DoS) and non-volume based attack (NS) using SVM is degraded. From the RoC curves presented in Figs. 8a and 8b, it is evident that the SVM classifier is sensitive to the type and volume of the anomalous traffic. At few instances the performance of the SVM classifier degraded when VM migration occurs. In case of volume based attack, DOS the SVM performs for both low and high intensity attacks but under influence of VM migration its performance decrease. The True Positive Rate (TPR) is decreased by 32% during VM migration even for low density NS attack. Fig. 8b presents the influence of migration for DoS attack detection; also it is observed that more than 85% of the TPR is achieved during with and without migration. The False Positive Rate (FPR) is increased by 5% when the migration starts. In parallel from Fig. 9a it can be observed that almost more than 85% of the anomalous traffic was detected under high anomalous (NS) traffic scenario for both with and without migration. Even for the same attack types under low anomalous traffic the model was able to yield an average of 80% of TPR with less false positive rate. For DoS attack the AE based anomaly detection is capable of identifying anomalous traffic for both attack types and it is not affected under migration. The FPR rate is less than 15% and it is acceptable under migration of VM. From the analysis of the results

it is inferred that the virtual migration has no impact on the performance of the AE in detecting the anomalies. More traces are samples from the same scenario in order make the detection more efficient.

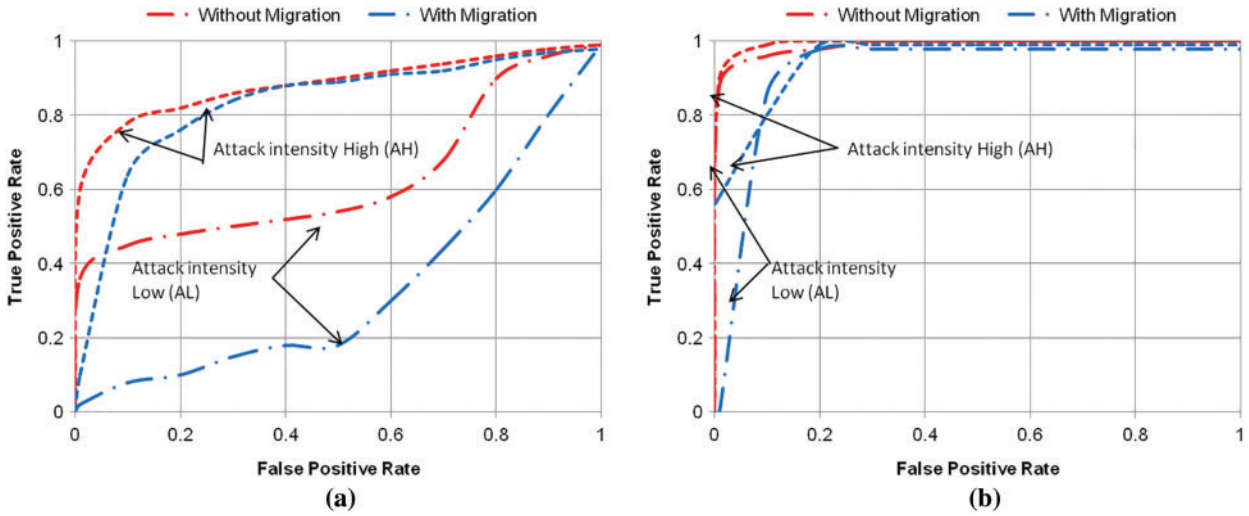

Figure 8: (a) RoC for NS using SVM (b) RoC for DoS using SVM

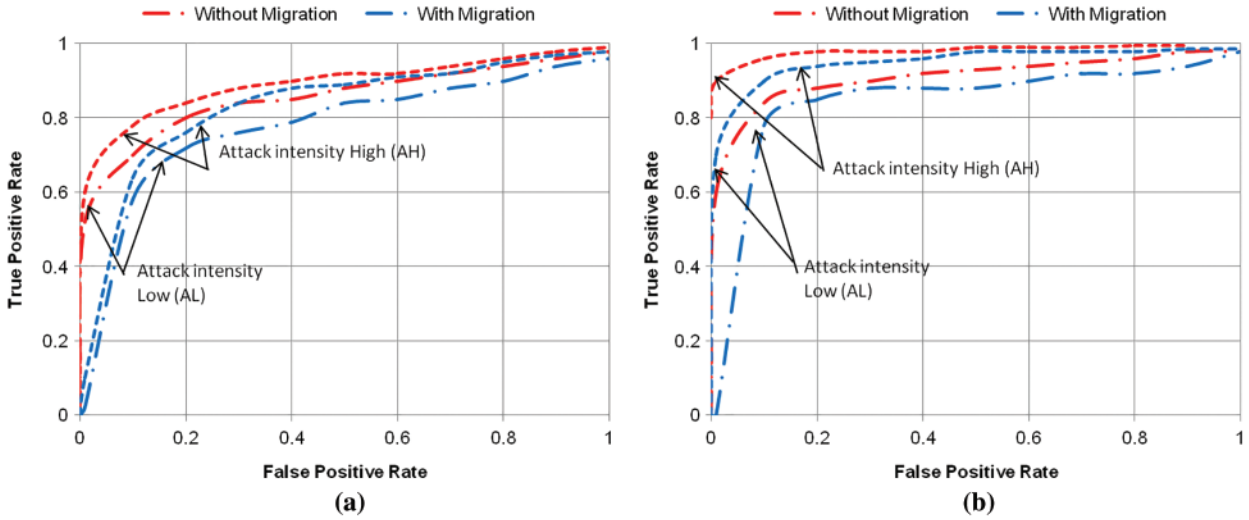

Figure 9: (a) RoC for NS using AE (b) RoC for DoS using AE

The results presented in Tab. 1 show the efficiency of the anomaly detection for network based and volume based attacks. The virtual migration has affected the performance of the detector only low density anomalous traffic in the simulation as the False Positive Rate was observed to be high during this period. The detector classifies the migration traffic also as anomalous. The increase in the FPR is also only 5% to 10% which does not impact the overall performance of the classifier. The feature set used for detection can be further expanded by including few statistical features derived from them. These additional features can be useful while detecting certain other complex attacks. Fig. 10 presents the plot of the training loss curve, for both discriminator and generator network over

generator iterations for the following models; WGAN with weight clipping and *RMSProp* and WGAN with gradient penalty with *ADAM*.

Table 1: Performance of AE in anomaly detection

Anomalous traffic density	VM migration	Recall	Precision	Accuracy	F-score	G-mean
High	Yes	0.9248	0.8795	1.00	0.9365	0.9378
	No	0.9895	0.9912	1.00	0.9923	0.9934
Low	Yes	0.8806	0.8216	0.9856	0.8950	0.8978
	No	0.9694	0.9585	0.9842	0.9612	0.9742

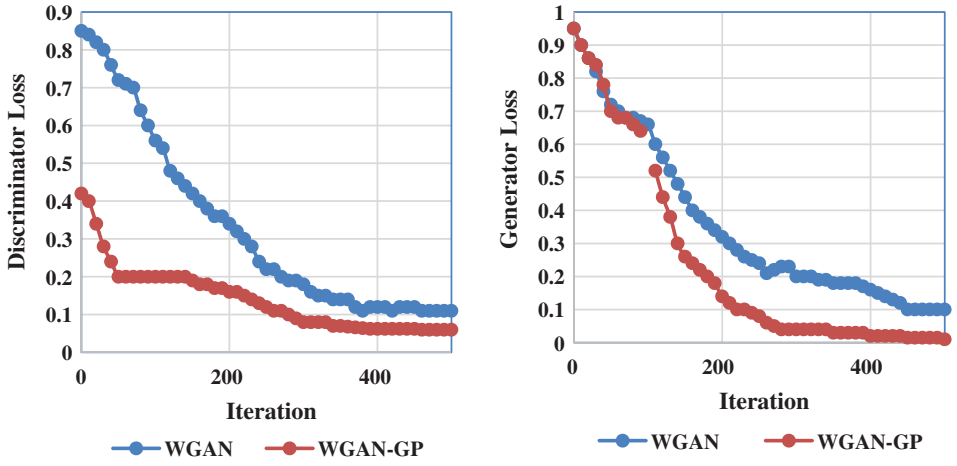

Figure 10: Training loss analysis

For analyzing the performance of the VMs deployed with certain workloads memory, CPU utilization, and cap values are tabulated in Tab. 2. Each VM hosts a selected workload by allocating one or more CPU cores, RAM from 512MB (shown in Fig. 11) – 2048MB. Simultaneously the CAP value is varied between the range 25% and 200% of the CPU utilization and the estimated the corresponding CPU utilization rate**.**

The proposed hybrid approach does include a deep architecture for synthetic generation of samples and a conventional machine learning algorithm, SVM for detecting anomalies. The selection of important hyper-parameters and architecture of the deep network introduces certain computational complexity as more layers are used to construct the deep model. In general the linear SVM detects the anomaly with a complexity of $O(d)$ where d is the dimension of the input data. The experiments adopted an RBF kernel which has a complexity of $O(d^2)$ whereas the polynomial kernel will have a complexity of $O(n \times d)$ where n denotes the number of supper vectors. During experiments it was observed that the complexity of the deep network increases as the depth of the network is increased. It was also observed that when optimal values of hyper parameters are configured for the network the synthetic samples were more similar to the original input data which shows that the depth of the network decides the performance of the network.

Table 2: CPU utilization in %

Cap	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	AVG
25	6.25	6.22	6.22	6.22	6.22	6.3	6.28	6.2	6.22	6.22	6.235
50	12.5	12.55	12.57	12.35	12.57	12.47	12.43	12.6	12.45	12.53	12.502
75	18.55	18.85	18.77	18.65	18.85	18.57	18.85	18.65	18.8	18.85	18.739
100	24.82	24.82	24.82	24.85	24.55	24.85	24.85	24.85	24.82	24.82	24.805
125	24.85	24.82	24.82	24.85	24.85	24.82	24.85	24.85	2.485	24.82	24.838
150	24.85	24.85	24.85	24.85	24.85	24.85	24.85	25.82	24.85	25.82	24.85
175	25.82	25.82	24.85	24.85	24.85	24.85	24.85	25.82	24.85	25.82	25.238
200	24.82	24.85	24.85	24.82	24.85	24.85	24.852	24.85	24.85	25.82	24.941

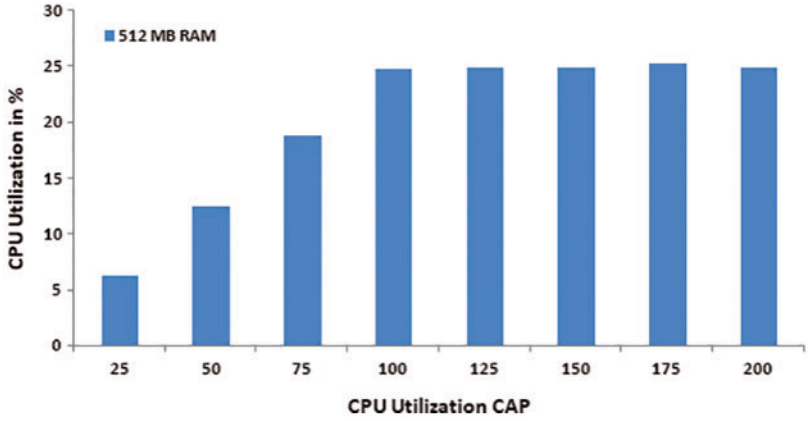

Figure 11: Analysis of CPU utilization (allocated RAM = 512 MB)

5 Conclusions

This paper explored the effect of VM migration on the performance of the anomaly detection and proposed features and robust classification approach to manage the resilience of the cloud service to overcome security issues. Experiments were conducted by simulating the VM migration and varying the density of the anomalous traffic in the network. The deep learning based detection mechanism and the features extracted from the network traces helped to retain the resilience of the cloud environment. The reconstruction error of the AE model is used as the anomalous score to detect deviation in the network traffic patterns. This anomaly detection is tested in a simulation environment wherein the anomaly detection was executed in parallel with other events of the simulation. This work focused on detecting two attacks namely NS and DoS. Future work will focus on detecting more number of attacks by enhancing the features set used in this work with more optimal features. The results are found to be satisfactory in resolving the security concerns in cloud services for its application in manufacturing sector.

As a benchmarked dataset is not available to test the resilience of the cloud infrastructure, data samples from the simulated network have been generated, balanced using GAN network and classified as either anomalous or normal using an AE model. The trained model is able to detect anomalous traffic only in similar cloud environment simulated in the experiments. To overcome this limitation

and develop a generic deep learning based anomaly detection system further data samples must be collected from a real time network, benchmarked and must be used for training the deep learning model.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Iqbal, M. L. M. Kiah, B. Dhaghighi, M. Hussain, S. Khan *et al.,* "On cloud security attacks: A taxonomy and intrusion detection and prevention as a service," *Journal of Network and Computer Applications*, vol. 74, pp. 98–120, 2016.

[2] V. Adat and B. B. Gupta, "Security in internet of things: Issues, challenges, taxonomy, and architecture," *Telecommunication Systems*, vol. 67, no. 3, pp. 423–441, 2018.

[3] A. Aldweesh, A. Derhab and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," *Knowledge Based Systems*, vol. 189, pp. 105124, 2020.

[4] N. Moustafa, G. Creech and J. Slay, "Big data analytics for intrusion detection system: Statistical decision-making using finite dirichlet mixture models," in *Data Analytics and Decision Support for Cybersecurity*, 1st ed., NY, USA: Springer International Publishing, pp. 127–156, 2017.

[5] L. Nie, D. Jiang and Z. Lv, "Modeling network traffic for traffic matrix estimation and anomaly detection based on Bayesian network in cloud computing networks," *Annals of Telecommunications*, vol. 72, no. 5, pp. 297–305, 2017.

[6] Z. Chiba, N. Abghour, K. Moussaid, A. El Omri and M. Rida, "Novel network ids in cloud environment based on optimized bp neural network using genetic algorithm," in *3rd Int. Conf. on Smart City Applications*, Tetouan Morocco, pp. 1–9, 2018.

[7] S. Potluri, S. Ahmed and C. Diedrich, "Convolutional neural networks for multi-class intrusion detection system," in *Int. Conf. on Mining Intelligence and Knowledge Exploration*, Cluj-Napoca, Romania, pp. 225–238, 2018.

[8] B. Zhang, Y. Yu and J. Li, "Network intrusion detection based on stacked sparse auto encoder and binary tree ensemble method," in *IEEE Int. Conf. on Communications Workshops*, Kansas City, MO, USA, pp. 1–6, 2018.

[9] A. Brock, J. Donahue and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *Int. Conf. on Learning Representations*, New Orleans, LA, USA, pp. 1–35, 2019.

[10] R. Shetty, M. Fritz and B. Schiele, "Adversarial scene editing: automatic object removal from weak supervision," in *32nd Conf. on Neural Information Processing Systems*, Montréal, Canada, pp. 1–11, 2018.

[11] C. Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu *et al.,* "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," *AAAI Conference on Artificial Intelligence*, New Orleans, Louisiana, USA, vol. 33, no. 1, pp. 1409–1416, 2018.

[12] H. Zhang, X. Yu, P. Ren, C. Luo and G. Min, "Deep adversarial learning in intrusion detection: A data augmentation enhanced framework," *Cryptography and Security*, arXiv preprint arXiv:1901.07949, pp. 1–10, 2019.

[13] X. Yuan, C. Li and X. Li, "Deep defense: Identifying DDoS attack via deep learning," in *IEEE Int. Conf. on Smart Computing*, Hong Kong, China, pp. 1–8, 2017.

[14] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya *et al.,* "A hybrid deep learning-based model for anomaly detection in cloud datacenter networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 924–935, 2019.

[15] S. Gupta, N. Muthiyan, S. Kumar, A. Nigam and D. A. Dinesh, "A supervised deep learning framework for proactive anomaly detection in cloud workloads," in *14th IEEE India Council Int. Conf.*, Roorkee, India, pp. 1–6, 2017.
[16] M. Abdelsalam, R. Krishnan and R. Sandhu, "Online malware detection in cloud auto-scaling systems using shallow convolutional neural networks," *Annual Conf. on Data and Applications Security and Privacy*, Charleston, SC, USA, pp. 381–397, 2019.
[17] J. Moura and D. Hutchison, "Review and analysis of networking challenges in cloud computing," *Journal of Network and Computer Applications*, vol. 60, pp. 113–129, 2016.
[18] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya *et al.,* "A hybrid deep learning-based model for anomaly detection in cloud datacenter networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 924–935, 2019.
[19] S. Mirjalili, S. M. Mirjalili and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
[20] S. Naseer, Y. Saleem, S. Khalid, M. K. Bashir, J. Han *et al.,* "Enhanced network anomaly detection based on deep neural networks," *IEEE Access*, vol. 6, pp. 48231–48246, 2018.
[21] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim *et al.,* "A survey of deep learning-based network anomaly detection," *Cluster Computing*, vol. 22, no. 1, pp. 949–961, 2019.
[22] R. M. Neal, "Annealed importance sampling," *Statistics and Computing*, vol. 11, no. 2, pp. 125–139, 2001.
[23] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Int. Conf. on Learning Representations*, Banff, AB, Canada, pp. 1–14, 2014.
[24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley *et al.,* "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, pp. 2672–2680, 2014.
[25] S. Nowozin, B. Cseke and R. Tomioka, "F-Gan: Training generative neural samplers using variational divergence minimization," in *30th Int. Conf. on Neural Information Processing Systems*, Barcelona, Spain, pp. 271–279, 2016.
[26] M. Arjovsky, S. Chintala and L. Bottou, "Wasserstein generative adversarial networks," in *Int. Conf. on machine learning*, Sydney, Australia, pp. 214–223, 2017.
[27] N. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 1, pp. 41–50, 2018.
[28] M. Yousefi Azar, V. Varadharajan, L. Hamey and U. Tupakula, "Auto encoder-based feature learning for cyber security applications," *Int. Joint Conf. on Neural Networks*, Anchorage, AK, USA, pp. 3854–3861, 2017.