Enhancing Cloud Security: An Optimization-based Deep Learning Model for Detecting Denial-of-Service Attacks

Lamia Alhazmi
Department of Management Information System,
College of Business Administration, Taif University,
P.O Box 11099, Taif, 21944,
Saudi Arabia

***Abstract*—DoS (Denial-of-Service) attacks pose an imminent threat to cloud services and could cause significant financial and intellectual damage to cloud service providers and their customers. DoS attacks can also result in revenue loss and security vulnerabilities due to system disruptions, interrupted services, and data breaches. However, despite machine learning methods being the research subject for detecting DoS attacks, there has not been much advancement in this area. As a consequence of this, there is a requirement for additional research in this field to create the most effective models for the detection of DoS attacks in cloud-based environments. This research paper suggests a deep convolutional generative adversarial network as an optimization-based deep learning model for identifying DoS bouts in the cloud. The proposed model employs Deep Convolutional Generative Adversarial Networks (DCGAN) to comprehend the spatial and temporal features of network traffic data, thereby enabling the attack detection of patterns indicative of DoS assaults. Furthermore, to make the DCGAN more accurate and resistant to attacks, it is trained on a massive collection of network traffic data. Moreover, the model is optimized via backpropagation and stochastic gradient descent to lessen the loss function, quantifying the gap between the simulated and observed traffic volumes. The testing findings prove that the suggested model is superior to the most recent technology methods for identifying cloud-based DoS assaults in Precision and the rate of false positives.**

***Keywords*—DOS attack; cloud database; generative adversarial networks; attack detection; security threats**

I. Introduction

Computer networks have become more open to cyber-attacks in recent years due to the expansion of web-associated devices and the rising dependence on internet-based administrations [1]. As shown in Fig. 1, the DoS attack, which involves flooding a targeted network or system with malicious traffic to make it inaccessible to authorized users, is one of the most frequent and disruptive attacks [2]. Network administrators and security experts must identify and prevent DoS attacks [3]. Statistical anomaly detection methods or predetermined signatures are frequently used in traditional forms of identifying DoS assaults [4]. However, these techniques have drawbacks like high false-positive rates, a limited ability to react to changing attack strategies, and a failure to identify previously undetected attacks [5].

Researchers have used deep learning techniques to address these issues, which have shown great promise in areas like PC vision, normal language handling, and discourse recognition [6]. Deep learning models have the potential to properly detect and categorize DoS assaults because they can automatically learn complicated patterns and representations from vast amounts of data [7].

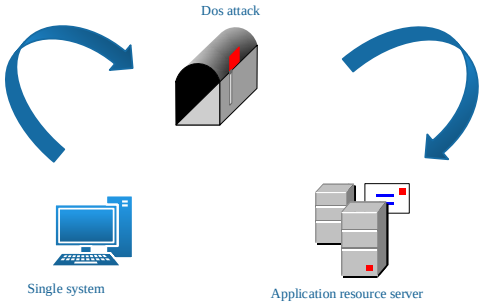

Fig. 1. Basic block diagram of DOS attack.

In this research, we suggest a deep learning model for identifying DoS assaults that are optimization-based. To increase the Precision and stoutness of DoS attack detection, our model takes advantage of deep neural network technology and integrates it with an optimization framework [8]. There are different advantages to recognizing forswearing of administration (DoS) attacks utilizing any procedure, including traditional strategies or state-of-the-art techniques like profound learning: The detection of DoS attacks makes it possible to detect malicious activity before it affects network resources [9]. Network administrators can safeguard the accessibility of essential network resources and services by detecting DoS attacks [10]. Early detection enables prompt remediation, guaranteeing that authorized users may access the network and preventing them from controlling its resources [11]. DoS assaults can significantly disrupt enterprises, resulting in losses in revenue and harm to their brand [12]. Systems for detecting DoS attacks can assist in locating and reducing harmful traffic, enhancing the entire network's performance [13]. It is necessary to filter out malicious traffic to optimally employ the available network resources for legal users and enhance response times and network operations [14].

DoS assaults can differ significantly in power, length, and attack methods. Identifying and correctly categorizing these attacks can be challenging, mainly when dealing with new or emerging attack patterns [15]. DoS attacks are being launched more frequently using encrypted traffic, making detecting and examining illicit activity more challenging [16]. While there are many ways to identify denial of service (DoS) assaults using artificial intelligence (AI), one optimization-based deep learning model that has been applied is based on detecting anomalies. The goal of detecting anomalies is locating odd patterns or actions significantly different from typical network data [17]. Various conventional techniques were employed, such as Particle Swarm Optimization-based Probabilistic Neural Network (PSO-PNN) [18], Recurrent Convolutional Neural Network (RC-NN) [19], Lightweight Random Neural Network (LraNN) [20]; however, not all social media platforms support those.

The key contributions of our proposed work are as follows:

- Initially, the datasets are gathered from the standard web source.
- The system was originally trained using the internet application that contained incursion data.
- Then, the pre-handling stage is completed to eliminate the commotion and blunder values.
- A novel Deep Convolutional Generative Adversarial Network (DCGAN) has been implemented with the necessary characteristics and processing stages.
- Subsequently, the feature extractions are done here; the unwanted features are removed.
- More of the wanted features are trained on the attack prediction model to detect the attack and classify the types of attacks based on the parts.
- Finally, the metrics have been validated and compared with another system through Accuracy, F1- score, Recall, and Precision.

The arrangement of this paper is structured as follows. The related work based on detecting DoS attack is detailed in Section II, and the system model and problem statement are elaborated in Section III. Also, the process of the proposed methodology is described in Section IV. Finally, the achieved outcomes are mentioned in Section V. The results are discussed in Section VI and the conclusion about the developed model is detailed in Section VII. Section VIII gives the future work to be conducted in this area.

II. Related Works

Some of the recent related research works are described below:

Understanding malware's behaviour across the entire behavioural space significantly improves traditional security. Rabbani et al. [21] propose an original technique to upgrade Cloud specialist organizations' ability to demonstrate client conduct. Here he applied a PSO-PNN for the detection and recognition process.

The advantage is that it can be used in safety checking and distinguishing unsafe leads. Since genuine users cannot access resources, they might not be able to find the information they need or take the necessary actions.

The Intrusion detection system is built on an Innovative, custom-optimized RC-NN and the Ant Lion optimization technique, which Thilagam et al. [22] proposed for intrusion detection. The advantage is the high accuracy of the IDS classification model, which raises the rate of detection or error rate. The custom-optimized RC-NN-IDS model has a lower error rate of 0.0012 and an improved classification accuracy of 94%. The most significant disadvantage of deploying an IDS is its inability to respond to or stop attacks once detected.

Samriya et al. [23], to increase, generally speaking, cloud-based registering conditions, another hybridization procedure for interruption location frameworks is proposed. Furthermore, this technique aids in dealing with many types of cloud security challenges. The significant advantage is it reduces computational time and enhances accuracy. The downside of host-based IDS is that it cannot detect network threats against the host.

Kushwah et al. [24] present a DdoS violence finding system based on a Self-Adaptive Evolutionary Extreme Learning Machine (Sae-ELM) that has been improved. The suggested attack detection system outperforms based on the original SaE-ELM and cutting-edge approaches. They overload the system, causing it to fail.

DoS attacks can affect various equipment and technologies used in network infrastructure. Therefore, security systems must be able to recognize these threats. The majority of the methods that have been previously provided employ an individual machine learning model that can pinpoint DoS attacks; however, it appears that combining a variety of learning models will improve the intrusion detection system's detection accuracy and reliability.

Majidian, et al. [25] has developed an Adaptive Neuro-Fuzzy Inference System (ANFIS) to enhance DoS attack detection accuracy compared to existing techniques.

The most alluring invention in the current landscape is probably cloud computing. Lowering the massive upfront cost of buying equipment foundations and processing power provides an expense-effective arrangement. By utilizing a portion of the work that is not registered, decreasing the reaction time at the edge devices of the end client, such as IoT, fog computing provides extra aid to cloud infrastructure.

Sattari, et al. [26] have developed a Software-defined network (SDN) that identifies 99.98% of sophisticated multi-variant bot attacks more accurately than earlier ones, with more excellent performance. Table I lists the difficulties with the existing works.

Ferrag et al. [31] developed a unique Weight-based Ensemble Machine Learning Algorithm (WBELA) to detect aberrant signals in a CAN bus network. Then, he creates a model based on many-objective optimization for CAN bus network intrusion detection. It considerably improves Precision, lowers the false positive rate, and outperforms other

approaches. The issue is that the intrusion detection model is not near to the actual CAN bus security protection.

TABLE I. CHALLENGES OF THE EXISTING WORKS

Sl. No	Author Name	Method	Advantages	Disadvantages
1	Rabbani *et al.* [21]	PSO-PNN	The advantage is that it is promising for security checking and distinguishing unsafe leads.	Premature convergence susceptibility, weak local optimization skills
2	Thilagam *et al.* [22]	Recurrent Convolutional Neural Network(RC-NN)	Each pattern can be presumed to be dependent on earlier ones, thanks to the ability of RNN to simulate a collection of records.	The position and orientation of objects are not encoded.
3	Samriya *et al.* [23]	Hybridization Technique	high sensitivity, exact anatomical localization, and quantification potential.	Low rate of good qualities recombining.
4	Kushwah *et al.* [24]	SaE-ELM	Reduce the training time	incorrect data interpretation
5	Majidian, *et al.*[25]	ANFIS	high capacity for generalization	high cost of calculation.
6	Sattari , *et al.* [26]	Software-defined network (SDN)	It enables engineers to reroute networks instantly.	Cost increase brought on by reconfiguration.
7	Ferrag *et al.* [31]	weight-based ensemble machine learning algorithm (WBELA)	Enhancing Predictive Performance	Growing Complexity

III. SYSTEM MODEL

Identifying DoS threats to improve cloud security is crucial. DoS attacks are intended to overburden cloud servers and networks, rendering them inoperable and preventing authorized users from accessing cloud resources. The detection of DoS assaults in cloud systems has several issues: Large volumes of traffic from DoS assaults might be challenging to discern from regular traffic. It can make it difficult to detect and respond to DoS assaults quickly. DoS assaults can spread over numerous computers and networks, making it challenging to pinpoint their origin. The user initiates a request by sending a specific command or action to access a service, browse a website, or communicate with another user. The request is transmitted over the internet. The request reaches the load balancer, which serves as a central entry point for incoming network traffic. The load balancer analyzes the request and determines the most appropriate target server to handle it based on factors such as server health, availability, or predefined load balancing algorithms. Before the request reaches the target server, it passes through the firewall. The firewall examines the request, checking if it complies with predefined security policies and rules. It blocks unauthorized or potentially malicious traffic, protecting the network from potential threats and attacks. Once the request passes through the firewall, it reaches the target server responsible for processing the request. The server performs the necessary operations based on the user's request, retrieves the requested resource or data, and prepares a response to be sent back to the user. The response generated by the target server passes through the firewall. The firewall ensures the response is secure and free from any malicious content or unauthorized data. The load balancer receives the response from the target server and ensures its integrity and consistency.

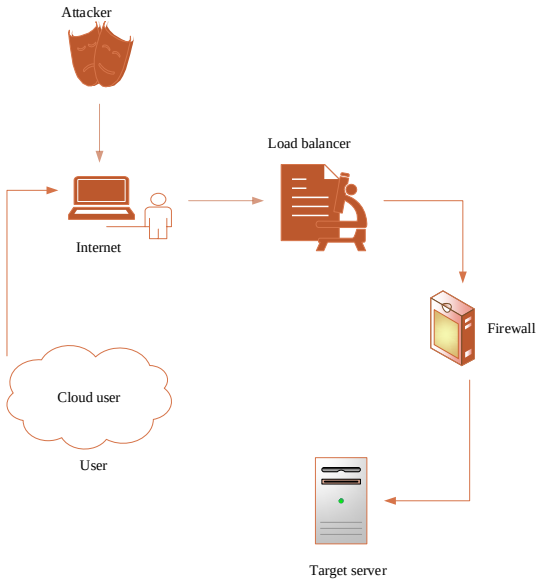

Fig. 2. System model of DOS attack.

It may also perform additional processing or optimization before forwarding the response to the user. The load balancer sends the response back to the user through the internet. The response travels through the network, passing through routers and switches, until it reaches the user's device. The system model is shown in Fig. 2.

Cloud environments are complex and dynamic, with multiple infrastructure, software, and service layers. This complexity can make it challenging to monitor and detect DoS attacks. Monitoring systems may generate false positives, identifying legitimate traffic as malicious and causing unnecessary disturbances in cloud services. So, the presented work has aimed to develop an optimized deep feature-based detecting mechanism for the process.

IV. PROPOSED METHODOLOGY

A novel DCGAN has been developed to detect harmful behaviour in cloud applications. The cloud application containing intrusion data has been purchased and put into the system to evaluate the proposed model. The proposed model employs DCGAN to comprehend network traffic data's

temporal and spatial characteristics, thereby enabling the detection of patterns indicative of DoS assaults. Fig. 3 shows the proposed architecture of DCGAN.

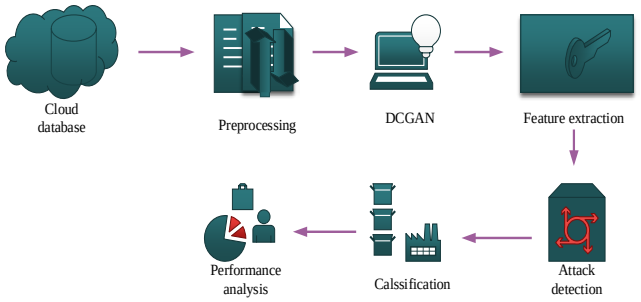

Fig. 3. The proposed DCGAN architecture.

Initially, the datasets are gathered from the authoritative web source. The system was originally trained using the internet application that contained incursion data. Then pre-processing phase is conceded to eliminate the noise and error values. A novel Deep Convolutional Generative Adversarial Network (DCGAN) has been implemented with the necessary characteristics and processing stages. Subsequently, the feature extractions are done here; the unwanted features are removed. More of the wanted features are trained to the attack prediction model to detect the attack and classify the types of attack based on the parts. Finally, the metrics have been validated and compared [32-34] with another system through Accuracy, F1-score, Recall, and Precision.

A. Pre-Processing

Pre-process the collected data by cleaning, normalizing, and transforming it into suitable input formats for training the deep learning model [27]. It may involve feature extraction, data augmentation, or encoding techniques specific to DCGANs. The initialization equation is expressed in Eq. (1).

$$A(v) = (v_1, v_2, v_3, \ldots\ldots v_n)Y \tag{1}$$

Where A denotes the collected cloud dataset from the dataset, v indicates the information present in the dataset, Y denotes the features available in the dataset and n represents the total count of data currently in the database [27]. After initialization, the next stage is the pre-processing phase. Here, the input dataset is pre-handled to eliminate the null values present in the dataset. The pre-processing equation is represented in Eq. (2).

$$H_B^*(A) = \frac{1}{n}\sum_{i=1}^{l}\left\|Y_i - L_i\right\|^2 \tag{2}$$

Where H_B^* denotes the pre-processing variable and L_i indicates the null values present in the dataset. Fitness also plays a role in the decision to virtualize the task without compromising the quality of the work. The GAN fitness function creates new data instances that resemble your training data. The fitness function attains less value for an effective result.

B. Proposed DCGAN

The features are given to the proposed DCGAN-based convolution neural network classifier for detecting DoS attacks after being appropriately chosen. DoS assaults are often found using intrusion detection systems and network monitoring tools, which can spot changes in network traffic that might indicate a DoS attack. The optimized via backpropagation and stochastic gradient descent model to improve the gap between the simulated and observed traffic volumes [28] [33]. Machine learning models, such as DCGANs, may help these detection efforts by examining network traffic patterns and spotting potential threats. The proposed DCGAN is expressed in Eq. (3).

$$lowighM(J,R) = E^*_{ndata(Y)}[log\, J(x^*)] + E^*_{qc(C)}\left[log\left(1 - J(N(C))\right)\right] \tag{3}$$

Where x^* denotes the dataset, c is the generator's blare at the input of the attacks, $n\,data$ is the data delivery, qc is the noise delivery, N is the maker of the cloud attacks and J is the discriminator of the DOS attacks.

C. Feature Extraction

The constructed model then moves on to feature extraction after the suggested technique. The valuable characteristics are extracted, and the unnecessary features are removed in the feature extraction module. Consider Y_i^* as the feature present in the pre-processed dataset [27]. The pre-processed dataset contains relevant features F_i and irrelevant features $F_i^{'}$. The feature extraction equation is expressed in Eq. (4).

$$G_B^*(A) = \rho \times Y_i^*\left[F_i - F_i^{'}\right] \tag{4}$$

Only pertinent characteristics remain in the dataset after the feature extraction stage. The dataset is then trained to recognize assaults using the suggested methodology.

D. Attack Detection and Classification

To identify known and unknown assaults, deep convolution neural networks are employed to detect noise and other undesirable aspects in the data. The neural network is used to reduce the dimensionality of the data during the training phase. The threshold for conventional traffic data is manually selected, despite the suggested approach lessening the data complexity. Given that it does not clearly show the network abnormality or the sort of attack, the model may be able to identify the device connected to the anomaly, making it suitable for fault detection. An assault that is known to have occurred is indicated in the dataset. Here, the system is trained to recognize attacks in the dataset using an attack trained on a massive collection of network traffic data. Eq. (5) [27] gives the equation for attack detection.

$$P_B^l(A) = \begin{cases} S_Q = 1; & Normal \\ S_Q = 0; & Attack \end{cases} \tag{5}$$

Where P_B^I it indicates the attack detection function and S_Q denotes the trained features, attack features are marked as "0" in this instance, while the standard element in the dataset is designated as "1". As a result, the assaults in the dataset are found [27]. Additionally, to see if the system recognizes and categorizes unidentified attacks like Denial of Service (DoS) launched into the system. The unknown attack detection is expressed in Eqn. (6).

$$P_B^{Ul}(A) = \begin{cases} if\ (A_{on} < 80\,Mbs) & ;\ Normal \\ else(A_{on} > 80Mbs); & Attack \end{cases} \quad (6)$$

Where P_B^{Ul} denotes the unknown attack detection function, A_{on} indicates overloading data rate. Based on the speed of data overloading, the standard and attack features are identified here. When data overloading rates are above 80Mbs, an assault is recognized. If the rate of data overloading is under 80Mbs, it is recognized as a typical feature [27]. Attack classification is completed after attack detection. The attack classification is expressed in Eq. (7).

$$P_D^{Ul}(A) = \begin{cases} Other & ;\ 80Mbs > A_{on} < 100Mbs \\ DoS; & A_{on} \geq 100Mbs \end{cases} \quad (7)$$

Where P_D^{Ul} denotes the attack type classification function. The other attack's data overloading rate range is between 80Mbs and 100Mbs. Similarly, if the data overloading rate is more than 100 MB, the attack is classified as a DoS attack. The workflow of the DCGAN approach is shown in Fig. 4.

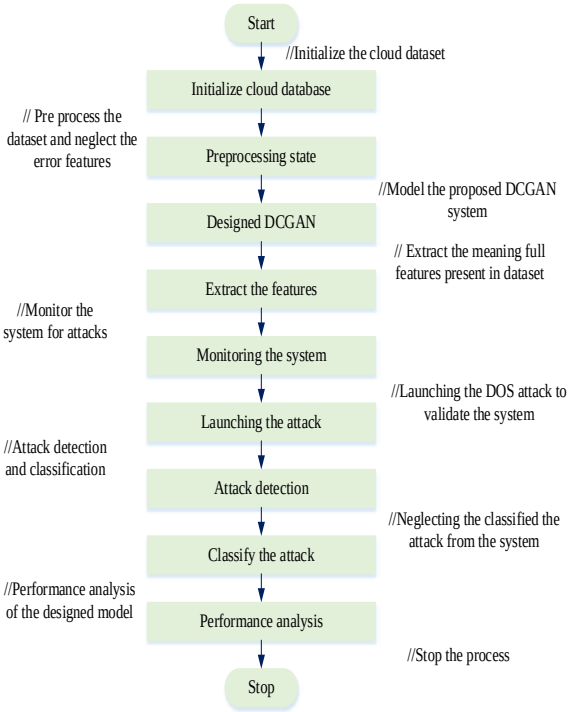

Fig. 4. Flowchart of the proposed DCGAN.

V. Result and Discussion

The presented method is implemented on the PYTHON platform using an Intel I Core I i5 CPU and 8GB RAM. This section discusses the experimental design and the efficacy of the proposed approach. Several measures, including Accuracy, Precision, Recall, and F-measure with the help of optimization and neural networks, are used to assess the organization's effectiveness.

Case Study: This case study discusses a deep convolution GAN network for DOS attacks to enhance cloud security. The cloud application with intrusion data has been bought and installed in the system. The noise and error values are then removed during the pre-processing step. The proposed system's cloud datasets are gathered from an authoritative web source. Then, cleanse, normalize and convert the obtained data into appropriate input formats before using it to train the deep learning model after carefully selecting the suggested DCGAN-based convolution neural network classifier to detect DoS attacks. And then are the feature extractions carried out; the undesirable elements are eliminated. Additionally, the desired features are trained to help the attack prediction model identify and categorize attacks based on their characteristics. Fig. 5 shows the overall process of GAN.0.

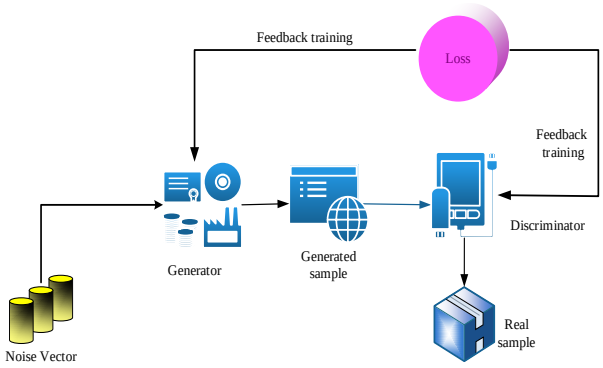

Fig. 5. The proposed GAN process.

A. Performances Metrics

The performance metrics are evaluated in terms of Accuracy, Recall, F-measure, and Precision to validate our proposed DCGAN model.

1) Accuracy measure: According to this definition, accuracy is the proportion of outcomes accurately predicted to all the predictions. The machine learning algorithms' performance parameter is most frequently employed [31] [35]. Accuracy is technically defined as the proportion of accurate positive to accurate negative discoveries needed to finish the machine learning model's outputs. Accuracy is expressed in equation (8),

$$A = \frac{TP' + TN'}{TP' + TN' + FP' + FN'} \quad (8)$$

Where, A indicates the accuracy, $TP^{'}$ denoted as true positive, $TN^{'}$ is genuinely harmful, $FP^{'}$ indicates the false positive, $FN^{'}$ is false negative.

2) Precision calculate: By dividing the total of true positives by all optimistic forecasts, Precision can be used to assess whether the model's optimistic predictions are accurate [31]. Precision is expressed in Eq. (9),

$$P = \frac{TP^{*}}{TP^{*} + FP^{*}} \quad (9)$$

3) Recall measure: The Recall is the percentage of positives the model identified adequately out of all potential positives by dividing true positives by the total number of actual positives [31][36]. Recall is expressed in equation (10),

$$R = \frac{TP^{*}}{TP^{*} + FN^{*}} \quad (10)$$

4) F-measure: F-measure represents a weighted average of Recall and Accuracy. The F-measure considers both positive and negative results to keep the balance between Recall and Precision [31][37]. F-measure is expressed in equation (11),

$$\boldsymbol{F - measure} = \frac{\boldsymbol{2(P+R)}}{\boldsymbol{P+R}} \quad (11)$$

Where P represents the Precision, R represents the Recall.

B. Evaluation of Performance

Comparing the constructed model's metrics for Accuracy, Precision, F-measure, and recall to those of other models confirmed its efficacy. The developed model will be implemented in the Python framework. Moreover, the existing techniques like You Only Look Once a Multilayer Perceptron (MLP) [29], Convolutional Neural Network (CNN) [29], Decision Tree (DT) [30], and Support Vector Machine (SVM) [30].

1) Comparison of the suggested with other existing techniques in terms of accuracy

Fig. 6 shows the accuracy of the suggested DCGAN is compared to that of the previous study. The proposed DCGAN accuracy rate is 0.997. While comparing the suggested DCGAN to other existing methods, the accuracy level is more significant. The accuracy rate for the DT method currently in use is 0.86. At the same time, the accuracy level is high compared to other existing methods like MLP, CNN, and SVM.

Compared to other approaches like SVM and CNN, the MLP accuracy level is 0.987, which is a great value. CNN's accuracy level is 0.986, which is poor compared to all other currently used methods. SVM accuracy level is 0.92 compared to other techniques like MLP, CNN, and DT. As a result, the proposed DCGAN performs superior to the other methods already in use.

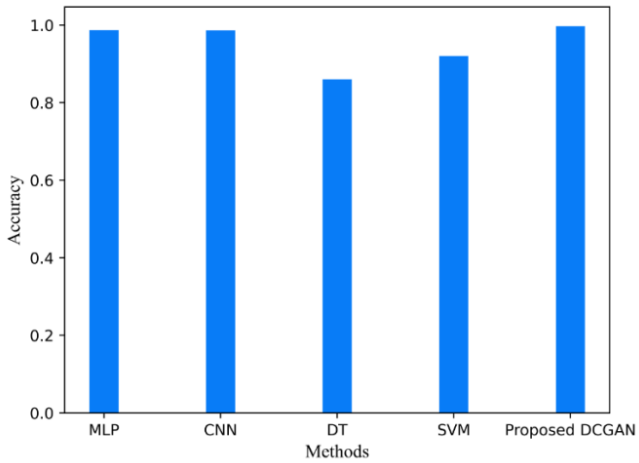

Fig. 6. Comparison of the proposed DCGAN in terms of accuracy.

2) Comparison of the suggested with other existing techniques in terms of Precision

In Fig. 7, the Precision of the suggested DCGAN is compared to that of the previous study. The proposed Convolution-based Buffalo Optimization (DCGAN) precision rate is 0.988. While comparing the suggested DCGAN to other existing methods, the precision level is more significant. The precision rate for the SVM method currently in use is 0.782. At the same time, the precision level is high compared to other existing methods like MLP, CNN, and SVM.

Compared to other approaches like CNN, DT and the MLP precision level is 0.968, which is a great value. The CNN precision level is 0.959, which is poor compared to all other methods currently in use. DT precision level is 0.9159 compared to other techniques like MLP, CNN, and DT. As a result, the proposed DCGAN performs superior to the other methods already in use.

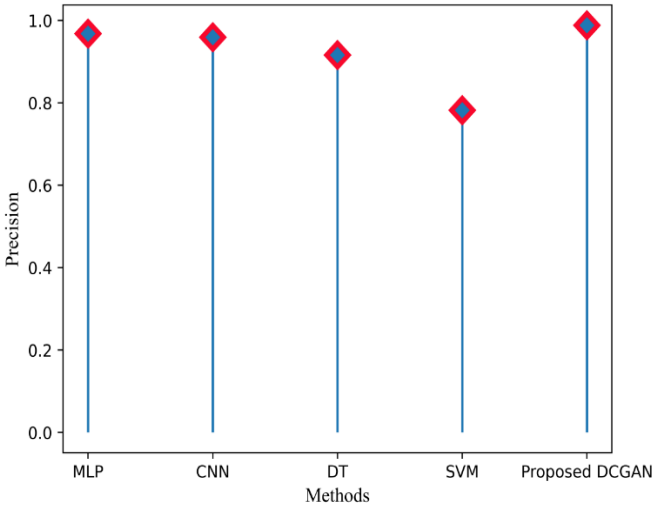

Fig. 7. Comparison of the proposed DCGAN in terms of precision.

3) Comparison of the suggested with other existing techniques in terms of F1-score

In Fig. 8, the F1-score of the suggested DCGAN is compared to that of the previous study. The proposed Convolution-based Buffalo Optimization (DCGAN) F1-score rate is 0.978. While comparing the suggested DCGAN to other existing methods, the F1-score level is more significant. The F1-score rate for the DT method currently in use is 0.7239. At the same time, the F1-score level is high compared to other methods like MLP, CNN, and SVM.

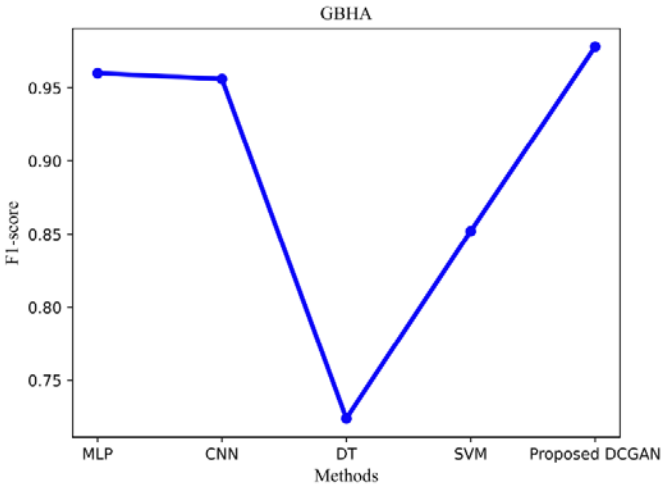

Fig. 8. Comparison of the proposed DCGAN in terms of F1-score.

Compared to other approaches like CNN, DT and the MLP F1-score level is 0.968, which is a great value. The CNN precision level is 0.956, which is poor compared to all other methods currently in use. SVM precision level is 0.852 compared to other techniques like MLP, CNN, and DT. As a result, the proposed DCGAN performs superior to the other methods already in use. The comparison of techniques in terms of accuracy, precision, recall and F1- score data's are mentioned in Table II.

TABLE II. COMPARISON OF TECHNIQUES

Technique	Accuracy	Precision	Recall	F1-score
MLP	0.987	0.968	0.953	0.96
CNN	0.986	0.959	0.954	0.956
DT	0.86	0.9159	0.6808	0.7239
SVM	0.92	0.782	0.936	0.852
Proposed DCGAN	0.997	0.988	0.96	0.978

4) Comparison of the suggested method with other existing techniques in terms of Recall

In Fig. 9, the Recall of the suggested DCGAN is compared to that of the previous study. The proposed Convolution-based Buffalo Optimization (DCGAN) recall rate is 0.96. While comparing the suggested DCGAN to other existing methods, the F1-score level is more significant. The recall rate for the DT method currently in use is 0.6808. At the same time, the recall level is high compared to other existing methods like MLP, CNN, and SVM.

Compared to other approaches like MLP DT, the CNN recall level is 0.954, which is a great value. MLP recall level is 0.953, which is poor compared to all other currently used methods. The SVM recall level is 0.936 when compared to other techniques like MLP, CNN, and DT. As a result, the proposed DCGAN performs superior to the other methods already in use.

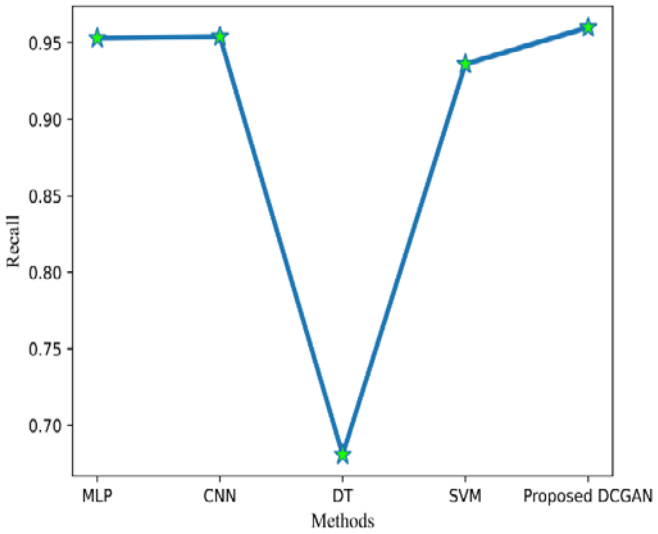

Fig. 9. Comparison of the proposed DCGAN in terms of recall.

Comparative analysis with other machine learning approaches, such as MLP, CNN, DT, and SVM, reveals the superior performance of the DCGAN model in terms of precision, recall, and F1-score. This comparative investigation showcases the advancement of the proposed model over existing techniques and highlights its potential for practical deployment in real-world cloud environments. The DCGAN model's improved precision, robust pattern recognition capabilities and superior performance in comparison to other methods contribute to the enhancement of cloud security measures and offer promising prospects for further advancements in the field.

VI. DISCUSSION

The presented IDS approach was designed and verified in the Python tool. The essential goal of the created model is to distinguish pernicious occasions in the WS organization. The neural network in the proposed approach improves the detection rate and accuracy.

Table III summarizes the performance study of the provided model. For the DCGAN dataset, the designed model attained 0.997 accuracies and 0.988 precision, a recall of 0.96, and an F-score of 0.978 in the developed model. Additionally, findings are compared to current approaches to confirm the great Precision in the calculation period of the developed model. The findings highlight the significance of the DCGAN model in enhancing cloud security by accurately detecting DoS attacks. The improved precision and reduced false positive rate

contribute to better protection against service disruptions, data breaches, and financial losses. The study demonstrates the potential of deep learning approaches, specifically DCGANs, in tackling complex security challenges in cloud environments. The results provide valuable insights for researchers and practitioners working on developing effective mechanisms for detecting and mitigating DoS attacks. By addressing the limitations and pursuing future research avenues, the study contributes to advancing the knowledge and practical applications of cloud security.

TABLE III. PERFORMANCE ANALYSIS

Metrics	***Performance***
F-score	0.978
Recall	0.96
Precision	0.988
Accuracy	0.997

The article introduces a novel approach using a DCGAN as an optimization-based deep learning model for identifying DoS attacks in the cloud. This represents a significant contribution to the field by exploring the effectiveness of DCGANs in detecting such attacks. The proposed model leverages DCGAN to comprehend the spatial and temporal features of network traffic data. By analyzing these features, the model can identify patterns indicative of DoS attacks, thereby enhancing the accuracy of detection. To improve the accuracy and robustness of the DCGAN model, it is trained on a large collection of network traffic data. This approach enables the model to learn from diverse attack scenarios and enhances its ability to detect previously unseen attacks. The model is optimized through back propagation and stochastic gradient descent to minimize the loss function, which quantifies the discrepancy between simulated and observed traffic volumes. This optimization process enhances the model's performance and fine-tunes its ability to detect DoS attacks effectively. The article provides testing findings that demonstrate the superiority of the proposed DCGAN model over recent technology methods in terms of precision and the rate of false positives. This comparative evaluation contributes valuable insights into the effectiveness of the DCGAN model for DoS attack detection in cloud environments.

VII. CONCLUSION

By effectively identifying DoS attacks, the optimization-based deep learning model, which utilizes techniques like DCGANs, has the potential to enhance cloud security. The model leverages deep learning to detect anomalies and learn intricate patterns within network traffic data. In this study, the model's performance is evaluated in terms of Recall, Precision, and F1 score for DoS attack detection. A controlled simulation environment is employed to test the model's efficacy in detecting DoS attacks. The obtained results are then compared with existing machine learning approaches such as MLP, CNN, DT, and SVM. The comparative analysis reveals that the developed DCGAN model surpasses other models in terms of performance. Notably, the Accuracy, Precision, Recall, and F1-score values exhibit significant improvements in the proposed DCGAN technique, achieving enhancements of 0.997, 0.988, 0.96, and 0.978, respectively. These results indicate the superiority of the DCGAN model in accurately identifying DoS attacks compared to alternative approaches. Furthermore, in addition to the empirical findings, this study contributes to the field by introducing newly formulated theoretical advancements. The utilization of DCGANs for DoS attack detection in cloud environments represents a novel approach with potential implications for enhancing cloud security. The incorporation of deep learning techniques and the specific architecture of DCGANs contribute to the model's ability to learn complex patterns and identify subtle anomalies in network traffic data. The study may have been limited by the availability of a specific dataset for training and testing the DCGAN model. The results may vary when applied to different datasets or real-world scenarios. The focus of the study was on detecting DoS attacks. The model's performance in detecting other types of attacks or more complex attack patterns was not explored. The study primarily focused on the performance evaluation of the DCGAN model in a controlled environment. The real-time implementation and assessment of the model's effectiveness in practical cloud environments were not addressed. The study thus offers theoretical insights into the application of DCGANs for cloud security and lays the foundation for further research in this area.

VIII. FUTURE WORK

In future it would be seeked to enhance the DCGAN model's ability to detect more sophisticated and evolving DoS attack techniques. This involves exploring novel network traffic features, developing more robust anomaly detection algorithms, and incorporating machine learning techniques to improve the accuracy and effectiveness of the model. Also by integrating the DCGAN model with existing cloud security infrastructure, such as intrusion detection systems, firewalls, or security incident response platforms. This integration can enable a comprehensive security ecosystem, leveraging the strengths of multiple security mechanisms to provide a more holistic and proactive defense against DoS attacks.

REFERENCES

[1] Yaacoub, J. P. A., Noura, M., Noura, H. N., Salman, O., Yaacoub, E., Couturier, R., & Chehab, A. (2020). Securing internet of medical things systems: Limitations, issues and recommendations. *Future Generation Computer Systems*, *105*, 581-606.

[2] ur Rehman, S., Khaliq, M., Imtiaz, S. I., Rasool, A., Shafiq, M., Javed, A. R., ... & Bashir, A. K. (2021). DIDDOS: An approach for detection and identification of Distributed Denial of Service (DDoS) cyberattacks using Gated Recurrent Units (GRU). Future Generation Computer Systems, 118, 453-466.

[3] Tayfour, O. E., & Marsono, M. N. (2021). Collaborative detection and mitigation of DDoS in software-defined networks. The Journal of Supercomputing, 77, 13166-13190.

[4] Mihoub, A., Fredj, O. B., Cheikhrouhou, O., Derhab, A., & Krichen, M. (2022). Denial of service attack detection and mitigation for internet of things using looking-back-enabled machine learning techniques. Computers & Electrical Engineering, 98, 107716.

[5] Skopik, Florian, Markus Wurzenberger, and Max Landauer. "Detecting Unknown Cyber Security Attacks Through System Behavior Analysis." Cybersecurity of Digital Service Chains: Challenges, Methodologies, and Tools. Cham: Springer International Publishing, 2022. 103-119.

[6] Poria, S., Majumder, N., Mihalcea, R., & Hovy, E. (2019). Emotion recognition in conversation: Research challenges, datasets, and recent advances. IEEE Access, 7, 100943-100953.

[7] Kunang, Y. N., Nurmaini, S., Stiawan, D., & Suprapto, B. Y. (2021). Attack classification of an intrusion detection system using deep learning and hyperparameter optimization. Journal of Information Security and Applications, 58, 102804.

[8] Smmarwar, S. K., Gupta, G. P., Kumar, S., & Kumar, P. (2022). An optimized and efficient android malware detection framework for future sustainable computing. Sustainable Energy Technologies and Assessments, 54, 102852.

[9] Zhang, Z., Al Hamadi, H., Damiani, E., Yeun, C. Y., & Taher, F. (2022). Explainable artificial intelligence applications in cyber security: State-of-the-art in research. IEEE Access.

[10] Bhayo, J., Jafaq, R., Ahmed, A., Hameed, S., & Shah, S. A. (2021). A time-efficient approach toward DDoS attack detection in IoT network using SDN. IEEE Internet of Things Journal, 9(5), 3612-3630.

[11] Rejeb, A., Rejeb, K., Treiblmaier, H., Appolloni, A., Alghamdi, S., Alhasawi, Y., & Iranmanesh, M. (2023). The Internet of Things (IoT) in healthcare: Taking stock and moving forward. Internet of Things, 100721.

[12] Ghobadpour, A., Boulon, L., Mousazadeh, H., Malvajerdi, A. S., & Rafiee, S. (2019). State of the art of autonomous agricultural off-road vehicles driven by renewable energy systems. Energy Procedia, 162, 4-13.

[13] Kumar, R., Kumar, P., Tripathi, R., Gupta, G. P., Garg, S., & Hassan, M. M. (2022). A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network. Journal of Parallel and Distributed Computing, 164, 55-68.

[14] Labayen, V., Magaña, E., Morató, D., & Izal, M. (2020). Online classification of user activities using machine learning on network traffic. Computer Networks, 181, 107557.

[15] Shamshirband, S., Fathi, M., Chronopoulos, A. T., Montieri, A., Palumbo, F., & Pescapè, A. (2020). Computational intelligence intrusion detection techniques in mobile cloud computing environments: Review, taxonomy, and open research issues. Journal of Information Security and Applications, 55, 102582.

[16] Rathee, G., Iqbal, R., Waqar, O., & Bashir, A. K. (2021). On the design and implementation of a blockchain enabled e-voting application within iot-oriented smart cities. IEEE Access, 9, 34165-34176.

[17] Koppu, S., Maddikunta, P. K. R., & Srivastava, G. (2020). Deep learning disease prediction model for use with intelligent robots. Computers & Electrical Engineering, 87, 106765.

[18] Rabbani, M., Wang, Y. L., Khoshkangini, R., Jelodar, H., Zhao, R., & Hu, P. (2020). A hybrid machine learning approach for malicious behaviour detection and recognition in cloud computing. Journal of Network and Computer Applications, 151, 102507.

[19] Thilagam, T., & Aruna, R. (2021). Intrusion detection for network based cloud computing by custom RC-NN and optimization. ICT Express, 7(4), 512-520.

[20] Latif, S., Zou, Z., Idrees, Z., & Ahmad, J. (2020). A novel attack detection scheme for the industrial internet of things using a lightweight random neural network. IEEE Access, 8, 89337-89350.

[21] Rabbani, Mahdi, Yong Li Wang, Reza Khoshkangini, Hamed Jelodar, Ruxin Zhao, and Peng Hu. "A hybrid machine learning approach for malicious behaviour detection and recognition in cloud computing." Journal of Network and Computer Applications 151 (2020): 102507.

[22] Thilagam, T., & Aruna, R. (2021). Intrusion detection for network based cloud computing by custom RC-NN and optimization. ICT Express, 7(4), 512-520.

[23] Samriya, Jitendra Kumar, and Narander Kumar. "A novel intrusion detection system using hybrid clustering-optimization approach in cloud computing." Materials Today: Proceedings (2020).

[24] Kushwah, Gopal Singh, and Virender Ranga. "Optimized extreme learning machine for detecting DDoS attacks in cloud computing." Computers & Security 105 (2021): 102260.

[25] Majidian, Z., TaghipourEivazi, S., Arasteh, B., & Babai, S. (2023). An intrusion detection method to detect denial of service attacks using error-correcting output codes and adaptive neuro-fuzzy inference. Computers and Electrical Engineering, 106, 108600.

[26] Sattari, F., Farooqi, A. H., Qadir, Z., Raza, B., Nazari, H., & Almutiry, M. (2022). A Hybrid Deep Learning Approach for Bottleneck Detection in IoT. IEEE Access, 10, 77039-77053.

[27] Velliangiri, S., Karthikeyan, P. and Vinoth Kumar, V., 2021. Detection of distributed denial of service attack in cloud computing using the optimization-based deep networks. Journal of Experimental & Theoretical Artificial Intelligence, 33(3), pp.405-424.

[28] Shieh, C.S., Nguyen, T.T., Lin, W.W., Lai, W.K., Horng, M.F. and Miu, D., 2022. Detection of Adversarial DDoS Attacks Using Symmetric Defense Generative Adversarial Networks. Electronics, 11(13), p.1977.

[29] Zhang, Chaoyun, Xavier Costa-Perez, and Paul Patras. "Adversarial attacks against deep learning-based network intrusion detection systems and defense mechanisms." IEEE/ACMTransactions on Networking 30, no. 3 (2022): 1294-1311

[30] Al-Abassi, Abdulrahman, Hadis Karimipour, Ali Dehghantanha, and Reza M. Parizi. "An ensemble deep learning-based cyber-attack detection in industrial control system." IEEE Access 8 (2020): 83965-83973.

[31] SaiSindhuTheja, Reddy, and Gopal K. Shyam. "An efficient metaheuristic algorithm based feature selection and recurrent neural network for DoS attack detection in cloud computing environment." Applied Soft Computing 100 (2021): 106997.

[32] Almalawi, A., Khan, A.I., Alsolami, F., Abushark, Y.B. and Alfakeeh, A.S., 2023. Managing Security of Healthcare Data for a Modern Healthcare System. Sensors, 23(7), p.3612.

[33] Alloqmani, A., Abushark, Y.B., Khan, A.I. and Alsolami, F., 2021. Deep learning based anomaly detection in images: insights, challenges and recommendations. International Journal of Advanced Computer Science and Applications, 12(4).

[34] Ferrag, Mohamed Amine, Othmane Friha, Leandros Maglaras, Helge Janicke, and Lei Shu. "Federated deep learning for cyber security in the internet of things: Concepts, applications, and experimental analysis." IEEE Access 9 (2021): 138509-138542.

[35] Sarker, I.H., Abushark, Y.B., Alsolami, F. and Khan, A.I., 2020. Intrudtree: a machine learning based cyber security intrusion detection model. Symmetry, 12(5), p.754.

[36] Almalawi, A., Khan, A.I., Alsolami, F., Abushark, Y.B., Alfakeeh, A.S. and Mekuriyaw, W.D., 2022. Analysis of the Exploration of Security and Privacy for Healthcare Management Using Artificial Intelligence: Saudi Hospitals. Computational Intelligence & Neuroscience.

[37] Alzaidi, B.S., Abushark, Y. and Khan, A.I., 2022. Arabic Location Named Entity Recognition for Tweets using a Deep Learning Approach. International Journal of Advanced Computer Science and Applications, 13(12).