

## Three level intrusion detection system based on conditional generative adversarial network

Hasan Abdulameer<sup>1</sup>, Inam Musa<sup>2</sup>, Noora Salim Al-Sultani<sup>3</sup>

<sup>1</sup>Department of Medical Instrumentation Techniques Engineering, Al-Hussain University College, Karbala, Iraq

<sup>2</sup>Department of Electrical Power Techniques Engineering, Al-Hussain University College, Karbala, Iraq

<sup>3</sup>Department of Water Resources Management Engineering, College of Engineering, Al-Qasim Green University, Babylon, Iraq

### Article Info

#### Article history:

Received Mar 23, 2022

Revised Sep 20, 2022

Accepted Oct 15, 2022

#### Keywords:

Conditional generative

adversarial network

Firewall

Intrusion detection system

Proximal policy optimization

### ABSTRACT

Security threat protection is important in the internet of things (IoT) applications since both the connected device and the captured data can be hacked or hijacked or both at the same time. To tackle the above-mentioned problem, we proposed three-level intrusion detection system conditional generative adversarial network (3LIDS-CGAN) model which includes four phases such as first-level intrusion detection system (IDS), second-level IDS, third-level IDS, and attack type classification. In first-level IDS, features of the incoming packets are extracted by the firewall. Based on the extracted features the packets are classified into three classes such as normal, malicious, and suspicious using support vector machine and golden eagle optimization. Suspicious packets are forwarded to the second-level IDS which classified the suspicious packets as normal or malicious. Here, signature-based intrusions are detected using attack history information, and anomaly-based intrusions are detected using event-based semantic mapping. In third-level IDS, adversary packets are detected using CGAN which automatically learns the adversarial environment and detects adversary packets accurately. Finally, proximal policy optimization is proposed to detect the attack type. Experiments are conducted using the NS-3.26 network simulator and performance is evaluated by various performance metrics which results that the proposed 3LIDS-CGAN model outperforming other existing works.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



### Corresponding Author:

Hasan Abdulameer

Department of Medical Instrumentation Techniques Engineering, Al-Hussain University College

Karbala, Iraq

Email: hasanabdulameer@huciraq.edu.iq

## 1. INTRODUCTION

In internet of things (IoT) environment, an enormous amount of data containing various types of information (sensitive and non-sensitive) are collected from the devices and transmitted through internet [1] There are many chances of compromising the information during data transmission by attackers due to lack in security and data encryption. Intrusion detection is one of the main techniques implemented to detect attacks. In recent days all devices are converted into smart devices, which means transferring data over a network without requiring human-to-human or human-to-computer interaction. Instead, all things are connected to the internet and performed the task via the Internet. Therefore, many transactions are performed at a particular time because of huge environment which increases network traffic [2]. At the same time, attackers are also present in the network traffic to hack the data. Hence intrusion detection system (IDS) is important in IoT environment [3]. The intrusion detection system is a system that monitors network traffic is normal or malicious. In IDS, both detections detect and analyze all the inbound and outbound network traffic

and prevention by proactively inspecting a system's incoming traffic to weed out malicious requests [4]. However, some researchers are only focused on detection, not on prevention, which leads to unable to prevent attacks that use a preexisting database for signature recognition based on traffic and behavioral anomalies [5]. An ID has two types such as signature-based intrusion detection (SIDS) and anomaly-based intrusion detection (AIDS). SIDS is used to detect known attacks such as structured query language (SQL), which can easily detect because these patterns already exist in system whereas AIDS is used to detect unknown attacks and these attacks are difficult to detect [6], [7].

Firewalls protect against outside cyber attackers by shielding your computer or network from malicious software or unnecessary network traffic. However, the firewall is an important component in IDS which is used to filter the network packets in an earlier stage. It classifies the network packets into normal or malicious [8]. It can be configured to allow necessary and relevant data through the network or computers and block malicious packets in an earlier stage. Moreover, this process helps to reduce the complexity and also increase the detection rate [9]. IDS has adversarial intrusion; detection of adversarial intrusions is difficult because they look like normal packets [10]. Adversarial learning is used for detecting these types of intrusions which learning adversarial environments and detecting adversarial intrusions in an accurate manner [11]. Several types of attacks are present in network traffic such as: i) distributed denial of service (DDoS), ii) phishing, iii) SQL injection, iv) man in the middle, and v) false data injection

These attacks are mitigated by using IDS and giving an alarm to every node present in a network which helps to prevent other nodes from intrusions [12]. The concept of human immunity against pathological diseases was leveraged to design a two-layer IDS in which the B-cells and T-cells were considered [13]. The detection accuracy of several deep neural networks (DNN) was compared to find that the neural network structure had better accuracy in detecting anomalies [14]. The reinforcement learning-based IDS provided resistance to many newly generated malware in the network thereby ensuring the security of resource constraint IoT environment [15], [16]. The complexity of the IDS is considerably reduced by performing filtering of packets and reducing of dimensionality of huge data [17].

Several rule-based anomaly detection techniques were implemented for precise detection of anomalies in the network, but these approaches classified only known attacks [18], [19]. The evaluation of several machine learning and deep learning algorithms against conventional intrusion detection algorithms was carried out to conclude the efficacy of those algorithms but those algorithms were effective only for particular types of attacks [20]–[25]. The execution of hybrid-based IDS was found to be more efficient in detecting the anomalies in the network [26].

In our paper, we propose hybrid intrusion detection of both signatures-based and anomaly-based attacks in IoT environments. The major problem statements encountered in detection of intrusion based on hybrid methods are described in this section. The existing hybrid intrusion detection models used support vector machine based (SVM) packet classification for accurate detection of malicious packets. The combination of C5 decision tree classifier and one class SVM was used for detection of both signatures-based and anomaly-based intrusions [27]. In [28] preprocessing is performed by implementing normalization of the data. Further, K means clustering algorithm is used to cluster the network traffic in terms of normal, denial of service (DoS), and Probe data, and SVM is utilized to classify the attacks in terms of normal, Dos, or Probe. Here, anomalies are detected by using hybridization of machine learning algorithms (C5 decision tree, one-class SVM) to give better results but SVM takes much time to select kernel which increases latency. K means clustering algorithm is used for clustering the network traffic, but it was not suitable for global optimal solution. The number of clusters should be defined initially which is not appropriate thereby reducing the performance of the process. In order to overcome these problems our proposed system uses SVM with golden eagle optimization which is used to select an optimal kernel.

This paper is further organized into several sections which are as follows: section 2 discusses the current research works on IDS in IoT. This section describes limitations of each work. Section 3 provides the most significant problems briefly faced by the hybrid IDS. Section 4 describes the methodology of the proposed three-level intrusion detection system conditional generative adversarial network (3LIDS-CGAN) model with the implementation of 3LIDS-CGAN model with the simulation setup. In this section, the validation of proposed work is carried out in terms of several performance metrics. Section 5 concludes the 3LIDS-CGAN model with future work in an elaborative manner.

## 2. RELATED WORKS

This section describes the literature on the previous state-of-the-art methods related to the proposed 3LIDS-CGAN model. Intrusion detection with hybrid sampling using deep hierarchical network was proposed in [29]. The main objective of the proposed system is to imbalance the network traffic data. The proposed system used hybrid method which includes one side selection (OSS) method that removes the noise for reducing the majority samples and SMOTE algorithm is used to increase the minority samples. Data

preprocessing is done by using deep hierarchical network. Classification is done by using hierarchical network which includes convolution neural network (CNN) and Belts. The performance of the proposed system is evaluated by using NSL-KDD and UNSW-NB15 datasets.

A lightweight intrusion detection system for IoT environment was proposed in [30]. The proposed system has two sections a training section and evaluation section. The features are extracted from the network traffic. After feature extraction, classification is done by using SVM classifier which classifies the traffic into normal or intrusion. Hybrid neural network-based anomaly detection was proposed in [31]. The proposed system has five phases, first phase is flow mapping and processing which is used to detect whether the flow-id timeout or not. Second phase is sequence packet features (SPF) of single processing which extracts the features from the network flow. Third phase is used to extract the general statistical features (GSF) of single flow processing. Fourth phase is performed to extract the environmental features of the flow processing which extracts the features from the active flows in the flow sliding window.

An ensemble-based intrusion detection system for IoT environment was proposed in [32]. The proposed system performs hybrid intrusion detection which includes both signatures-based and anomaly-based intrusion detection with the help of C5 and one-class SVM. The proposed system aims to detect both well-known and zero-day attacks.

A hybrid intrusion detection using principal component analysis-grey wolf optimization (PCA-GWO) and DNN which is suitable for IoT environments was proposed in [33]. The proposed system performs preprocessing by using one-hot encoding which converts all values into numerical. After that, normalization is proposed for converting the data within the range of zero to one. PCA and GWO algorithms are used to reduce the dimensionality of the dataset.

Atefi *et al.* [34] proposed an ensemble-based modified adaptive boosting algorithm to detect network intrusions that have two types such as M-Adaboost-A-SMV and M-Adaboost-A-PSO. This proposed work aimed to solve the imbalance in the network intrusions detection which covers the area of boosting process.

Wireless intrusion detection by using improved convolution neural network (ICNN) was proposed in [35]. The proposed system has three processes such as preprocessing and normalization, ICNN training, and classification. Preprocessing is used to convert the data into numerical value and the processed numerical value is mapped to the feature range which is known as normalization that has a standard value.

Intrusion detection by using improved genetic algorithm (GA) and deep belief network for IoT environment was proposed in [36]. GA is used to detect optimal solutions and DBN is used to classify the network attacks. An optimization-based hybrid IDS was proposed by Rose *et al.* [37]. The binary grey wolf optimization algorithm (BGWO) was combined with statistical algorithms like naïve Bayes (NB) to perform optimal detection of intrusions in the IoT network.

A novel intrusion detection technique implementing both multi-objective genetic algorithm (NSGA-II) and artificial neural network (ANN) along with decision tree-based random forest classifier for effective detection of anomalies in the network was proposed in [38]. The activity of the log data is mapped using the algorithm as heuristic miner in [39]. Initially, data collection was performed by extracting the data from the corresponding event log. Secondly, checking process is performed to obtain the matrix casual data's fitness value by heuristic miners. This process is followed by enhancement phase to obtain the placement model.

Kumar and Harikiran [40] proposed an approach to preserve the privacy of the activities by recognizing the actions using a prediction algorithm that comes under deep neural network. This algorithm anonymized the content of video to protect the privacy from various adversaries. Finally, the recognition framework is used to recognize the privacy-preserved actions.

Kim *et al.* [41] performed anomaly detection for the vibration data city train using generative adversarial network. For vibration data analysis, spectral density evaluation was carried out. Train the vibration data using long short-term memory algorithm.

### 3. 3LIDS-CGAN MODEL

Our proposed system focuses to detect both signature and anomaly-based intrusions in an IoT environment. It has four consecutive phases such as: i) first level IDS, ii) second level IDS, iii) third level IDS, and iv) attack type classification. The real-time packets entering the IDS model contain several existing and new attacks which are identified with improved accuracy. The first phase classifies incoming packets into three classes namely normal, suspicious, and malicious from which the malicious packets are dropped and suspicious packets are sent to the second phase in which the signature-based and anomaly-based IDS takes place which results in classification of those packets into normal and malicious from which the malicious packets are dropped. The normal packets from first and second phases are processed in the third phase to detect the adversaries to improve the security of the IoT environment.

**3.1. First level IDS**

In first level IDS, first process is packet flow-based feature extraction. The packet features are extracted by using firewall which filters the incoming packets with the features of packet interval time, packet size, packet type, payload length, and timestamp. The extracted features are classified by using SVM and golden eagle optimization which is used to select the kernel function of the SVM like linear, polynomial, radial basis function (RBF), and sigmoid which has four parameters such as cost, gamma, coefficient, and degree. Intrusions are detected based on the extracted features from the firewall. The SVM is already with the normal patterns. Every new packet is matched to the normal patterns if it varies from the threshold then it is marked as intrusion or attack. In this research, we used multiclass SVM for classification. In SVM hyperplane is used to classify the features into three classes, that hyperplane needs to follow the rule in (1),

$$F(y) = (v, y) + a \tag{1}$$

where  $v$  represents the normal vector and  $a$  represent the bias value and  $y$  represents the test sample. In SVM the intrusion detection is performed by selecting optimal kernel for that we proposed golden eagle optimization which selects the optimal kernel from the four kernels (linear, RBF, sigmoid, and polynomial) of SVM. The classification function of SVM is defined as (2).

$$F(y) = \begin{cases} -1, & \text{if } y \in \text{malicious} \\ 0, & \text{if } y \in \text{suspicious} \\ 1, & \text{if } y \in \text{normal} \end{cases} \tag{2}$$

In next stage of SVM, assume  $y_1, y_2, \dots, y_n$  be a training sample. And then, separate the data from the origin for that we need to solve the quadratic programming problems.

$$\text{Min} \frac{1}{2} \|W\|^2 + \frac{1}{vn} \sum_{i=1}^n E_i - P \tag{3}$$

$$W \times \varphi(y_i) \geq \sigma - E_i \quad i = 1, 2, \dots, n \quad E_i \geq 0 \tag{4}$$

If  $W$  and  $\sigma$  solve the quadratic programming problem, then the decision function will be normal for maximum instances in the training set.

$$F(y) = \text{Sign}((W \times \varphi(y_i)) - \sigma) \tag{5}$$

This research used RBF kernel function which is selected by golden eagle optimization which optimizes the parameters of  $c$  and  $\gamma$ . Every kernel has specific parameters that can be enhanced to get the best performance result which is illustrated in Table 1. SVM identifies the behavior of the normal packets using extracted features. It proposed that SVM classifies the current packets into normal, malicious, or suspicious.

Table 1. Types of kernel functions and their parameters

Function of Kernel	Equation	Parameter
RBF	$k(y_n, y_i) = \exp(-\gamma \ y_n - y_i\ ^2 + c)$	$c$ and $\gamma$
Linear	$k(y_n, y_i) = (y_n, y_i)$	$c$ and $\gamma$
Polynomial	$k(y_n, y_i) = (\gamma (m(y_n, y_i) + s))^b$	$c, \gamma, s$ and $b$
Sigmoid	$k(y_n, y_i) = \tanh(\gamma (y_n, y_i) + s)$	$c, \gamma$ and $s$

Where  $c$  represents cost and  $\gamma$  denotes gamma and  $s$  represents coefficient and  $b$  represents degree. For getting the optimal value from the kernel, the search method is performed using the parameters  $c, \gamma, s$  and  $b$ .

The first process of golden eagle optimization is defined as follows. The attack is modeled through a vector beginning from the current position of the golden eagle; the attack vector is calculated as (6):

$$\vec{a}_i = \vec{y}_f^* - \vec{y}_i \tag{6}$$

where,  $\vec{a}_i$  is represent the eagle  $i$  attack vector and  $\vec{y}_f^*$  represent the best location visited by eagle  $f$ , and  $\vec{y}_i$  represent the current location of the eagle  $i$ . Next process is to calculate the cruise vector concerning attack vector. Cruise vector is a tangent vector to the circular and that is positioned perpendicular to the attack vector. The tangent hyperplane is calculated as (7),

$$H_1 y_1 + \dots H_n y_n = D \rightarrow \sum_{j=1}^n H_j y_j = D \quad (7)$$

where  $\vec{h} = [H_1, \dots, H_n]$  represent the normal vector and  $Y = [y_1, \dots, y_n]$  represent the variable vector,  $D = \vec{h} \cdot \vec{S}$ .  $\vec{S}$  represent the arbitrary point. Therefore, the hyperplane is represented as (8),

$$\sum_{j=1}^n A_j y_j = \sum_{j=1}^n A_j^t y_j^* \quad (8)$$

where  $\vec{a}_i = [A_1, \dots, A_n]$  represent the attack vector and  $Y^* = [y_1^*, \dots, y_n^*]$  is represent the location. The new position of the eagle is defined as (9),

$$\Delta y_i = \vec{R}_1 P_A \frac{\vec{a}_i}{\|\vec{a}_i\|} + \vec{R}_2 P_b \frac{\vec{b}_i}{\|\vec{b}_i\|} \quad (9)$$

where,  $P_A^t$  is represent the attack coefficient at iteration  $t$  and  $P_b$  represent the cruise coefficient and  $\vec{R}_1$  and  $\vec{R}_2$  represent the random vectors between the interval of [0,1]. And  $\|\vec{A}_i\|$  and  $\|\vec{b}_i\|$  represent Euclidean norm of attack that is defined as (10).

$$\|\vec{A}_i\| = \sqrt{\sum_{j=1}^n A_j^2} \text{ and } \|\vec{b}_i\| = \sqrt{\sum_{j=1}^n b_j^2} \quad (10)$$

The position of the eagle is calculated as (11).

$$y^{t+1} = y^t + \Delta y_i^t \quad (11)$$

The fitness of new position of eagle  $i$  is better than the current position; hence the new position is updated.

In this algorithm,  $P_A, P_b$  is used to shift from exploration to exploitation. Initially, this algorithm starts with minimum  $P_A$  and maximum  $P_b$ . Starting and finishing parameters are determined by the user and intermediate values are calculated using the linear function, which is defined as (12),

$$\begin{cases} P_A = P_A^0 + \frac{t}{T} |P_A^T - P_A^0| \\ P_b = P_b^0 + \frac{t}{T} |P_b^T - P_b^0| \end{cases} \quad (12)$$

where  $t$  represents the current iteration and  $T$  represents maximum iteration and  $P_b^0$  and  $P_b^T$  represent the initial and final values of attack ( $P_b$ ) and  $P_A^0$  and  $P_A^T$  represent the initial and final values of attack ( $P_A$ ). Finally, the best kernel function is selected using this algorithm. In our work, RBF is selected as the best kernel for classification. Our proposed system performs accurate classification which improves the accuracy of the process. The firewall ignores the malicious packets and forwards suspicious packets into next-level IDS. The pseudo-code for first-level IDS is provided below in which the selection of kernel is described above in an elaborative manner.

#### Pseudocode for support vector machine (SVM) with golden eagle optimization

*INPUT:* Extracted features  $F = \{f_1, f_2, \dots, f_n\}$ , kernel parameters  $c, s, b, \gamma$

*OUTPUT:* Kernel selection

Begin

Initialize  $P_A$  and  $P_b$

Initialize  $c, s, b$  and  $\gamma$

Form SVM by training dataset and initialized position of each attack

Evaluate the fitness function

for each iteration  $t$  do

Update  $P_A$  and  $P_b$  using (12)

for each eagle  $i$  do

Select random prey from the memory of populations

Compute attack vector  $\vec{a}_i$  using (6)

if ( $\vec{a}_i \neq 0$ ) then

Compute cruise vector using (7) and (8)

Compute step vector using (9) and (10)

Position is updated using (11)

Calculate fitness value for new position

if new position is better than current position then

replace new position as the best position (fitness)

```

end if
end if
end for
//(after select optimal kernel)
Compute classification for every  $f_i$  using kernel
end

```

### 3.2. Second level IDS

The suspicious packets are entered into the second level of IDS. In this stage, suspicious packets are classified as normal or malicious. Generally, two types of IDS are presented in network traffic such as signature-based intrusion and anomaly-based intrusion. Signature-based intrusions are detected by using the history of the attacks which are trained and stored in a database. Anomaly-based intrusions are detected by using the event-based semantic mapping which means that the intrusions are mapped with respect to the time series.

This map is constructed by using naive Bayes algorithm and this map has event-based intrusions. The probability of packets having the most likely features is formulated as (13),

$$\hat{a} = \arg \max_a P(a|b) \quad (13)$$

where,  $P(a|b)$  can be expressed as (14),

$$P(a|b) = \frac{P(b|a)P(a)}{P(b)} \propto P(b|a)P(a) \quad (14)$$

where, the likelihood of packet b of class a, can be represented as  $P(b|a)$ . The likelihood is calculated based on the similarity of features possessed by the packets. The packet features are classified into two types namely spatial and temporal features. The spatial features include information all about the source and destination and the temporal features comprise time-related features which affect the network. From this, the (13) can be formulated as (15).

$$\hat{a} = \arg \max_a P(b_s b_t | a) P(a) \quad (15)$$

The independency of two types of features can be represented as,

$$P(b_s b_t | a) = P(b_s | a) P(b_t | a) \quad (16)$$

Initially, the semantic features of normal packets are considered. The semantic score of the normal packets is formulated and the mapping of incoming suspicious packets is taken place. Here the packets having highest matching with the normal packets are considered normal and those packets which are lowest matching are termed anomalies as shown in Figure 1. By doing so, the IDS can detect even new anomalies generated in the network. The score calculation is computed by cumulating the individual score of spatial features, temporal features, and labels. The classification of packets based on score can be formulated as (17),

$$\hat{a} = \arg \max_a P(b_s | a)^\beta P(a) \prod_{f \in b} P(f | a)^{n(f,b)} \quad (17)$$

where  $\beta$  denotes the scaling factor,  $f$  denotes the packet features and  $n(f, b)$  represents the frequency of packet features.

### 3.3. Third-level IDS

This level performs to detect an adversary packet which is also under anomaly detection but adversary packets look like normal packets. Detection of adversary packets is a challenging task in network intrusion detection. In our proposed system we used conditional generative adversarial network (CGAN) which detects adversary packets because it automatically learns the adversarial environment and detects the adversarial packets well. The pseudo-code for CGAN-based adversarial sample detection is presented above in which the detection of new adversaries is performed to ensure the overall security of IoT environment. Figure 2. Illustrates the CGAN based adversary detection. This process improves the attack detection rate and accuracy. A traditional GANT consists of two kinds of modules as generator and discriminator. Both types of modules use neural network models. Especially, generator module used neural network module, and discriminator used convolutional neural network module. For adversarial intrusion detection task, anomaly score is computed to quantify the incoming attack patterns and CGAN is used to judge how the samples are not similar to the trained set and it computes the great difference between the testing samples and learned patterns of normal packets. Here, adversary score is computed as (18) and (19).

$$AS = (1 - \sum_{i=1}^n \lambda_i) \mathfrak{R}_l + \sum_{i=1}^n \lambda_i l_{D_i} \tag{18}$$

$$\mathfrak{R}_l = |x - G(z)| \tag{19}$$

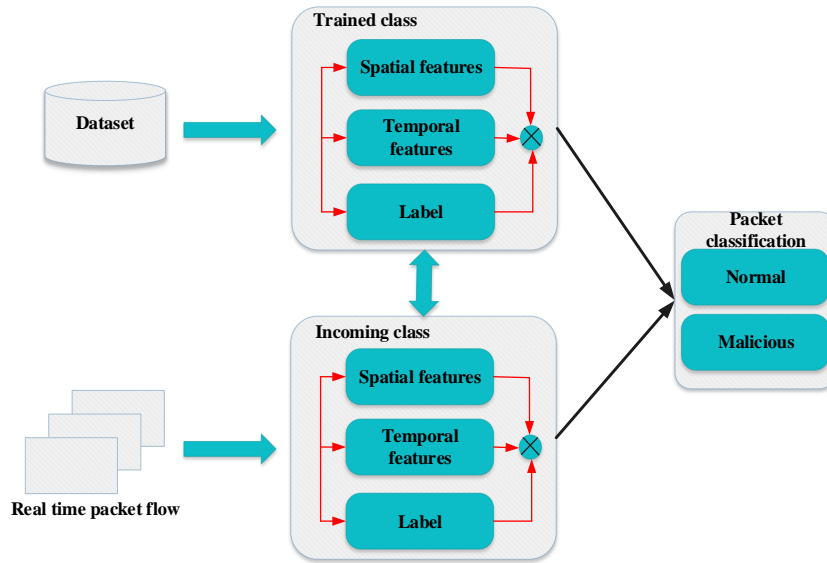


Figure 1. Event-based semantic mapping

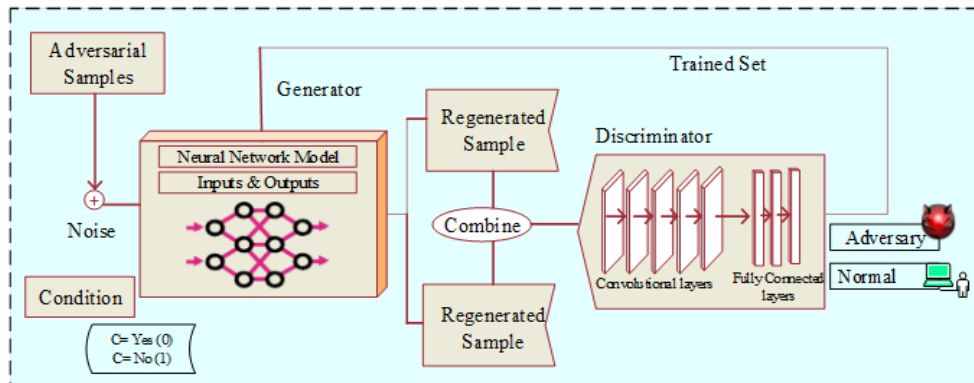


Figure 2. CGAN for adversarial samples detection

**Pseudocode for conditional generative adversarial network (CGAN)**

INPUT: Anomaly packets and patterns

OUTPUT: Adversary/Normal

1. Begin
2. Initialize attack patterns
3. For packet 1...n
4. Learning rate is initialized for Generator (G) and Discriminator (D)
5. Adjust the parameters
6. Weight value is set for both generators ( $W_G$ ) and discriminator ( $W_D$ )
7. While (not satisfied) do
8. Update G
9. Collect new attack patterns
10. Predict  $AS_i$  for all packets
11. Collect updated  $W_G$
12. Update the module D
13. end while
14. end for
15. end

### 3.4. Attack type classification

After completing three-level IDS we know that the suspicious packets are normal or malicious. And next classification is performed by using proximal policy optimization (PPO) algorithm which is under reinforcement learning as illustrated in Figure 3 which learned the type of attacks from the training dataset by correlating the abnormal events, hence it provides accurate classification. This stage classifies what type of attack is occurring in our network (ex. DDoS, man in the middle, phishing, SQL injection, and false data injection). PPO is a deep reinforcement learning algorithm that follows actor and critic architecture. PPO balances between implementation ease, tuning ease, and sample complexity and tries to compute the updates for each step and minimizes the cost function which ensures the deviation from the previous policies are relatively lower. A stochastic policy is defined as  $\pi_{\theta}(\alpha_t|\delta_t)$  that maps states with Gaussian distribution on the set of actions. A critic value function is  $V_w(\delta_t)$  that result from the average reward in state  $\delta_t$ . In order to update the actor's parameters, a clipped surrogate objective is applied. Further, loss function is used to clip the surrogate objective in which the  $r_t(\theta)$  denotes the probability ratio which is computed as (20),

$$r_t(\theta) = \frac{\pi_{\theta}(\alpha_t|\delta_t)}{\pi_{\theta_{old}}(\alpha_t|\delta_t)} \quad (20)$$

where  $\theta_{old}$  is the actor's parameter vector before the update. A new varied objective function is defined as (21),

$$L^{CLIP}(\theta) = \hat{E}_t[\min(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1 - \varepsilon, 1 + \varepsilon)\hat{A}_t)] \quad (21)$$

where  $\theta$  is the policy parameter,  $\hat{E}_t$  is the empirical expectation over the time steps,  $r_t$  represents the ratio of the probability from the old and new policies, respectively,  $\hat{A}_t$  represents the estimated advantage at time t.  $\varepsilon$  represents the hyper parameter, which is usually between 0.1 and 0.2. In PPO, attacks are classified into specific types based on the attack patterns. From the source, packet and flow information is acquired. Table 2 illustrates the attack patterns and the corresponding classes. Pseudocode for PPO can be followed in which the attack types are classified based on the observed patterns of the attacks. The PPO learns the environment and estimates the attack type utilizing respective policies. By learning attack patterns from the source packet to the destination, PPO computes the corresponding actions from the policy attributes, and loss is computed to update the training set.

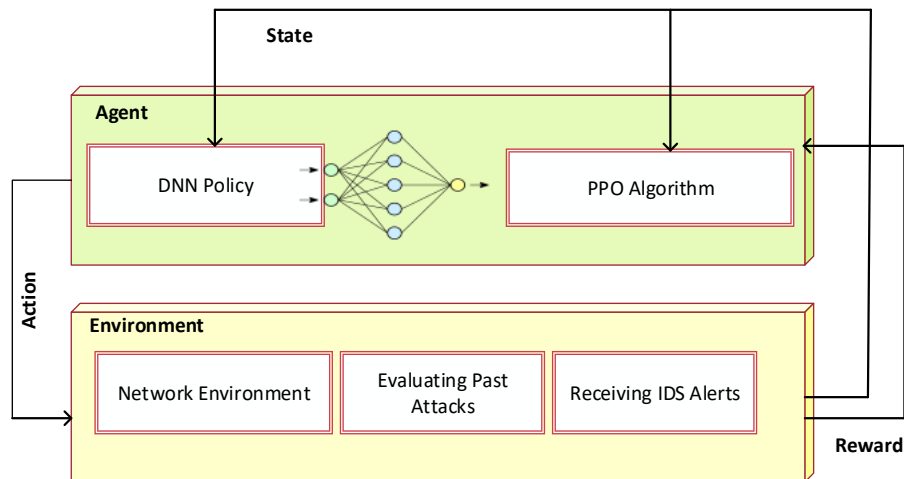


Figure 3. PPO algorithm for attack type classification

#### Procedure for proximal policy optimization (PPO)

1. Begin
- // Attacks types detection
2. Initialize all attack patterns
3. For  $e \in \text{episodes}$  do
4. For  $a \in \text{actors}$  do
5. Run policy  $\pi_{\theta_{old}}$  in environment for  $T$  time steps
6. Compute the advantage estimations  $\hat{A}_1 \dots \hat{A}_T$



7. Predict attack type by policies
8. End For
9. Optimizes actor's loss function i.e.,  $L^{CLIP}(\theta)$  by. (21)
10.  $\theta_{old} \leftarrow \theta$
11. End for

Table 2. Attack type classification

No.	Attack Patterns	Type of Attack
1	Duration of connection Source bytes Number of files creations Number of files accessed	Probe Attack
2	Percentage of connections Source bytes Percentage of packets with errors	DoS Attacks
3	Service requested Host level features	R2L Attacks
4	Number of failed login attempts Number of file creations Number of shell prompts invoked	U2R Attacks

#### 4. RESULTS AND DISCUSSION

In this section, we describe the experimental analysis of the proposed 3LIDS-CGAN model in IoT environment to validate the performance. This section includes three sub-sections such as simulation setup, comparative analysis, and research summary. The result section shows that the proposed 3LIDS-CGAN model achieves superior performance compared to the previous models. In the following, a detailed design of the experimental study is presented.

##### 4.1. Simulation setup

This section explains the simulation setup of the proposed 3LIDS-CGAN model which is experimented with and evaluated using NS-3.26 network simulator. This simulation tool consists of the specifications which are correlated to the proposed 3LIDS-CGAN model. The simulation of the proposed 3LIDS-CGAN method is performed with  $800 \times 1000$  m simulation area. Tables 3 and 4 illustrate the system configuration and the network configuration respectively.

Table 3. System configuration

Software Specifications		Hardware specifications	
Operating system	Ubuntu 14.04 LTS	Hard Disk	60 GB
Network Simulator	NS3.26	RAM	2 GB
		Processor	Pentium Dual Core and above

Table 4. Simulation parameters

Parameter	Value	Parameter	Value
Network Parameters		Mobility Parameters	
No. of IoT users	100	Mobility of node	10 m/s
No. of firewall	1	Range of transmission	150 m
No. of gateway	1	Type of mobility model	Random waypoint
No. of cloud server	1	Network Traffic Parameters	
Simulation Area	$800 \times 1000$ m	Type of queue	FIFO
No. of malicious nodes	1-5 nodes	No. of traffic flows	2-5
Initial Energy	100 J	Type of traffic	CBR, TCP, UDP
Simulation time	300s	Bandwidth of channel	100 MHz
Modules	Wi-Fi, Internet, Ipv4	Security Parameters	
Packet Transmission parameters		Attack frequency	10-20 packets per sec
Type of protocol	UDP	Probability ratio of attacks	1:10
Data rate of packets	100 Mbps	No. of attacks	~4
Interval of packets	1s	Interval of attacks	3-5 Sec
No. of packets	1000	Detected attacks	Probe attack, DoS attack, DDoS attack, SQL injection attack, R2L attack, and U2R attack
No. of retransmission	7		
Size of packets	64, 128, 256, 512, 1024 bytes		

The NSL-KDD dataset is an extended version of KDDCUP00 and the network-based intrusions are categorized into the following four classes which are presented in Table 5.

- Probe: if an attacker wants to use the gain information of the target network via device/system/host scanning activities.
- DoS: if an attacker interrupts the connection of authorized hosts that access to a given node is allowed
- U2R: if any attackers attempt for an escalating the limited user's privilege to the super user or the root access. For instance, malware infection or credentials stolen
- R2L: if an attacker wants to remote access the victim machine that imitates the previous legitimate hosts.

Table 5. Attacks in NSL-KDD dataset

Attack Main Type	Subclass (Attack) in Trained Set	New Sub Classes (Attacks) in Tested Set
DoS	Back, Land, Smurf, Pod, Neptune, TearDrop	Apache 2, Processtable, Mailbomb
Probe	Imap, Multihop, Phf, Spy, Warezclient, Warezmater, Ftp write, Guess Passwd	Mscan, Saint
U2R	Buffer Overflow, Perl, Load Module, Rootkit	Httpunnel, Ps, Sqlattack, Xterm
R2L	Ipsweep, NMAP, Portsweep, Satan	Sendmail, Named, Snpgetattack, Snm guess, Xlock, Xsnoop, and Worm

## 4.2. Comparative study

In this section, we explain the comparison between the proposed 3LIDS-CGAN model and existing methods such as hybrid intrusion detection system (HIDS) [35] and scalable and hybrid intrusion detection system (SHIDS)-long short-term memory (LSTM) [36]. This work aims to detect IDS in IoT environment. The proposed work is compared to the existing works and proved that the proposed 3LIDS-CGAN model achieves better performance in terms of attack detection rate, false-positive rate, true positive rate, F-score, accuracy, energy consumption, precision, recall, roc curve.

### 4.2.1. Impact of attack detection rate (ADR)

This metric is used to calculate the number of attacks detected in a given amount of time, which is calculated with respect to true positive values, which are calculated as (22),

$$ADR = \frac{\tau}{2} \times 100\% \quad (22)$$

where ADR represents attack detection rate and  $\tau$  represents number of detected attacks and 2 represents total number of attacks.

Figure 4 represents the comparison of attack detection rate for both existing HIDS and SHIDS-LSTM and proposed 3LIDS-CGAN model with respect to number of malicious nodes. The figure clears that the proposed 3LIDS-CGAN model achieves high detection rate compared to existing works. A network with high attack detection rate increases the security against various types of threats. The attack detection rate decreases with increasing the number of malicious nodes which is otherwise known as the attack detection rate are inversely proportional to the number of malicious nodes. In the proposed 3LIDS-CGAN model, we detect the attacks by performing three-level IDS and classification, hence we achieve high attack detection rate. The previous works perform only anomaly-based intrusion detection or signature-based intrusion detection that does not provide accurate results in attack detection. Some of the works are concentrated on both signature and anomaly-based intrusion detection [35], [36] which performs well, however, it does not detect adversarial packets that are also one of the intrusions that looks like a normal packet. Detecting adversary packets is a challenging task in that if it does not detect may lead to destroying our data or network.

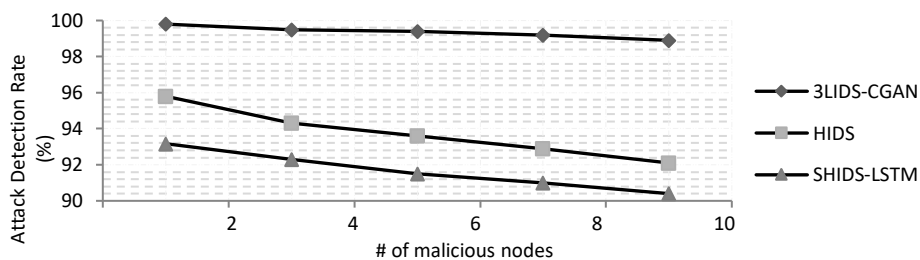


Figure 4. Attack detection rate vs. # of malicious nodes

In order to overcome the aforementioned issues, we detect the adversary packets with the help of CGAN in the proposed 3LIDS-CGAN model that increases attack detection rate compared to the existing works. We achieve high attack detection rate due to detecting adversarial packets in the network. The proposed 3LIDS-CGAN model achieves 98.9% of attack detection rate for the 10 malicious nodes which are 8% higher than SHIDS-LSTM and 6% higher than HIDS models for the same number of malicious nodes. Table 6 illustrates the numerical analysis of attack detection rate. That compares the number of malicious packets and corresponding attack detection rate for both proposed (3LIDS-CGAN) and existing models (HIDS, SHIDS-LSTM).

Table 6. Attack detection rate (%) analysis

No. of. Malicious nodes	3LIDS-CGAN	HIDS	SHIDS-LSTM
2	99.8 ± 0.12	95.8 ± 0.34	93.17 ± 0.52
4	99.5 ± 0.16	94.3 ± 0.36	92.3 ± 0.51
6	99.4 ± 0.14	93.6 ± 0.35	91.5 ± 0.57
8	99.2 ± 0.17	92.9 ± 0.39	91 ± 0.53
10	98.9 ± 0.19	92.1 ± 0.37	90.4 ± 0.55

#### 4.2.2. Impact of false positive rate (FPR)

This metric is used to detect wrongly classified packets in the environment, which is calculated as (23),

$$FPR = \frac{\eta}{\alpha} \times 100\% \quad (23)$$

where FPR represents the false positive rate and  $\eta$  represents the number of misclassified packets and  $\alpha$  represents the total amount of packets. The main aim of false-positive rate is to reduce misclassification.

Figure 5 represents the comparison of false-positive rate with respect to number of malicious nodes between the proposed 3LIDS-CGAN model and other existing approaches. A network with low false-positive rate increases the security in terms of detecting the attacks accurately. In the proposed 3LIDS-CGAN model, intrusions are detected in three stages. In first stage, the malicious packets are dropped, and suspicious packets are sent to the next level for detection of intrusions. In second level, the suspicious packets are classified as normal or malicious. Third-level IDS detects adversary packets. After completing the three levels we detect intrusion accurately, hence it reduces false positive rate and misclassification. In [36], the intrusions are detected at one stage thus increasing misclassification and they are not focused to detect adversary packet that increases misclassification and false-positive rate. The comparison results show that the proposed 3LIDS-CGAN model achieves less false positive rate compared to existing works. The false-positive rate is increased exponentially with the increased number of malicious nodes. The proposed 3LIDS-CGAN model achieves 0.2 false-positive rates for 10 malicious nodes which are 0.4 less than SHIDS-LSTM and 0.2 less than HIDS for the same number of malicious nodes. Table 7 shows the numerical analysis of false positive rate and that shows the average value of the false positive rate with respect to number of malicious nodes.

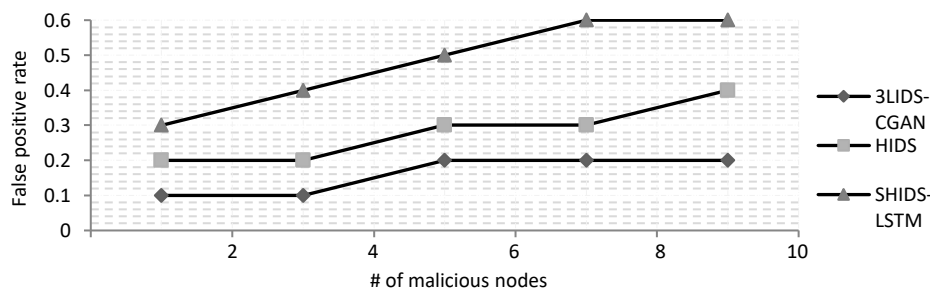


Figure 5. False-positive rate vs. # of malicious nodes

Table 7. False-positive rate (%) analysis

No. of. Malicious nodes	3LIDS-CGAN	HIDS	SHIDS-LSTM
2	0.1 ± 0.014	0.2 ± 0.026	0.3 ± 0.052
4	0.1 ± 0.017	0.2 ± 0.032	0.4 ± 0.057
6	0.2 ± 0.013	0.3 ± 0.035	0.5 ± 0.051
8	0.2 ± 0.018	0.3 ± 0.033	0.6 ± 0.056
10	0.2 ± 0.011	0.4 ± 0.037	0.6 ± 0.055

#### 4.2.3. Impact of true positive rate (TPR)

This metric is used to detect correctly classified packets from the total amount of packets. If the system has high true positive then it will achieve high accuracy. The calculation of true positive rate is defined as (24),

$$TPR = \frac{TP}{TP+FN} \quad (24)$$

where TPR represents the true positive rate and TP represents true positive, FN represents false negative.

Figure 6 represents the comparison of true positive rates for both proposed and existing models with respect to number of malicious nodes. A network with high true positive rate increases the attack detection accuracy. The true positive rate decreases with increasing the number of malicious nodes. The comparison results show that the proposed 3LIDS-CGAN model achieves high true positive rate compared to other existing models because the proposed 3LIDS-CGAN model deploys firewall for detecting and avoiding malicious packets in the first stage thus increasing true positive rate. The malicious packets are detected and avoided in the first level of IDS and the anomalies are detected and dropped in the second level of IDS, finally, adversary packets are detected and removed from the environment thus increasing true positive rate and reducing misclassification. Whereas the existing method [36], hybrid IDS uses dataset to detect the anomalies which are trained that reduce the true positive rate due to lack of ability to detect the real-time anomalies. Table 8 illustrates the numerical analysis of true positive rate. From the numerical analysis, the proposed 3LIDS-CGAN model achieves 0.8 true positive rates for 10 malicious nodes which are 0.27% higher than SHIDS-LSTM model and 0.12% higher than HIDS model.

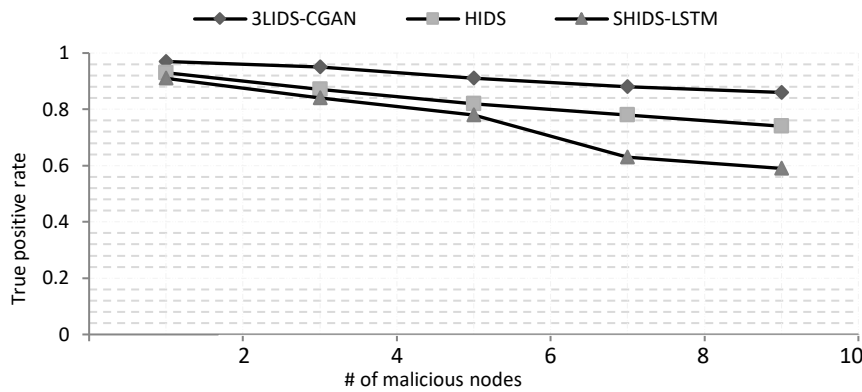


Figure 6. True positive rate vs. # of malicious nodes

Table 8. True positive rate (%) analysis

No. of. Malicious nodes	3LIDS-CGAN	HIDS	SHIDS-LSTM
2	0.97 ± 0.010	0.93 ± 0.045	0.91 ± 0.057
4	0.95 ± 0.015	0.87 ± 0.048	0.84 ± 0.052
6	0.91 ± 0.013	0.82 ± 0.942	0.78 ± 0.058
8	0.88 ± 0.016	0.78 ± 0.050	0.63 ± 0.053
10	0.86 ± 0.018	0.74 ± 0.047	0.59 ± 0.060

#### 4.2.4. Impact of F-score

This metric is used to evaluate the test accuracy. This score is calculated from the precision and recall harmonic mean. The calculation of F-score is defined as (25),

$$F = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (25)$$

Figure 7 represents the comparison of F-score with respect to number of malicious nodes for both proposed and existing models. The F-score decreases with increasing the number of malicious nodes. The result shows that the proposed 3LIDS-CGAN model achieves high F-score compared to existing models because we detect harmonic mean of precision and recall accurately compared to existing work.

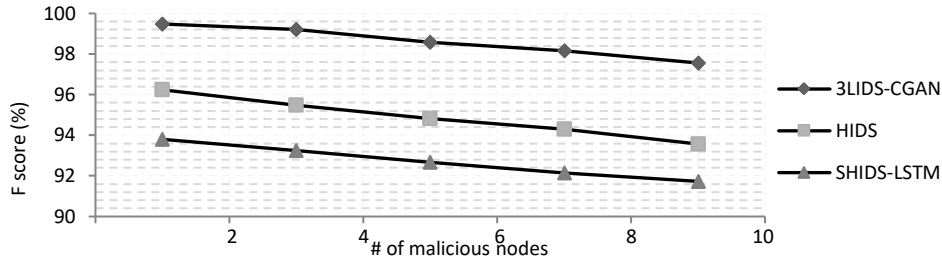


Figure 7. F-score vs. # of malicious nodes

The proposed 3LIDS-CGAN correctly classified the intrusions from the environment thus increasing F-score value. For instance, SHIDS [36] used ISCX-UNB dataset for intrusion detection, hence it detects only trained anomalies using of single dataset it does not detect real-time anomalies because the attack behavior is changed frequently, but trained anomalies store only limited behavior thus leads to poor accuracy and less F-measure value. Table 9 illustrates the numerical analysis of F-score which represent the average value of F-score [42], [43]. The numerical analysis proved that the proposed 3LIDS-CGAN model achieves high F-score compared to existing models. The proposed 3LIDS-CGAN achieves 97.5% of F-score which is 6% higher than SHIDS-LSTM and 4% higher than HIDS.

Table 9. F score (%) analysis

No. of. Malicious nodes	3LIDS-CGAN	HIDS	SHIDS-LSTM
2	99.48 ± 0.15	96.24 ± 0.37	93.8 ± 0.58
4	99.21 ± 0.18	95.47 ± 0.36	93.25 ± 0.51
6	98.58 ± 0.16	94.82 ± 0.38	92.68 ± 0.59
8	98.16 ± 0.13	94.31 ± 0.34	92.14 ± 0.57
10	97.56 ± 0.17	93.57 ± 0.32	91.73 ± 0.54

#### 4.2.5. Impact of accuracy

This metric is used to calculate the accuracy of the proposed 3LIDS-CGAN model. If the system has high accuracy, it represents the system detects intrusions correctly. Accuracy is calculated as (26),

$$A = \frac{TP+TN}{TP+TN+FP+FN} \quad (26)$$

where A represents accuracy and FP represents false positive, TN represents true negative. Figure 8 represents the comparison of accuracy for both the proposed 3LIDS-CGAN model and existing models with respect to number of malicious nodes. The accuracy decreases with increasing the number of malicious nodes. A network with high accuracy increases the detection rate of attacks. In the existing method [35], lack of considering the adversaries during hybrid IDS results in low accuracy. In [36], lack detection the real-time anomalies reduces the accuracy of attack detection. In order to overcome the aforementioned issues, firewall is implemented that filters the incoming packets, and the malicious packets are dropped earlier which increases the accuracy. The comparison results show that the proposed 3LIDS-CGAN achieves high accuracy compared to other existing models. For instance [38] consider two datasets for intrusion detection still do not detect real-time intrusion thus reducing accuracy and attack detection rate.

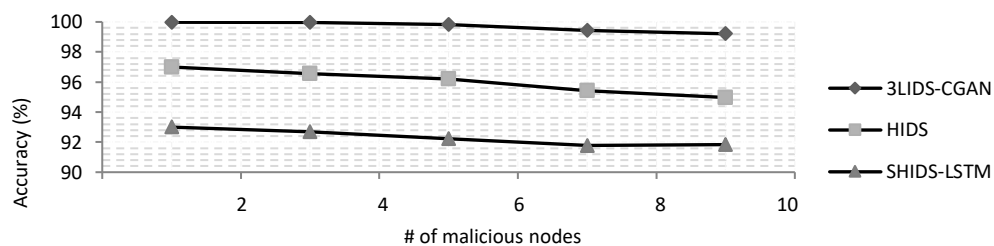


Figure 8. Accuracy vs. #of malicious nodes

Table 10 illustrates the numerical analysis of accuracy. From the analysis, the proposed 3LIDS-CGAN achieves high accuracy. The 3LIDS-CGAN achieves high accuracy of about 99% for 10 malicious nodes which are 8% higher than SHIDS-LSTM and 5% higher than HIDS model for the same number of malicious nodes.

Table 10. Accuracy (%) analysis

No. of. Malicious nodes	3LIDS-CGAN	HIDS	SHIDS-LSTM
2	99.98 ± 0.15	97 ± 0.36	93 ± 0.54
4	99.97 ± 0.18	96.54 ± 0.37	92.68 ± 0.57
6	99.84 ± 0.16	96.21 ± 0.32	92.23 ± 0.51
8	99.43 ± 0.17	95.43 ± 0.38	91.78 ± 0.59
10	99.21 ± 0.13	94.97 ± 0.34	91.84 ± 0.58

#### 4.2.6. Impact of energy consumption

This metric is used to calculate the energy consumed by the IoT devices during intrusion detection, which is calculated as (27),

$$EC = I_E - R_E \quad (27)$$

where  $EC$  represents energy consumption and  $I_E$  represent initial energy and  $R_E$  represent residual energy.

Figure 9 represents the comparison of energy consumption of proposed 3LIDS-CGAN model and existing model with respect to number of malicious nodes. The energy consumption increases with increasing the number of malicious nodes which is otherwise known as the energy consumption is directly proportional to the number of malicious nodes. The comparative result shows that the proposed 3LIDS-CGAN model consumes less energy compared to existing models because the proposed 3LIDS-CGAN method used firewall to drop malicious packets at an earlier stage thus reducing energy consumption and considering only legitimate packets as an input rather than considering all packets as an input that also reduces the energy consumption. In [36] consider all the packets as an input thus increasing energy consumption and latency. Hence, the proposed 3LIDS-CGAN consumes less energy. Table 11 illustrates the numerical analysis of energy consumption which shows the average value of energy consumption with respect to number of malicious nodes. The result shows that the proposed 3LIDS-CGAN consumes low energy consumption of about 2.8 J for 10 malicious nodes which are 3 J less than SHIDS-LSTM and 2 J less than HIDS model for the same number of malicious nodes.

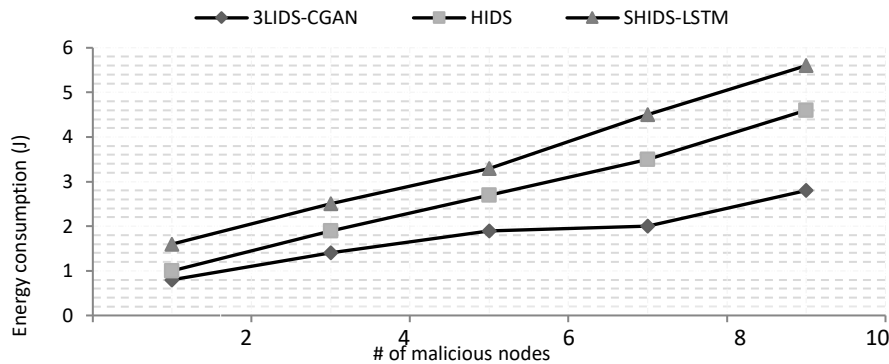


Figure 9. Energy consumption vs., # of malicious nodes

Table 11. Energy consumption (J) analysis

No. of. Malicious nodes	3LIDS-CGAN	HIDS	SHIDS-LSTM
2	0.8 ± 0.017	1 ± 0.027	1.6 ± 0.054
4	1.4 ± 0.019	1.9 ± 0.031	2.5 ± 0.058
6	1.9 ± 0.012	2.7 ± 0.034	3.3 ± 0.052
8	2 ± 0.018	3.5 ± 0.032	4.5 ± 0.055
10	2.8 ± 0.013	4.6 ± 0.037	5.6 ± 0.057

#### 4.2.7. Impact of precision

This metric is used to calculate the correctly classified packets from the total amount of packets. The calculation of precision is defined as (28),

$$Precision = \frac{TP}{TP+FP} \quad (28)$$

Figure 10 represents the comparison of precision for both proposed and existing models with respect to number of malicious nodes. The precision decreases with increasing the number of malicious nodes. The comparison result shows that the proposed 3LIDS-CGAN achieves high precision compared to other existing models. In [35] perform normalization to for detecting intrusions, if any error occurs in the process of normalization, then the system does not detect accurate intrusions thus increasing false positive rate and reducing true positive rate hence it also reduces precision value. In the proposed 3LIDS-CGAN model deploy firewall instead of preprocessing, which is used to filter the incoming packets and drop the malicious packets in the first stage thus increasing true positive rate and precision. In this way, we achieve high precision when compared to SHIDS-LSTM and HIDS models. Table 12 illustrates the numerical analysis of precision which provides the average value of the precision with respect to number of malicious nodes. The proposed 3LIDS-CGAN achieves high precision of about 97.83% for 10 malicious nodes 6% higher than SHIDS-LSTM model and 3% higher than HIDS model for same number of malicious nodes.

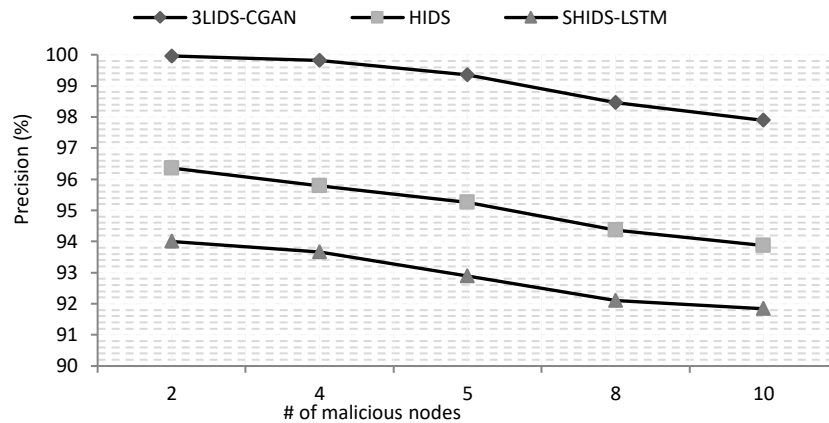


Figure 10. Precision vs. # of malicious nodes

Table 12. Precision (%) analysis

No. of. Malicious nodes	3LIDS-CGAN	HIDS	SHIDS-LSTM
2	99.97 ± 0.16	96.48 ± 0.34	93 ± 0.57
4	99.83 ± 0.11	95.8 ± 0.37	92.68 ± 0.53
6	98.78 ± 0.19	95.17 ± 0.31	92.23 ± 0.51
8	98.36 ± 0.13	94.69 ± 0.35	91.78 ± 0.54
10	97.93 ± 0.18	94.23 ± 0.32	91.84 ± 0.58

#### 4.2.8. Impact of recall

This metric is to calculate the percentage of actual positives in the correctly classified packets which is calculated as (29).

$$Recall = \frac{TP}{TP+FN} \quad (29)$$

Figure 11 represents the comparison of recall for both proposed and existing models with respect to number of malicious nodes. The recall decreases with increasing the number of malicious nodes. The comparison result shows that the proposed 3LIDS-CGAN model achieves high recall compared to existing models because the existing works only detect anomaly and signature-based intrusions however, lack of detection of the adversary packets leads to misclassification and increase false alarm rate, but the proposed

3LIDS-CGAN detect adversary packets using CGAN which detect adversary packets because it automatically learns the adversarial environment and detects adversary packets well thus improves detection rate and recall and reduce misclassification when compared with the existing works.

Table 13 illustrates the numerical analysis of recall that presents the average value of recall with the corresponding malicious packets. From the numerical analysis, the proposed 3LIDS-CGAN achieves 97.895 for 10 malicious nodes which is 6% higher than SHIDS-LSTM and 4% higher than HIDS model for the same number of malicious nodes.

**4.2.9. Impact of ROC curve**

ROC curve stands for receiver operating characteristics curve which shows the classification performance at all classification thresholds. ROC curve includes true positive rate and false-positive rate. It is graph that plots True positive rate vs. false positive rate at various classification thresholds.

Figure 12 represents the ROC curve for both proposed and existing models. The true positive value is increased exponentially with the increasing number of false-positive rates. The figure clearly states that the proposed 3LIDS-CGAN model achieves high true positive rate compared to SHIDS-LSTM and HIDS models. The proposed model has high true positive rate due to detecting intrusions accurately because we deploy firewall to drop the malicious packets in an earlier stage and perform three-level IDS. And CGAN is also used for detecting adversary packets that only the proposed 3LIDS-CGAN achieves high true positive range in ROC curve compared to existing models such as SHIDS-LSTM and HIDS. Whereas, the existing methods, lack adversary and real-time anomaly detection reducing the true positive rate and increasing the false positive rate.

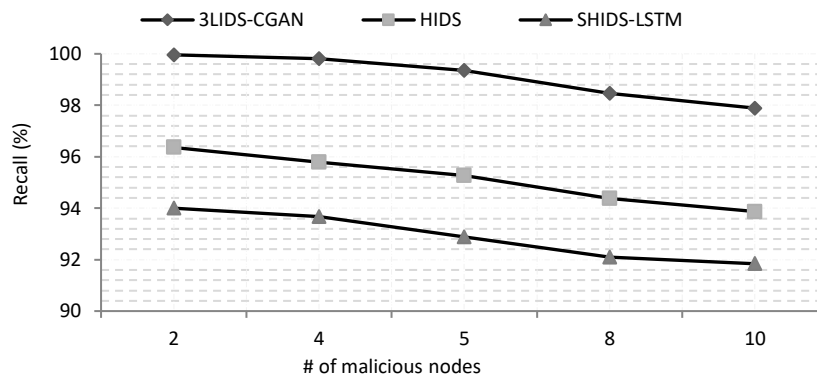


Figure 11. Recall vs. # of malicious nodes

Table 13. Recall (%) analysis

No. of. Malicious nodes	3LIDS-CGAN	HIDS	SHIDS-LSTM
2	99.96 ± 0.18	96.36 ± 0.38	94 ± 0.52
4	99.81 ± 0.12	95.78 ± 0.33	93.67 ± 0.54
6	99.35 ± 0.13	95.26 ± 0.31	92.89 ± 0.56
8	98.47 ± 0.16	94.37 ± 0.36	92.1 ± 0.53
10	97.89 ± 0.19	93.87 ± 0.35	91.84 ± 0.51

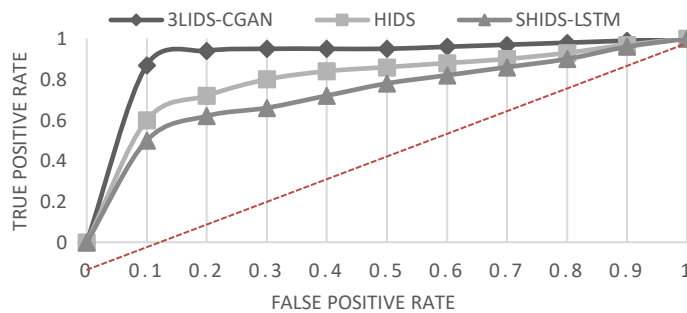


Figure 12. True positive rate vs. false positive rate



## 5. CONCLUSION

In IoT environment, security and privacy are the significant issues. In this paper, 3LIDS-CGAN model is proposed for detecting intrusions in IoT environment. In first level IDS, firewall is deployed for avoiding malicious packets in an earlier stage, for that firewall extracts the features of incoming traffic patterns.

Only the suspicious packets are sent to the next level for detecting anomaly packets. In the second level IDS, event-based semantic map is constructed for detecting anomaly extracted features the packet is classified into normal, malicious, and suspicious using SVM and golden eagle optimization packets from the suspicious packets which increase attack detection accuracy. In third level IDS detect adversary packets detected which are under anomaly detection. The adversary packets look like normal packets we need to identify, for that, we proposed CGAN which detects anomaly packets accurately that accuracy and reduces false alarm rate. Final process is attacking type classification, after detecting intrusion the PPO is deployed to classify the type of attacks such as Probe attack, DoS attack, DDoS attack, SQL injection attack, R2L attack, and U2R attack. The PPO learned and stored the attack type and its behavior in the dataset by correlating the current behavior of the current attack the PPO classifies the specific attack type which increases accuracy of the process. In final analysis, the performance of the proposed 3LIDS-CGAN is evaluated and outperforms in terms of attack detection rate, false-positive rate, true positive rate, F-Score, Accuracy, energy consumption, precision, recall, and ROC curve. In future, we planned to integrate blockchain for enhancing security in IoT environment.





## REFERENCES

- [1] K. Mrhar, O. Douimi, M. Abik, and N. C. Benabdellah, "Towards a semantic integration of data from learning platforms," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 3, pp. 535–544, Sep. 2020, doi: 10.11591/ijai.v9.i3.pp535-544.
- [2] V. Kumar, A. K. Das, and D. Sinha, "UIDS: a unified intrusion detection system for IoT environment," *Evolutionary Intelligence*, vol. 14, no. 1, pp. 47–59, Mar. 2021, doi: 10.1007/s12065-019-00291-w.
- [3] A. Kim, M. Park, and D. H. Lee, "AI-IDS: Application of deep learning to real-time web intrusion detection," *IEEE Access*, vol. 8, pp. 70245–70261, 2020, doi: 10.1109/ACCESS.2020.2986882.
- [4] I. Dutt, S. Borah, and I. K. Maitra, "Immune system based intrusion detection system (IS-IDS): A proposed model," *IEEE Access*, vol. 8, pp. 34929–34941, 2020, doi: 10.1109/ACCESS.2020.2973608.
- [5] R. A. Khamis, M. O. Shafiq, and A. Matrawy, "Investigating resistance of deep learning-based IDS against adversaries using min-max optimization," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Jun. 2020, pp. 1–7, doi: 10.1109/ICC40277.2020.9149117.
- [6] M. Sewak, S. K. Sahay, and H. Rathore, "DOOM: A novel adversarial-DRL-based op-code level metamorphic malware obfuscator for the enhancement of IDS," in *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers*, Sep. 2020, pp. 131–134, doi: 10.1145/3410530.3414411.
- [7] M. Alenezi, M. Nadeem, and R. Asif, "SQL injection attacks countermeasures assessments," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 21, no. 2, pp. 1121–1131, Feb. 2021, doi: 10.11591/ijeecs.v21.i2.pp1121-1131.
- [8] M. J. Babu and A. R. Reddy, "SH-IDS: Specification heuristics based intrusion detection system for IoT networks," *Wireless Personal Communications*, vol. 112, no. 3, pp. 2023–2045, Jun. 2020, doi: 10.1007/s11277-020-07137-0.
- [9] A. Alhawaide, I. Alsmadi, and J. Tang, "PCA, Random-forest and Pearson correlation for dimensionality reduction in IoT IDS," in *2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, Sep. 2020, pp. 1–6, doi: 10.1109/IEMTRONICS51293.2020.9216388.
- [10] G. Soni and R. Sudhakar, "A L-IDS against dropping attack to secure and improve RPL performance in WSN aided IoT," in *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)*, Feb. 2020, pp. 377–383, doi: 10.1109/SPIN48934.2020.9071118.
- [11] M. A. Ferrag, L. Maglaras, A. Ahmim, M. Derdour, and H. Janicke, "RDTIDS: Rules and decision tree-based intrusion detection system for internet-of-things networks," *Future Internet*, vol. 12, no. 3, pp. 44–58, Mar. 2020, doi: 10.3390/fi12030044.
- [12] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, "Network intrusion detection for IoT security based on learning techniques," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019, doi: 10.1109/COMST.2019.2896380.
- [13] V. Morfino and S. Rampone, "Towards near-real-time intrusion detection for IoT devices using supervised learning and Apache spark," *Electronics*, vol. 9, no. 3, Mar. 2020, doi: 10.3390/electronics9030444.
- [14] H. L. A. Q. B and J. Ahmad, *Internet of things*. Springer International Publishing, 2019.
- [15] M. M. Shurman, R. M. Khrais, and A. A. Yateem, "IoT denial-of-service attack detection and prevention using hybrid IDS," in *2019 International Arab Conference on Information Technology (ACIT)*, Dec. 2019, pp. 252–254, doi: 10.1109/ACIT47987.2019.8991097.
- [16] K. Jiang, W. Wang, A. Wang, and H. Wu, "Network intrusion detection combined hybrid sampling with deep hierarchical network," *IEEE Access*, vol. 8, pp. 32464–32476, 2020, doi: 10.1109/ACCESS.2020.2973730.
- [17] S. U. Jan, S. Ahmed, V. Shakhov, and I. Koo, "Toward a lightweight intrusion detection system for the internet of things," *IEEE Access*, vol. 7, pp. 42450–42471, 2019, doi: 10.1109/ACCESS.2019.2907965.
- [18] C. Ma, X. Du, and L. Cao, "Analysis of multi-types of flow features based on hybrid neural network for improving network anomaly detection," *IEEE Access*, vol. 7, pp. 148363–148380, 2019, doi: 10.1109/ACCESS.2019.2946708.
- [19] A. Khraisat, J. Gondal, P. Vamplew, J. Kamruzzaman and A. Alazab, "A novel ensemble of hybrid intrusion detection system for detecting internet of things attacks," *Electronics*, vol. 8, no. 11, Oct. 2019, doi: 10.3390/electronics8111210.
- [20] S. P. R.M. *et al.*, "An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IoMT




- architecture,” *Computer Communications*, vol. 160, pp. 139–149, Jul. 2020, doi: 10.1016/j.comcom.2020.05.048.
- [21] Y. Zhou, T. A. Mazzuchi, and S. Sarkani, “M-AdaBoost-A based ensemble system for network intrusion detection,” *Expert Systems with Applications*, vol. 162, Dec. 2020, doi: 10.1016/j.eswa.2020.113864.
- [22] D. Mohammad, I. Aljarrah, and M. Jarrah, “Searching surveillance video contents using convolutional neural network,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 2, pp. 1656–1665, Apr. 2021, doi: 10.11591/ijece.v11i2.pp1656-1665.
- [23] W. J. Hadi, S. M. Kadhem, and A. R. Abbas, “Fast discrimination of fake video manipulation,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 3, pp. 2582–2587, Jun. 2022, doi: 10.11591/ijece.v12i3.pp2582-2587.
- [24] M. Berrahal and M. Azizi, “Optimal text-to-image synthesis model for generating portrait images using generative adversarial network techniques,” *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 25, no. 2, pp. 972–979, Feb. 2022, doi: 10.11591/ijeecs.v25.i2.pp972-979.
- [25] M. A. Zaytar and C. El Amrani, “Satellite image inpainting with deep generative adversarial neural networks,” *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, pp. 121–130, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp121-130.
- [26] H. Yang and F. Wang, “Wireless network intrusion detection based on improved convolutional neural network,” *IEEE Access*, vol. 7, pp. 64366–64374, 2019, doi: 10.1109/ACCESS.2019.2917299.
- [27] K. S. Gill, S. Saxena, and A. Sharma, “GTM-CSec: Game theoretic model for cloud security based on IDS and honeypot,” *Computers & Security*, vol. 92, May 2020, doi: 10.1016/j.cose.2020.101732.
- [28] S. Huang and K. Lei, “IGAN-IDS: An imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks,” *Ad Hoc Networks*, vol. 105, Aug. 2020, doi: 10.1016/j.adhoc.2020.102177.
- [29] Y. Zhang, P. Li, and X. Wang, “Intrusion detection for IoT based on improved genetic algorithm and deep belief network,” *IEEE Access*, vol. 7, pp. 31711–31722, 2019, doi: 10.1109/ACCESS.2019.2903723.
- [30] E. Ülker and I. M. Nur, “A new hybrid IoT-based IDS using binary gray wolf optimization (BGWO) and Naive Bayes (NB),” (in Turkish), *European Journal of Science and Technology*, pp. 279–286, Oct. 2020, doi: 10.31590/ejosat.804113.
- [31] J. C. S. Sicato, S. K. Singh, S. Rathore, and J. H. Park, “A comprehensive analyses of intrusion detection system for IoT environment,” *Journal of Information Processing Systems*, vol. 16, no. 4, pp. 975–990, 2020, doi: 10.3745/JIPS.03.0144.
- [32] A. Golrang, A. M. Golrang, S. Y. Yayilgan, and O. Elezaj, “A novel hybrid IDS based on modified NSGAII-ANN and random forest,” *Electronics*, vol. 9, no. 4, Mar. 2020, doi: 10.3390/electronics9040577.
- [33] P. Sun *et al.*, “DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system,” *Security and Communication Networks*, vol. 2020, pp. 1–11, Aug. 2020, doi: 10.1155/2020/8890306.
- [34] K. Atefi, H. Hashim, and T. Khodadadi, “A hybrid anomaly classification with deep learning (DL) and binary algorithms (BA) as optimizer in the intrusion detection system (IDS),” in *2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA)*, Feb. 2020, pp. 29–34, doi: 10.1109/CSPA48992.2020.9068725.
- [35] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, “Hybrid intrusion detection system based on the stacking ensemble of C5 decision tree classifier and one class support vector machine,” *Electronics*, vol. 9, no. 1, Jan. 2020, doi: 10.3390/electronics9010173.
- [36] M. Khan, M. Karim, and Y. Kim, “A scalable and hybrid intrusion detection system based on the convolutional-LSTM network,” *Symmetry*, vol. 11, no. 4, Apr. 2019, doi: 10.3390/sym11040583.
- [37] T. Rose, K. Kifayat, S. Abbas, and M. Asim, “A hybrid anomaly-based intrusion detection system to improve time complexity in the internet of energy environment,” *Journal of Parallel and Distributed Computing*, vol. 145, pp. 124–139, Nov. 2020, doi: 10.1016/j.jpdc.2020.06.012.
- [38] C. A. de Souza, C. B. Westphall, R. B. Machado, J. B. M. Sobral, and G. dos S. Vieira, “Hybrid approach to intrusion detection in fog-based IoT environments,” *Computer Networks*, vol. 180, Oct. 2020, doi: 10.1016/j.comnet.2020.107417.
- [39] S. F. Pane, R. M. Awan, M. A. H. Siregar, and D. Majesty, “Mapping log data activity using heuristic miner algorithm in manufacture and logistics company,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 19, no. 3, pp. 781–791, Jun. 2021, doi: 10.12928/telkomnika.v19i3.18153.
- [40] K. V. Kumar and J. Harikiran, “Privacy preserving human activity recognition framework using an optimized prediction algorithm,” *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 11, no. 1, pp. 254–264, Mar. 2022, doi: 10.11591/ijai.v11.i1.pp254-264.
- [41] T. Kim, C. Ro, and K. Suh, “Experiments on city train vibration anomaly detection Using deep learning approaches,” *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 20, no. 1, pp. 329–337, Oct. 2020, doi: 10.11591/ijeecs.v20.i1.pp329-337.
- [42] M. Eskandari, Z. H. Janjua, M. Vecchio, and F. Antonelli, “Passban IDS: An intelligent anomaly-based intrusion detection system for IoT edge devices,” *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6882–6897, Aug. 2020, doi: 10.1109/JIOT.2020.2970501.
- [43] G. Zhang, X. Wang, R. Li, Y. Song, J. He, and J. Lai, “Network intrusion detection based on conditional Wasserstein generative adversarial network and cost-sensitive stacked autoencoder,” *IEEE Access*, vol. 8, pp. 190431–190447, 2020, doi: 10.1109/ACCESS.2020.3031892.

## BIOGRAPHIES OF AUTHORS






**Hasan Abdulameer**     received his master's degree in Communication and Computer Engineering from the National University of Malaysia, Selangor, Malaysia. He is currently an Assistant Lecturer of Information Technology with Al-Hussain University College. His research interests include computer networks, network security, IDS, and cyber security. He is a certified ethical hacker (CEH). He can be contacted at email: hasanabdulameer@huciraq.edu.iq.



**Inam Musa**    received the B.Sc. degree in communication engineering and M.Sc. degrees in computer and communication engineering from Arts, Sciences & Technology University in Lebanon in 2010 and 2015, respectively. She has been an assistant Lecturer of electronic communication engineering and computer science with Al-Hussain University College, since 2015. She is currently the coordinator of electric power technique engineering department. Her research interests include the applications of image processing and security of communication systems. She can be contacted at email: [anaam.r.majidi@gmail.com](mailto:anaam.r.majidi@gmail.com).



**Noora Salim Al-Sultani**    received the B.Sc. degree in electrical engineering from Babylon university, Babylon, Iraq, 2008, M.Sc. degree in electronic and communication engineering from National Energy University, Malaysia, 2013 and Ph.D. student in National University of Malaysia since 2021 till now, 14 research papers conference and journals. Her research interests include internet of things applications in machine learning, 6G antenna theory, radio spectrum utilization beyond 5G and fiber optics engineering communication. She can be contacted at email [noora.salim@wrec.uoqasim.edu.iq](mailto:noora.salim@wrec.uoqasim.edu.iq).