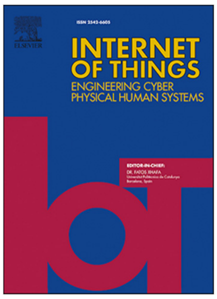
Journal Pre-proof



Revolutionizing intrusion detection in industrial IoT with distributed learning and deep generative techniques

Djallel Hamouda, Mohamed Amine Ferrag, Nadjette Benhamida,
Hamid Seridi, Mohamed Chahine Ghanem

PII: S2542-6605(24)00091-X
DOI: https://doi.org/10.1016/j.iot.2024.101149
Reference: IOT 101149

To appear in: *Internet of Things*

Received date : 30 August 2023
Revised date : 26 February 2024
Accepted date : 26 February 2024

Please cite this article as: D. Hamouda, M.A. Ferrag, N. Benhamida et al., Revolutionizing intrusion detection in industrial IoT with distributed learning and deep generative techniques, *Internet of Things* (2024), doi: https://doi.org/10.1016/j.iot.2024.101149.

Revolutionizing Intrusion Detection in Industrial IoT with Distributed Learning and Deep Generative Techniques

Djallel Hamouda, Mohamed Amine Ferrag, Nadjette Benhamida, Hamid Seridi, Mohamed Chahine Ghanem

***Abstract*—In response to escalating cyber threats and privacy issues within the Industrial Internet of Things (IIoT), this research presents FedGenID, an advanced Federated Generative Intrusion Detection System, to safeguard IIoT networks. Our approach introduces a three-model framework: 1) a federated generative model, incorporating a Conditional Generative Adversarial Network (cGANs) for data augmentation, emphasizing only generator model updates to be shared among clients. This model uses a Wasserstein loss function with Gradient Penalty to amplify sample diversity, indicative of varying cyber threats. Concurrently, we address the issues of imbalanced and distributed data and deploy a data curation technique to align generated data within specific constraints. 2) A secondary model fine-tunes local Critics for enhanced resilience and detection of various adversarial attacks. 3) The third model focuses on precise cyber threat identification, leveraging augmented data for improved training under a synthetic federated learning schema, bolstering detection capability, especially against zero-day threats. Our evaluation of FedGenID, utilizing a novel industrial cybersecurity dataset, highlights its efficacy in non-IID, multi-class cyber threat detection and its resilience to adversarial attacks. Furthermore, we demonstrate how FedGenID can mitigate the negative impact of differential privacy-enhanced FL on model performance. The findings underscore FedGenID's proficiency in detection accuracy, surpassing traditional FedID by 10% in the presence of zero-day attacks and high privacy regimes.**

***Index Terms*—Cybersecurity, Generative AI, GAN, Intrusion Detection, Industrial IoT.**

I. Introduction

Driven by the demand for increased automation, autonomy, and business reliability, the industrial Internet of Things (IoTs) exemplifies an emergent paradigm that enables a seamless connection between machinery and the digital sphere. This facilitates data acquisition and processing using emerging technologies, including cloud/fog computing, 5G/6G wireless networks, and big data analytics for the functioning of the smart factory. However, cyber-criminals are leveraging the inherent security weaknesses of internet-connected systems, besides the insecure-by-design industrial communication protocols [1], to breach valuable assets and conduct severe cyber attacks, including denial of service and privacy intrusions [2]. Consequently, the industry and security researchers are developing new cybersecurity strategies to protect privacy and secure industrial networks and control systems from large-scale cyber threats. As an effective countermeasure, intrusion detection systems (IDS) have been successfully applied to provide security monitoring to identify any possible cyber-attacks in progress. The system's role in network security is to detect abnormal activities based on behavior analysis of generated network traffic data [3]–[5].

Recently, machine learning (ML) and deep learning (DL) classification approaches have been effectively employed in this field to treat and handle the required cyber attack behaviors, degrees of difficulty, and complexity [6]. However, these approaches are computationally intensive, and their effectiveness is constrained by the availability of high-quality training data, which is crucial for defending against zero-day attacks. These constraints have implications for the security of industrial IoTs, where data is heterogeneous and may not cover the required quantity for efficient detection; data privacy is a top priority and major concern; and industrial systems are resource-constrained, which restricts available resources for IDS computation [7]. In this context, a novel distributed learning paradigm called "Federated Learning" (FL) has emerged to overcome these limitations, improving the performance of IDS in terms of detection accuracy and resource utilization for the security of industrial IoTs [6], [8]. It enables many edge devices, where data is generated and resides, to jointly train a global model through transfer learning in each synchronized round of local training without data sharing, thus ensuring data privacy protection. However, a significant challenge that compromises the efficacy of FL depends on the frequency of non-iid (non-independent and identically distributed) data.

In addition, recent studies have demonstrated the vulnerability of ML and DL models to adversarial attacks, primarily attributed to the issue of data inaccessibility [9]. These attacks exploit the vulnerabilities in FL's training and inference processes, compromising model integrity and data privacy. During training, adversaries employ poisoning attacks to manipulate the model's learning process and compromise performance [10]. To respond to these threats, researchers are exploring secure aggregation and authentication schemes to ensure model

M. A. Ferrag is the corresponding author.

D. Hamouda is with Labstic Laboratory, Department of Computer Science, Guelma University, B.P. 401, 24000, Algeria e-mail: hamouda.djallel@univ-guelma.dz

M. A. Ferrag is with Department of Computer Science, Guelma University, B.P. 401, 24000, Algeria and with Technology Innovation Institute, 9639 Masdar City, Abu Dhabi, United Arab Emirates email: mohamed.ferrag@tii.ae

N. Benhamida is with Labstic Laboratory, Department of Computer Science, Guelma University, B.P. 401, 24000, Algeria e-mail: benhamida.nadjette@univ-guelma.dz

H. Seridi is with Labstic Laboratory, Department of Computer Science, Guelma University, B.P. 401, 24000, Algeria e-mail: hamid.seridi@univ.guelma.dz

M. C. Ghanem is with Cyber Security Research Centre, London Metropolitan University, London, UK e-mail: ghanemm@staff.londonmet.ac.uk

reliability. In the inference stage, adversaries employ evasion attacks by manipulating the data during the operational phase. Their objective is to deceive a previously trained model by providing misleading inputs known as adversarial examples [11]. This deceptive data can lead to incorrect detection of cyber threats, as adversaries may employ zero-day attack techniques that mimic the behavior of adversarial examples and evade detection [12], [13].

Our study is focused on enhancing the effectiveness and resilience of FL-based cyber threat detection in the inference stage. We aim to address limited and non-IID data challenges and mitigate the threat landscape of zero-day and adversarial evasion attacks. In this context, we recognize the emergence of deep generative models (DGMs) as a promising approach that enhances data augmentation and enables robust optimization to effectively counter adversarial threats without predetermined assumptions about the capabilities of potential adversaries [11]. This study investigates the following research question: how generative models contribute to the effectiveness and resilience of DL-based IDS and explores the potential of federated generative models to address privacy concerns and challenges associated with imbalanced and non-IID data in the IIoT. Specifically, we propose a novel privacy-preserving and secure framework that leverages FL and generative adversarial networks (GANs) to secure industrial IoT networks. Our framework includes a three-model approach using 1) a federated generative model for data augmentation to limit the attack surface for potential zero-day and adversarial attacks. 2) A secondary model fine-tunes GAN-Critics for enhanced resilience and detection of various adversarial attacks, and 3) The classifier model focuses on precise cyber threat identification, leveraging augmented data for improved training under a synthetic FL schema, ultimately enhancing the efficiency and reliability of cyber threat detection.
Our contributions are as follows:

- We introduce a novel security framework that leverages federated learning and conditional-GAN approach (FedGenID) to augment distributed and multi-class cyber threats and ensure the security of IIoT networks. The framework consists of a three-model approach that utilizes a federated generative model, a local discriminator model, and a classifier model for efficient and robust cyber threat detection.
- Through meticulous analysis of the generated data, we propose a data curation method to align the generated data with the original data's constraints and traffic feature boundaries, ensuring consistency and reliability in the synthetic data.
- We thoroughly evaluate the efficiency of our proposed FedGenID framework with a new industrial IoTs network-based dataset (EdgeIIoTset 2022) for non-IID and multi-class cyber threat detection and robustness against zero-day and adversarial attacks. In addition, we demonstrate how FedGenID can mitigate the negative impact of differential privacy-enhanced FL on detection accuracy.

The remainder of this paper is organized as follows. Section II provides essential concepts for the proposed framework. We review related works in Section III. We overview our proposed framework in Section IV. Section V demonstrates experimental results and the effectiveness of the proposed framework. Finally, we conclude our work in Section VI.

II. Background

Generative models, including Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Autoregressive Models (ARMs) have been commonly applied to provide high-quality and diverse data to alleviate data scarcity in many application fields [14], [15]. While these models differ in their approach to generating new data, they all aim to capture the contextual representation of real data and produce high-quality samples. Recently, GANs have shown promise in their efficiency in generating high-quality and diverse data distribution and their potential application in data augmentation and privacy protection.

A. Deploying GANs for Cyber Threat Detection

To overcome challenges facing ML and DL-based cyber threat detection like privacy concerns, limited data availability, imbalanced data, and vulnerability to adversarial attacks, GANs have shown promise in addressing these challenges [14]. These models can produce high-quality new data similar to original training data, thereby addressing the issue of data availability and privacy protection by learning the underlying data distribution without memorizing sensitive individual data [15]. Furthermore, the adversarial training nature of GANs enhances the robustness and resilience of cyber threat detection, enabling models to defend against zero-day and emergent adversarial attacks. Although different GAN models employ different approaches to generate new data, their common objective is to capture the feature representation of real data and generate high-quality and diverse data samples. The GAN structure incorporates two deep learning models: the generator G generates new data similar to the training data, and the discriminator D differentiates between generated and original data. The training process of GAN can be formulated as a minimax game between G and D:

$$\underset{G}{min}\ \underset{D}{max} = [log(D(X)] + [log(1 - D(X')] \tag{1}$$

Where D tries to maximize the objective function by correctly classifying real and synthetic data, while G tries to minimize the objective function by deceiving the discriminator. However, GANs are computationally demanding [16] and suffer from various training issues, such as undesirable convergence properties that may lead to the mode collapse phenomenon, which occurs when the generator produces limited variants of the same set of samples, resulting in a lack of diversity in the generated data [17]. These concerns have led to researchers investigating other loss functions, model architectures, and training procedures to alleviate these issues.

B. Zero-day and Adversarial attacks in the field of cyber threat detection

Zero-day attacks exploit unknown vulnerabilities in software or hardware, consequently producing novel behaviors making them difficult to detect and identify, especially in scenarios where training data are scarce [18] Various methods have been proposed to simulate and detect zero-day attacks. For instance, some researchers have used GANs to simulate these attacks by generating similar but slightly different behaviors from known attack variants [19]–[21].

. Adversarial attacks, on the other hand, refer to techniques employed to generate false data inputs, known as adversarial examples, that closely resemble authentic data to deceive classification models when classifying them. In the context of cyber threat detection, cyber criminals may leverage these techniques to conduct complex cyber attacks that replicate the behavior of adversarial examples and evade detection by well-trained IDS classifier models [22]. Notably, these adversarial attacks may be generated intentionally or accidentally through software or hardware errors, raising concerns about the reliability of detection in real-world scenarios. Adversarial attacks can be categorized into black-box and white-box attacks. Black-box attacks involve limited knowledge about the target model's internal structure and parameters, while white-box attacks assume complete knowledge of the target model. Popular white-box attack techniques include the Fast Gradient Sign Method (FGSM), Basic Iterative Method (BIM), DeepFool, Carlini & Wagner attacks, and Jacobian-based Saliency Map Attack (JSMA). These attacks can cause misclassifications and compromise the reliability and security of ML and DL-based intrusion detection models [23]. Understanding and mitigating these attacks is crucial for enhancing the robustness and trustworthiness of DL-based cyber threat detection.

III. Related Works

The advent of generative adversarial networks (GANs) marks a promising breakthrough in the realm of DL applications due to their unique ability to generate synthetic data for augmentation and enable robust optimization. Consequently, researchers are now exploring their potential to address challenges related to cyber threat detection [11]–[13], [24]–[31]. In this section, we review previous studies of GAN application and their integration with the emergent FL training paradigm for cyber threat detection.

A. GAN Applications for Cyber Threat Detection

GANs introduced new possibilities in this field, either to treat data shortages or improve the resilience of IDS by detecting the zero-days and adversarial attacks that could deceive the detection module [11]. In [25], The authors proposed two variants of GAN models, an Encoder-GAN and a Bidirectional-GAN ()to detect and classify network attacks. The authors claimed the effectiveness of both methods in terms of classification metrics. However, their dataset has a limited set of features, and their approach can be computationally intensive, which could limit the effectiveness of the proposed model. Similarly, in [24], Kaplan et al. proposed two methods to improve the BiGAN training by adding extra steps for the generator. This includes minimizing the mean squared error between input and output, starting with a pre-trained generator, and enhancing the training process. In [26], Wu et al. proposed a deep convolutional GAN for intrusion detection. The authors addressed the limited resources of edge devices and proposed a feature reduction technique using the fuzzy method. Then, the GAN was proposed to augment the training data with synthetic samples and to optimize the discriminative CNN network for detecting various types of attacks. However, their method may not be able to detect some sophisticated or adversarial attacks that can evade detection, especially if the attackers can manipulate the network data or the generator. To detect data-tampering threats in the controller area network, Xie et al. [27] proposed an enhanced GAN discriminator. They implemented a traditional GAN model by feeding it improved attack data to supplement the insufficient training samples, consequently improving discriminator efficiency in detecting intrusions and data-tampering threats. The authors claimed their GAN model can generate more diverse and realistic attacked samples than existing methods. However, their model lacks rigorous analysis and guarantees on its convergence, stability, and generalization properties. Siniosoglou et al. [28] proposed a GAN architecture as an Auto-Encoder unified model for detecting anomalies and classifying attacks in a smart grid environment. The authors used the generator model as a decoder to produce synthetic samples and the discriminator model as an encoder to validate generated samples and detect and classify anomalies. The authors combine two different loss functions for this objective. However, they did not provide proof of training stability, convergence, and validity of generated data.

Although the above studies present compelling findings, they exhibit certain drawbacks. Notably, they tend to be computationally intensive, potentially limiting their applicability in IoT environments. An important challenge overlooked is privacy preservation, which is a significant concern in the IoT given the potential for data breaches. To address these issues, recent studies have also employed GANs training within the emerging federated learning framework (FL), taking advantage of its features for efficient computation and data privacy preservation.

B. Federated generative adversarial networks

In recent works, Zhang et al. [12] proposed a FL-GAN framework with a Mix-Generator module to handle Non-IID data issues at the edge. They divided the generator into two layers: the sharing layer extracts common features across all datasets, and the personalizing layer extracts unique features specific to each dataset. However, their approach is prone to GAN stability issues and poor generalization. In a related effort, Chuenbubpha et al. [13] introduced a federated GAN to address non-IID data distribution in FL. The authors trained a conditional GAN model to augment each client's local data with synthetic images per class and then started the FL process with augmented data for classification. However, their framework is computationally intensive and lacks privacy preservation because both GAN models are shared between clients. Similar in [32], Rasouli et al. studied federated GAN training (fedGAN) across non-IID sources, addressing privacy concerns. Their FedGAN uses local generators and discriminators at each source, periodically synchronizing them with a server node. The authors evaluated fedGAN's efficiency using benchmark image datasets, showing promising communication efficiency and synthetic data quality results. However, they did not consider adding noise or differential privacy mechanisms to enhance privacy protection. In contrast, [33], Xin et al. introduced a federated GAN for image generation with enhanced privacy, using a serial training method where each client updates the same model's parameters sequentially, adding noise to the discriminator's gradient to prevent sensitive information leakage. Although this is difficult and involves carefully selecting the differential privacy settings to balance privacy and GAN utility, the authors found that their proposed approach generates high-quality synthetic data. In [34], Li et al. trained a GAN model using the FL procedure for renewable scenario generation, using FL as a data privacy strategy. They proposed a least squares loss function to generate high-quality data, avoiding vanishing gradients and mode collapse problems. However, this framework assumes similar computing power and data distribution, which may not be feasible in realistic scenarios.

In the context of cyber threat detection, Tabassum et al. [29] trained federated GAN to tackle limited and unbalanced data on IoT devices. They trained generators and discriminators locally and synchronously, using gradient exchange and model updates. However, there's no loss function for evaluating the generator's output, and training the discriminator on a mix of data distributions might reduce classification performance. Additionally, sharing GAN models could expose client data. In a different study, Zang et al. [30] introduced a GAN-based approach as an attack model, aiming to generate artificial samples for the backdoor and label-flipping attacks. Notably, the authors utilized the exchanged updates between the server and clients to update a Discriminator architecture without prior knowledge of the training data. Based on this updated architecture, they successfully trained a generator. In [31], the authors introduced FL-GAN with differential privacy as an enhanced security and data privacy mechanism for IDS. Additionally, they proposed a Long Short-Term Memory (LSTM) architecture for both conditional GAN models, aiming to address the challenges posed by imbalanced and insufficient data samples in the context of effective IDS. However, they did not consider the validation of the consistency of the generated network traffic data.

Despite the promising potential demonstrated by previous studies on GANs, their application in securing Industrial IoTs against emerging cyber threats, including zero-days and adversarial attacks, is still in its early stages. Furthermore, the exploration of the feasibility of deploying privacy-preserving federated GANs, especially in handling imbalanced and non-IID, is not well discovered. Several research gaps necessitate attention. In addition, there is a notable absence of rigorous analysis regarding the consistency and validity of the synthetic IDS samples, questioning model accuracy in reflecting real cyber threat behaviors. Furthermore, an oversight of adversarial attack advancements, like white-box attacks, raises doubts about model robustness. This paper addresses these gaps by presenting an advanced Federated Generative IDS specifically tailored for the Industrial IoT. We propose a three-model framework employing a well-designed conditional GAN network, addressing non-IID data and adversarial attacks. At the same time, the third model focuses on precise zero-day and cyber threat identification.

IV. FedGenID

A. Overview

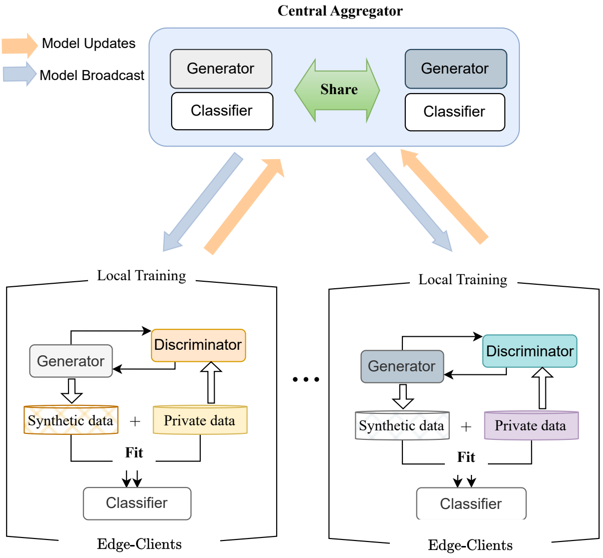


Fig. 1: The proposed Federated Generative Intrusion Detection System (FedGenID)

In this paper, we propose a novel framework for enhancing the efficiency and robustness of DL-based cyber threat detection named (FedGenID), through the incorporation of federated learning (FL) and Conditional Generative Adversarial Networks (cGANs). Figure 1 illustrates the workflow of our framework. Specifically, we employ FL to address privacy concerns and the computation efficiency of industrial IoTs, allowing models to train on distributed data locally on user devices while only exchanging model updates. In addition, we propose a generative framework to overcome limited data, imbalanced, and non-IID data challenges and enhance adversarial resilience, allowing robust and efficient cyber threat detection.

In this context, we design a three-model paradigm consisting of a federated generative model (i.e., cGAN Generator), a Discriminator model (i.e., cGAN Critic), and a Classifier model. The federated generative model generates (FGM) diverse artificial samples, the Discriminator (D) learns to distinguish between artificially generated and real samples, and the Classifier (C) trains on both original and artificially generated data for efficient and robust cyber threat identification. In the federation proceeding of FedGenID, we propose to share both the cGAN Generator and Classifier models between clients, where the cGAN Discriminator resides on the client side. This arrangement is motivated by the need to enhance the stability and privacy preservation of distributed GAN training, which is also susceptible to adversarial attacks. By leveraging the cGAN Discriminator locally, clients can detect and flag such adversarial attacks for further analysis. In addition, this will improve communication efficiency and privacy considerations of FL. By sharing the Generator, clients can generate diverse artificial samples locally and augment their local datasets, which helps identify zero-day and sophisticated adversarial attacks.

On the other hand, the global Classifier, shared between clients, undergoes updates that are also influenced by the artificial samples generated using the global Generator instead of only relying on local updates contributed by individual participants. This allows the classifier to train on large and diverse data sets. Consequently, the classifier would generalize and perform very well in identifying various attacks based on their characteristics, providing valuable insights for threat analysis and response. Moreover, this methodology aims to improve the overall resilience of the model and mitigate the potential risks associated with learning attacker-induced patterns from poisoned updates.

B. Conditional-GAN training procedure

Our training objective is to achieve an equilibrium point where the generator produces diverse and realistic samples. At the same time, the critic accurately distinguishes between real and generated data, providing meaningful feedback to the generator to produce samples that align with the specified condition (i.e., the target class label). Our implementation of the Conditional-GAN capitalizes on the capabilities of deep convolutional neural networks (CNNs) to effectively extract salient features from the conditioning input samples :

- **The Discriminator model** (D): depicted in Figure 2, and composed of four convolutional layers with a rectified linear unit (ReLU) activation function. It takes in both generated and real data samples and outputs the estimated Wasserstein distance between the fake and the real data distribution as a loss function for training objectives, providing improved feedback to the generator and guiding it to produce samples that closely resemble the real data distribution while matching the specified condition on target classes. In addition, D also performs fine-tuning for adversarial attack prediction in the post-GAN training phase. To achieve this, we incorporate a Dense layer that applies binary cross-entropy loss with Sigmoid function on its outputs to quantify the discrepancy between the predicted and ground truth values of real and generated data samples. By employing this approach, we aim to enhance the Critic's ability to discern and classify adversarial attacks effectively.
- **The Generator model** (G): depicted in Figure 2, and composed of four transposed convolutional layers with batch normalization and ReLU activation function. G takes in random samples picked from a uniform latent space denoted as $z \in \mathbb{R}^d$ where d is the dimension of the feature, along with a condition vector of class labels denoted as y. The aim is to produce the required labeled examples. The generator's output is passed through a Sigmoid activation function to map the generated features into normalized values between 0 and 1, according to the real data distribution.

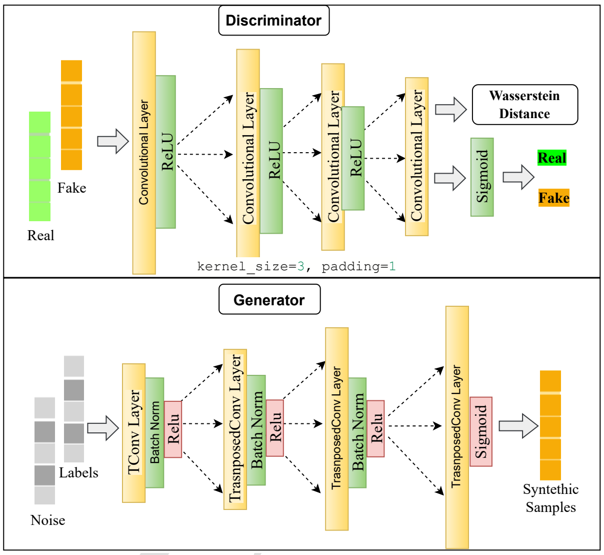


Fig. 2: The proposed three-model approach for efficient and robust cyber threat detection

- **The Classifier Model** (C): an independent CNN model designed explicitly for multi-class classification tasks. By leveraging augmented data for training, C effectively captures real-world data's intricate variations and complexities. Consequently, C demonstrates proficiency in identifying a wide range of attack classes, showcasing its robustness and resilience when manipulated with adversarial attempts.
- **Federated learning Objective**: The objective of the FL is to update the global Generator model, denoted as $\mathcal{G}$ and the global Classifier, denoted as $\mathcal{C}$, using K local models from corresponding clients. To achieve this, we employ an averaging algorithm, which can be expressed as follows:

$$\mathcal{G} \leftarrow \frac{1}{K}\sum_{k=1}^{K} G_k, \quad \mathcal{C} \leftarrow \frac{1}{K}\sum_{k=1}^{K} C_k$$

  Averaging allows for consolidating knowledge from multiple clients and collaborative learning in a distributed setting, enhancing model performance and generalization.
- **Local Training Objective**: The training objective of cGAN at the client side involves alternately updating the critic and the generator networks. We integrated the Wasserstein loss function to both models' goals [35], which represents the approximation functions that measure how closely generated and real data distributions are

based on how much one distribution needs to be moved to create the other. The goal is to prevent the generator from collapsing into one mode and ensure the generated samples are realistic. The Wasserstein loss is defined as follows:

$$\min_{G}\max_{D}\left(\mathbb{E}_{x\sim P_r}[D(x|y)] - \mathbb{E}_{z\sim P_z}[D(G(z|y))]\right) \tag{2}$$

where P_z represents the noise distribution and generates synthetic data samples. $D(\cdot|\cdot)$ the critic function, also known as the critic, which evaluates and distinguishes between real data samples x drawn from the real data distribution P_r and the generated samples produced by the generator function $G(\cdot|\cdot)$.

Intuitively, the critic aims to distinguish diverse real data from fake data conditioned on the given labels. At the same time, the generator tries to fool the critic by producing as realistic data as possible given the target labels. For better stability of cGAN, we added the gradient penalty (GP) to the previous loss (eq) as an approximation for enforcing the 1-Lipschitz continuity on the critic gradient norm to be one almost everywhere. The implementation of GP is as follows :

$$\min_{G}\max_{D}\left(2 + \lambda \cdot \frac{1}{n}\sum_{i=1}^{n}\left[\left\|\nabla_{\tilde{x_i}} D(\tilde{x_i}|y_i)\right\|_2 - 1\right]^2\right) \tag{3}$$

Where, λ is the hyper-parameter controlling the strength of the gradient penalty, $\tilde{x_i}$ is a sample randomly interpolated between real data x_i and generated data $G(z_i|y_i)$, and $\nabla_{\tilde{x_i}} D(\tilde{x_i}|y_i)$ represents the gradient of the critic's output concerning $\tilde{x_i}$.

Furthermore, for better performance, the critic independently applies the binary cross-entropy loss expressed as:

$$\min_{D}\left(\frac{1}{n}\sum_{i=1}^{n}\left[\log(D(x_i|1)) + \log(1 - D(G(z_i|y_i)|0))\right]\right) \tag{4}$$

Where $D(.|1)$ and $D(.|0)$ represent the D's prediction for the input data sample as real or fake, respectively, compared to the ground truth values (0,1).

On the other hand, for updating the classifier for multi-class classification, the objective can be formulated as:

$$\min_{C}\left(-\frac{1}{n}\sum_{i=1}^{n}\sum_{c=1}^{C} y_{i,c}\log(C(x_i))\right) \tag{5}$$

where C is the classifier, x_i is the augmented data sample, $y_{i,c}$ is the ground truth label for class c, and $C(x_i)$ is the predicted probability distribution over the classes.

TABLE I: Notation

Symbol	Description
K	A set of participating clients
I	Local iterations
E	Global epochs
m	Local batch size
α_G	Learning rate of Generator
α_D	Learning rate of Critic
λ	Penalty
$\mathcal{G}$	Global Generator
$\mathcal{D}$	Global Critic
P_z	Noise distribution
$D(\cdot\|\cdot)$	Critic function
$G(\cdot\|\cdot)$	Generator function
P_r	Real data distribution
x	Real data sample
z	Noise vector
y	Random label
$\tilde{x}$	Interpolated sample
$\nabla_{\tilde{x}} D(\tilde{x}\|y)$	Gradient of critic's output with respect to $\tilde{x}$
$\mathcal{L}_{\text{gen}}$	Generator loss
$\mathcal{L}_{\text{disc}}$	Critic loss

Algorithm 1 FedGenID : conditional-GAN Training

```
Require: a set of clients $\mathcal{K}$, Local iterations $I$, global epochs $E$, local batch size $m$, learning rate of Critic $\alpha_D$, learning rate of Generator $\alpha_G$, gradient penalty $\lambda$
Ensure: Trained Critic $\mathcal{D}$ and Generator $\mathcal{G}$
Initialize Generator $\mathcal{G}$ with random weights
for $r = 1$ to $R$ do
    Parallel. For $client\ k \in |\mathcal{K}|$
    for $t = 1$ to $E$ do
        Train Local Critic $\mathcal{D}_n$ on client $n$ using Alg. 3
        Train Local Generator $\mathcal{G}_n$ on client $n$ using Alg. 2
        Check convergence condition: if distance between fake and real predictions $\leq 0.1$ then break
    end for
    end
    Update Global Generator $\mathcal{G}$ by averaging local generators:
        $\mathcal{G} \leftarrow \frac{1}{|\mathcal{K}|}\sum_{n=1}^{|\mathcal{K}|}\mathcal{G}_n$
    return Trained Generator $\mathcal{G}$ to Clients
end for
```

C. FedGenID Complexity Analysis

While our proposed federated approach offers scalability, privacy, and distributed resource utilization crucial in IoT environments, we further formulate the computational and communication complexities of the proposed FedGenID approach against the traditional centralized approach. Our FedGenID incorporates both federated conditional GAN training (Algorithm 1) and the subsequent federated classifier training phases. We evaluate this process's resource requirements, scalability, and efficiency, considering both phases and the involvement of multiple IoT devices k.

- **Computation Complexity :**

$$\Theta\left(2I \cdot |m| \cdot (|\omega_{gi}| + |\omega_{di}| + IM) \cdot E \right.$$
$$\left. +2I_{\text{Cnn}} \cdot |m| \cdot |\omega_{\text{Cnn}}| \cdot E_{\text{Cnn}} \cdot K\right)$$

- **Communication Complexity :**

$$\Theta(I \cdot (|\omega_{gi}| + I_{\text{CNN}} \cdot |\omega_{\text{CNN}}| \cdot K)$$

Where, I represents the local iterations, $2I$ is for the forward and backward operations, m is the local batch size, $|\omega_{gi}|$ and $|\omega_{di}|$ are the sizes of the generator and discriminator parameter sets, IM accounts for floating-point operations, E is the total global training. Similarly, we add the complexity of the federated CNN-classifier training.

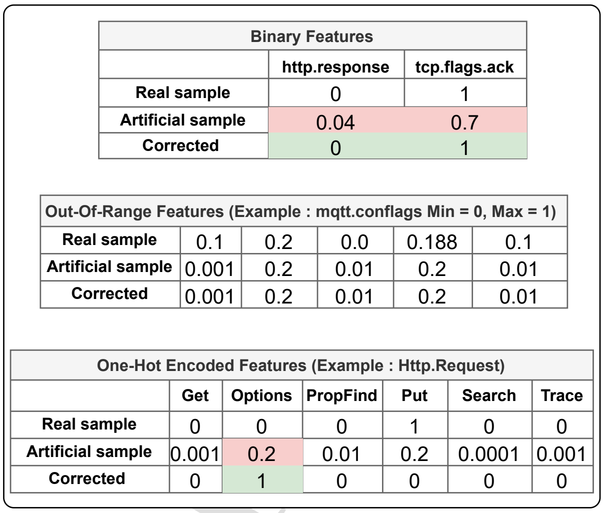
Algorithm 2 FedGenID: Local Generator Training

```
Require: Local iterations $I$, local batch size $m$, learning rate of Generator $\alpha_G$, penalty $\lambda$
Ensure: Trained Generator $\mathcal{G}$
Upload Generator $\mathcal{G}$ from Server
for $i = 1$ to $I$ do
    Sample $m$ noise vectors $\{z_1, z_2, \ldots, z_m\}$ from noise distribution $P_z$
    Sample $m$ random labels $\{y_1, y_2, \ldots, y_m\}$ from clients
    Generate synthetic samples: $\{G(z_1|y_1), G(z_2|y_2), \ldots, G(z_m|y_m)\}$
    Compute generator loss using Wasserstein loss:
        $\mathcal{L}_{\text{gen}} = \frac{1}{m}\sum_{i=1}^{m} D(G(z_i|y_i))$
    Update Generator weights using gradient descent:
        $\mathcal{G} \leftarrow \mathcal{G} - \alpha_G \cdot \nabla_{\mathcal{G}}\mathcal{L}_{\text{gen}}$
end for
return Trained Generator $\mathcal{G}$
```

Algorithm 3 FedGenID: Local Critic Training

```
Require: Local iterations $I$, local batch size $m$, learning rate of Critic $\alpha_D$, penalty $\lambda$
Ensure: Trained Critic $\mathcal{D}$
Initialize Critic $\mathcal{D}$ with random weights
for $i = 1$ to $I$ do
    Sample $m$ real data samples $\{x_1, x_2, \ldots, x_m\}$ from clients
    Sample $m$ noise vectors $\{z_1, z_2, \ldots, z_m\}$ from uniform distribution $P_z$
    Sample $m$ random labels $\{y_1, y_2, \ldots, y_m\}$ from clients
    Generate synthetic samples: $\{G(z_1|y_1), G(z_2|y_2), \ldots, G(z_m|y_m)\}$
    Sample $m$ random interpolation factors $\{\alpha_1, \alpha_2, \ldots, \alpha_m\}$ from a uniform distribution
    Compute interpolated samples: $\{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_m\} = \alpha_i x_i + (1 - \alpha_i)G(z_i|y_i)$
    Compute critic loss using Wasserstein loss with gradient penalty:
        $\mathcal{L}_{\text{disc}} = \frac{1}{m}\sum_{i=1}^{m}\Big[D(x_i|y_i) - D(G(z_i|y_i)) + \lambda \cdot \left(\left\|\nabla_{\tilde{x}_i} D(\tilde{x}_i|y_i)\right\|_2 - 1\right)^2\Big]$
    Update Critic weights using gradient descent:
        $\mathcal{D} \leftarrow \mathcal{D} - \alpha_D \cdot \nabla_{\mathcal{D}}\mathcal{L}_{\text{disc}}$
end for
```

Our approach optimizes resource efficiency by distributing training across multiple IoT devices, while a centralized approach may be limited by server capacity, especially for large-scale devices. However, our FedGenID may have higher communication overhead due to model parameters exchange during FL-cGAN and CNN learning phases. In contrast, the centralized approach may entail uploading a substantial amount of data to the centralized server for training.



Binary Features		
	http.response	tcp.flags.ack
Real sample	0	1
Artificial sample	0.04	0.7
Corrected	0	1

Out-Of-Range Features (Example : mqtt.conflags Min = 0, Max = 1)					
Real sample	0.1	0.2	0.0	0.188	0.1
Artificial sample	0.001	0.2	0.01	0.2	0.01
Corrected	0.001	0.2	0.01	0.2	0.01

One-Hot Encoded Features (Example : Http.Request)						
	Get	Options	PropFind	Put	Search	Trace
Real sample	0	0	0	1	0	0
Artificial sample	0.001	0.2	0.01	0.2	0.0001	0.001
Corrected	0	1	0	0	0	0

Fig. 3: Example of Data Curation for Artificial Network Traffic samples

D. Validity of generated traffic data

The data generated by the conditional GAN requires additional processing and validation to align with the constraints and traffic feature boundaries of the original data. Algorithm 4 aims to ensure the correctness of generated data that may contain errors or discrepancies, particularly in specific traffic feature categories. To address these issues, we consider features that contain out-of-range values, incorrect values for binary features, and incorrect values for one-hot encoded features. For out-of-range features, we identify samples where the synthetic data falls outside the valid range defined by the original data and clip their values to the real range. We rectify these values for binary features by rounding them to the nearest integer. Finally, for one-hot encoded features, the algorithm finds the index of the highest value in the one-hot encoded feature vector and sets all other values to 0. This approach can effectively guide researchers to address errors and discrepancies in synthetic data generated by GANs for network traffic data, enabling the generation of more consistent and reliable synthetic datasets for network-based cyber threat detection.

Figure 3 illustrates an example of cleaning artificial samples based on selected one-hot encoded features. For instance, a feature like ‘mqtt.conflags’ should only have values of 0 or 1, while a feature like ‘Http.Request’ should only be one of the six predefined categories. Furthermore, features within the range of real examples remain unchanged.

V. Performance evaluation and analysis

Our experiment with the proposed FedGenID security framework was conducted on Google Collaboratory using PyTorch and Tesla-T4 GPU accelerators. We equipped participating clients with non-iid datasets, as demonstrated in Table. IV. We initially established the federated cGAN and proceeded

Algorithm 4 Curation of Generated Data

```
Require: Original data $O$ with $n$ instances and $d$ features, synthetic data $S$ with the same shape as $O$ and $d$ features, indices $R$ of features that need to be corrected for out-of-range values, indices $B$ of binary features that need to be corrected for incorrect values, indices $C$ of one-hot encoded features that need to be corrected for incorrect values
Ensure: Corrected synthetic data $S'$ with the same shape as $S$
procedure CORRECTDATA($O, S, R, B, C$)
    $S' \leftarrow S$    ▷ Create a copy of synthetic data
    for $i \in R$ do    ▷ Correct out-of-range values
        $v_{\min,i} \leftarrow \min(O_{:,i})$
        $v_{\max,i} \leftarrow \max(O_{:,i})$
        $S'_{:,i} \leftarrow \max(\min(S_{:,i}, v_{\max,i}), v_{\min,i})$
    end for
    for $i \in B$ do    ▷ Correct binary values
        $S'_{\text{incorrect},i} \leftarrow (S_{:,i} \neq 0) \wedge (S_{:,i} \neq 1)$    ▷ Identify incorrect values
        $S'_{\text{corrected},i} \leftarrow \lfloor S_{\text{incorrect},i} \rceil$    ▷ Round incorrect values to nearest integer
        $S'_{\text{corrected},i} \leftarrow S_{\text{corrected},i} \cdot S'_{\text{incorrect},i} + S_{:,i} \cdot (\neg S'_{\text{incorrect},i})$    ▷ Replace incorrect values with corrected values
    end for
    for $i \in C$ do    ▷ Correct one-hot encoded values
        $h_i \leftarrow \text{argmax}(S_{:,i})$    ▷ Find index of highest value
        $S'_{:,i} \leftarrow \boldsymbol{e}_{h_i}$    ▷ Set all but highest value to 0
    end for
    return $S'$    ▷ Return corrected synthetic data
end procedure
```

with training. Subsequently, we leverage the federated generative model to enhance the training of the federated classifier model by providing augmented data. After that, we introduced deferential privacy training [8] for the global classifier model and evaluated how augmented data in synthetic FL alleviated the negative effects of DP and improved robustness against zero-day attacks.

Moreover, it is essential to note that each client maintains its critic model, which is utilized as a discriminator for detecting adversarial examples. Details regarding the experimental settings and learning parameters employed in this study can be found in Table II. Figure 4 illustrates the flowchart of our proposed framework for robust and resilient cyber threat detection using a distributed framework. To evaluate the impact of security constraints on the learning process, we have employed various metrics to evaluate both detection efficiency and effectiveness. These metrics include Accuracy, Precision, Detection Rate, false positive rate and false negative rate. By analyzing these measures, we aim to gain insights into the performance and robustness capabilities of our proposed framework for detecting zero-day cyber threats. Furthermore, we aim to understand the influence of security constraints, including distributed learning and DP training, on its efficacy.

- *Accuracy (Acc)*: given by:

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \quad \text{where :} \tag{6}$$

  TP: refers to the count of negative samples that are accurately classified.
  TN: refers to the count of negative samples that are accurately classified.
  FP: refers to the count of positive samples that are incorrectly categorized.
  FN: refers to the count of negative samples that are incorrectly categorized.
- *Precision (Pr)*: denotes the proportion of proper attack classifications (TP) attack predictions to the total amount of predicted attack results and given by :

$$Pr = \frac{TP}{TP + FP} \tag{7}$$

- *Detection rate (Dr)*: denotes the proportion of proper attack classifications (TP) relative to the overall count of all samples that ought to have been identified as attacks and given by :

$$Dr = \frac{TP}{TP + FN} \tag{8}$$

- *False positive rate* (FPR) represents the proportion of incorrectly categorized negative samples (FP) relative to the overall count of all samples that should have been classified as negatives. It is calculated using the following formula:

$$FPR = \frac{FP}{FP + TN} \tag{9}$$

- *False negative rate* (FNR): represents the proportion of incorrectly categorized positive samples (FN) relative to the overall count of all samples that should have been classified as positives. It is calculated using the following formula:

$$FNR = \frac{FN}{FN + TP} \tag{10}$$

TABLE II: Experimental settings

	Parameter	Values
Federated cGAN	cGAN Generator	Refer to 2
	cGAN Critic	Refer to 2
	Local cGAN epochs	10
	Critic repeats for one epoch	2
	Learning rate	0.0002
	Local Batch_size	32
	Global rounds	5
Federated Classifier	Classifier	CNN 15-class
	Local Batch_size	64
	Global rounds	15
	Learning rate	0.001
Differential privacy	Epsilon (ϵ)	1
	Delta (δ)	1.5e-5
	Gradient norm bound (C)	1.2
*	Optimizer	Adam

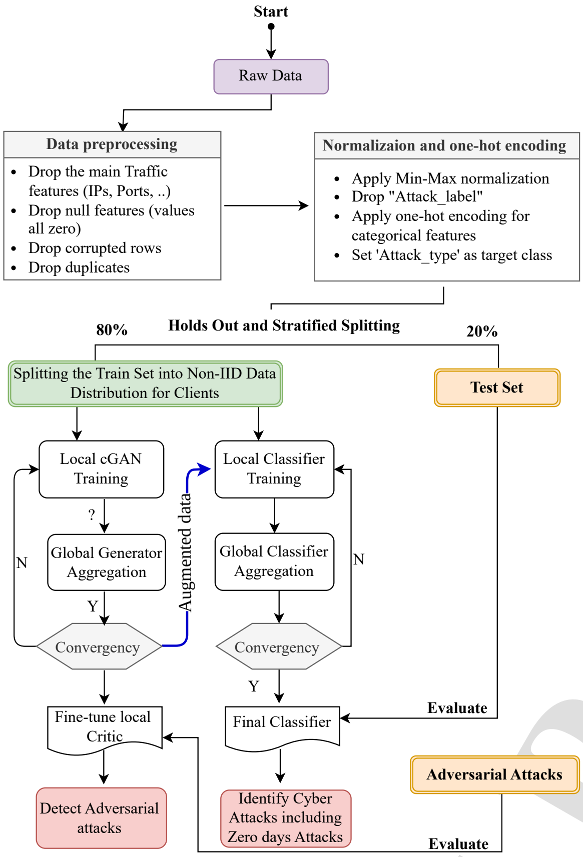

Start
Raw Data
Data preprocessing
Drop the main Traffic features (IPs, Ports, ..)
Drop null features (values all zero)
Drop corrupted rows
Drop duplicates
Normalizaion and one-hot encoding
Apply Min-Max normalization
Drop "Attack_label"
Apply one-hot encoding for categorical features
Set 'Attack_type' as target class
80%
Holds Out and Stratified Splitting
20%
Splitting the Train Set into Non-IID Data Distribution for Clients
Test Set
Local cGAN Training
Local Classifier Training
?
Augmented data
N
Global Generator Aggregation
Global Classifier Aggregation
N
Y
Convergency
Convergency
Y
Evaluate
Fine-tune local Critic
Final Classifier
Adversarial Attacks
Detect Adversarial attacks
Identify Cyber Attacks including Zero days Attacks
Evaluate

Fig. 4: Flowchart of FedGenID framework aggregation, training and evalution

TABLE III: Edge-IIoTset Data distribution.

Classes	Original Train Count	Original Test Count
Normal	1046926	323129
Backdoor	19890	4972
Vulnerability_scanner	40088	10022
DDoS_ICMP	93149	23287
Password	40122	10031
Port_Scanning	18051	4513
DDoS_UDP	88027	22007
Uploading	30107	7527
DDoS_HTTP	39929	9982
SQL_injection	40962	10241
Ransomware	8740	2185
DDoS_TCP	40050	10012
XSS	12732	3183
MITM	320	80
Fingerprinting	801	200

A. Data Selection and Processing:

Our proposed framework uses the new Edge-IIoTset [6], which exhibits characteristics of both imbalanced and non-IID datasets. This dataset comprises fourteen labeled network attacks. Our data preprocessing involved removing duplicates and handling missing values by dropping instances with 'NAN' or 'INF' values. Additionally, irrelevant traffic features like IP addresses and payload information (e.g., frame.time, ip.src_host, ip.dst_host, arp.src.proto_ipv4, arp.dst.proto_ipv4, http.file_data, http.request.full_uri, icmp.transmit_timestamp, http.request.uri.query, tcp.options, tcp.payload, tcp.srcport, tcp.dstport, udp.port, mqtt.msg) were excluded. Categorical features such as 'http.request.method', 'http.referer', 'http.request.version', 'dns.qry.name.len', 'mqtt.conack.flags', 'mqtt.protoname', and 'mqtt.topic' were encoded using one-hot encoding, resulting in a total of 95 features. The data was then standardized using min-max scaling. Finally, a hold-out split strategy was applied to partition the data into Training and Test Sets. The initial distribution of the dataset is depicted in Table III.

To emulate the distribution heterogeneity and nature in FL, we divided the training set into non-IID partitions and allocated them to ten clients. For this, we implemented a label partition method, ensuring that each client has a random subset of labels with the same feature vector of training data, assuming that each client has partial knowledge of the total classes involved in the problem, as demonstrated in Table IV.

B. FedGenID: Federated cGAN Training

A series of comprehensive experiments were conducted to uncover the ideal hyperparameter setup for training stability of our designed federated cGAN scheme. Our findings reveal that utilizing several local epochs with fewer federated rounds improves stability. Figures 5 demonstrate the local training loss of Federated cGAN using the Wasserstein distance with gradient penalty (Wass-GP) reported in predetermined training steps. Both cGAN models are directly related to the Wasserstein distance, where the Critic loss represents the approximate negative of the Wasserstein distance. As demonstrated, unlike regular loss functions, Wass-GP is unbounded and can output any number. This feature enhances the critic without suffering from the vanishing gradient problem. We can see that the critic's loss starts at a relatively high value and gradually decreases over time. This signifies an improvement in the Critic's ability to distinguish between actual and generated samples. Conversely, the generator loss starts at a lower value and slightly increases over time. This can be attributed to the enhanced performance of the Critic, which poses a more challenging adversarial objective for the generator. Notably, as the training progresses, a convergence pattern becomes evident, where the losses associated with both the generator and Critic tend to approach each other and ideally converge.

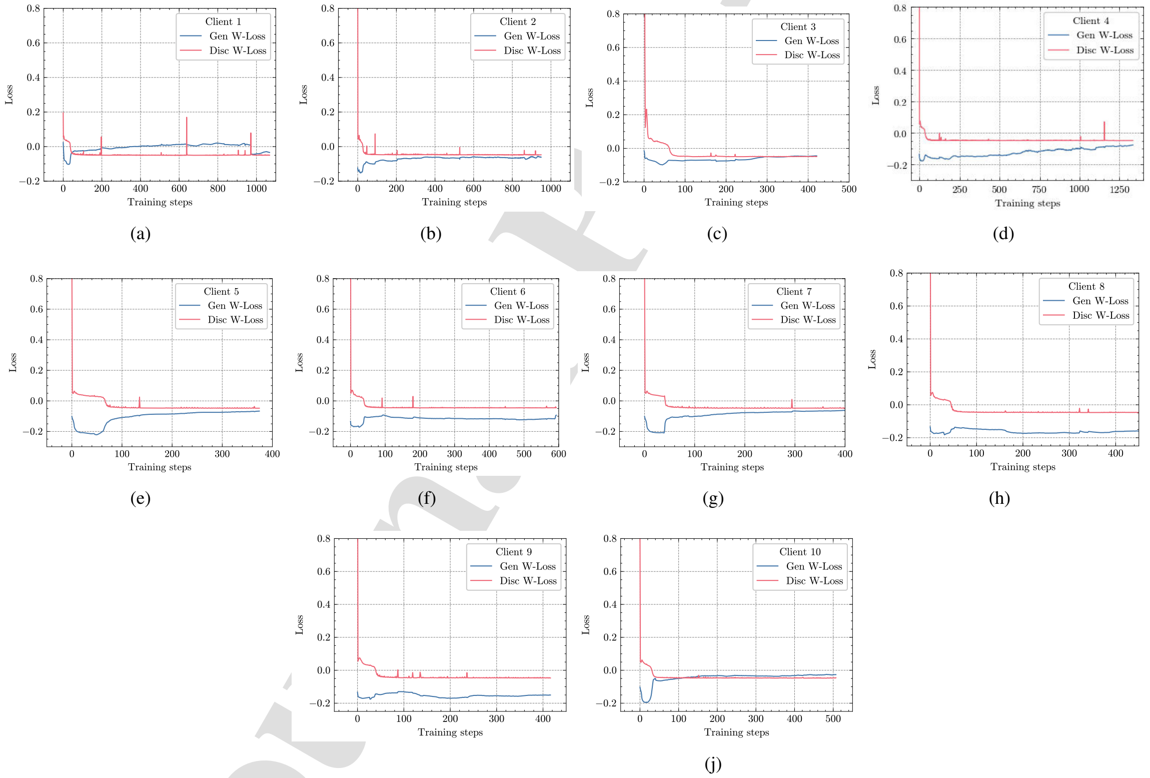
C. FedGenID: Adversarial Attack Detection

The resilience of IDS to sophisticated adversarial attacks is a critical aspect often overlooked. Relying on a single model to defend against all adversarial and zero-day attacks presents a potential vulnerability. To enhance the resilience and adaptability of our framework against continually evolving adversarial attacks, we enhance the capability of local critics to detect adversarial examples by adjusting their decision threshold by applying the Sigmoid function. It is noteworthy that Critic models were trained using the Wasserstein loss that maximizes the distance between real and fake inputs. Therefore, if we applied an activation function, we could

TABLE IV: Non-IID Data Distribution

Client 1		Client 2		Client 3		Client 4		Client 5	
Attack classes	Count	Attack classes	Count	Attack classes	Count	Attack classes	Count	Attack classes	Count
Normal	5914	Normal	8420	Normal	5145	Normal	2902	Normal	4340
Backdoor	3023	Backdoor	4066	Backdoor	2600	Vul_scanner	4274	Backdoor	2215
DDoS_ICMP	14121	Vul_scanner	10069	Vul_scanner	6080	DDoS_ICMP	14369	DDoS_UDP	12777
Password	6031	Uploading	7422	DDoS_UDP	15119	DDoS_UDP	8500	Uploading	3808
Port_Scan	2659	SQL_injection	10164	Uploading	4597	Uploading	2696	DDoS_HTTP	7154
DDoS_UDP	13223	Ransomware	2121	SQL_injection	6293	DDoS_TCP	2841	SQL_injection	5308
DDoS_HTTP	6016	DDoS_TCP	8401	DDoS_TCP	5200			DDoS_TCP	4428
DDoS_TCP	6009								

Client 6		Client 7		Client 8		Client 9		Client 10	
Attack classes	Count	Attack classes	Count	Attack classes	Count	Attack classes	Count	Attack classes	Count
Normal	4862	Normal	1364	Normal	1143	Normal	892	Normal	5018
Backdoor	2952	Vul_scanner	2020	Backdoor	843	Backdoor	659	Backdoor	3532
Vul_scanner	7395	DDoS_ICMP	10430	DDoS_ICMP	9041	Vul_scanner	1628	Ransomware	4680
Uploading	4340	Port_Scanning	2566	Port_Scanning	2177	DDoS_ICMP	7122	XSS	12732
SQL_injection	7210	DDoS_UDP	6195	DDoS_UDP	5278	Password	5255	MITM	320
DDoS_TCP	4870	Uploading	1140	Uploading	985	DDoS_UDP	4209	Fingerprinting	801
		SQL_injection	1923	DDoS_HTTP	4398	SQL_injection	1610		
		Ransomware	1100	DDoS_TCP	1126	Ransomware	839		
		DDoS_TCP	1376						



(a) (b) (c) (d)

(e) (f) (g) (h)

(j)

Fig. 5: Local cGAN training: Loss vs Training Steps

predict adversarial examples and evaluate them against a corresponding ground truth value. Nevertheless, our experiments revealed that our cGAN-Critic models exhibited limitations in generalization to other persistent adversarial inputs, thereby reducing the practicality of the defense mechanism in real-world scenarios. To address this issue, we further fine-tune the critic models using data from the global generator and data from more sophisticated attack methods to increase the adversarial diversity. To this end, we refined our approach by fine-tuning each local cGAN-Critic. Specifically, we added a linear layer and trained using genuine data from the clients' datasets in combination with data from the global generator

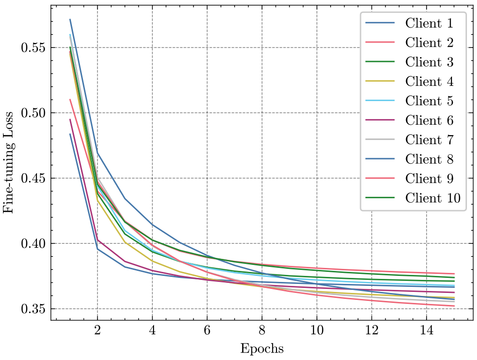


Fig. 6: Fine-tuning Loss for Clients Critic for adversarial attack detection

TABLE V: Performance Comparison of Proposed individual Detector against the three evaluated adversarial attacks

Attacks	Clients	Accuracy %	DR %	FPR %
FGSM	Worst client : 2	92.05	98.64	15.76
	Best client 8	96.74	97.29	04.57
BIM	Worst client : 10	92.97	98.96	18.52
	Best client : 9	98.04	98.92	04.01
DeepFool	Worst client : 2	92.12	98.82	15.76
	Best client: 9	98.79	100	04.01

and more sophisticated adversarial inputs obtained from the FGSM attack. Figure 6 illustrates the fine-tuning history of Clients Critic for 15 epochs. The results are reported in Table V.

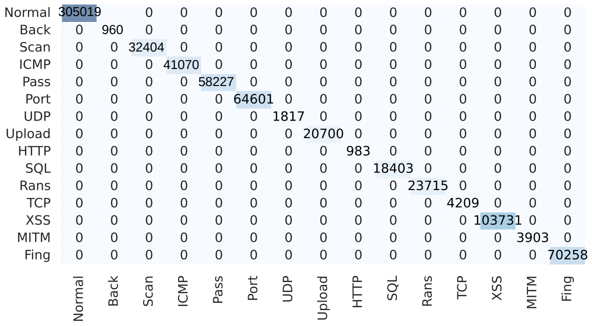
D. *FedGenID: Zero-day Attack Detection*

We expand our framework's robustness evaluation against zero-day attacks, addressing the dynamic nature of these threats. Our non-IID setup simulates the unpredictability of these threats by excluding specific attack classes from certain clients' datasets. Additionally, we simulate these attacks by augmenting the TestSet to include variations and novel instances, leveraging the global generator. We denote the combined original TestSet and the generated zero-day attack samples as the Augmented TestSet. We removed any duplicate records to ensure its integrity and labeled the generated zero-day attack samples with their corresponding known attack labels. Furthermore, our data curation approach ensures that the samples are realistic and represent the real data distribution. Figure 8 and Table VIII demonstrate performance results in detecting and identifying Zero-day attacks.

E. *FedGenID: Numerical results*

Figure 7 demonstrates the class distribution of generated augmented train data using our proposed federated generative model (FGM). Notably, our FedGenID incorporates class-conditioned labels, which; although not immune to ensuring label accuracy, significantly enhances data diversity. Our investigation produced a dataset comprising 50,000 instances for each distinct attack class. However, following the application of our data curation methodology, which introduces marginal modifications to feature values, a mismatch was detected between the initially specified target classes and the resulting predicted labels upon employing a well-trained DL classifier. In the scope of our research, we proceed with this labeling technique using the DL classifier with 96% accuracy on the original train data to rectify the labeling discrepancies. However, it is worth noting that techniques such as self-supervised learning could be investigated in prospective studies.

The results demonstrate that the approach successfully captures the underlying patterns and features of classes such as Normal, XSS, Fingerprinting, Portscanning, and Password, as indicated by their relatively high sample counts (Figure 7). These results highlight the Wasserstein conditional GAN's ability to generate synthetic data that faithfully exhibits the distinct characteristics associated with each class. However, it is worth noting that certain classes, including Backdoor, HTTP, and DDoS_UDP, exhibit relatively low counts, suggesting the presence of fewer distinctive patterns or features, posing challenges for an accurate generation. Nevertheless, by integrating these generated samples into the local training process of participating clients, we aim to enhance robustness and classification efficiency against adversarial and zero-day cyber attacks.



	Normal	Back	Scan	ICMP	Pass	Port	UDP	Upload	HTTP	SQL	Rans	TCP	XSS	MITM	Fing
Normal	305019		0	0	0	0	0	0	0	0	0	0	0	0	0
Back	0	960	0	0	0	0	0	0	0	0	0	0	0	0	0
Scan	0	0	32404	0	0	0	0	0	0	0	0	0	0	0	0
ICMP	0	0	0	41070	0	0	0	0	0	0	0	0	0	0	0
Pass	0	0	0	0	58227	0	0	0	0	0	0	0	0	0	0
Port	0	0	0	0	0	64601	0	0	0	0	0	0	0	0	0
UDP	0	0	0	0	0	0	1817	0	0	0	0	0	0	0	0
Upload	0	0	0	0	0	0	0	20700	0	0	0	0	0	0	0
HTTP	0	0	0	0	0	0	0	0	983	0	0	0	0	0	0
SQL	0	0	0	0	0	0	0	0	0	18403	0	0	0	0	0
Rans	0	0	0	0	0	0	0	0	0	0	23715	0	0	0	0
TCP	0	0	0	0	0	0	0	0	0	0	0	4209	0	0	0
XSS	0	0	0	0	0	0	0	0	0	0	0	0	103731	0	0
MITM	0	0	0	0	0	0	0	0	0	0	0	0	0	3903	0
Fing	0	0	0	0	0	0	0	0	0	0	0	0	0	0	70258

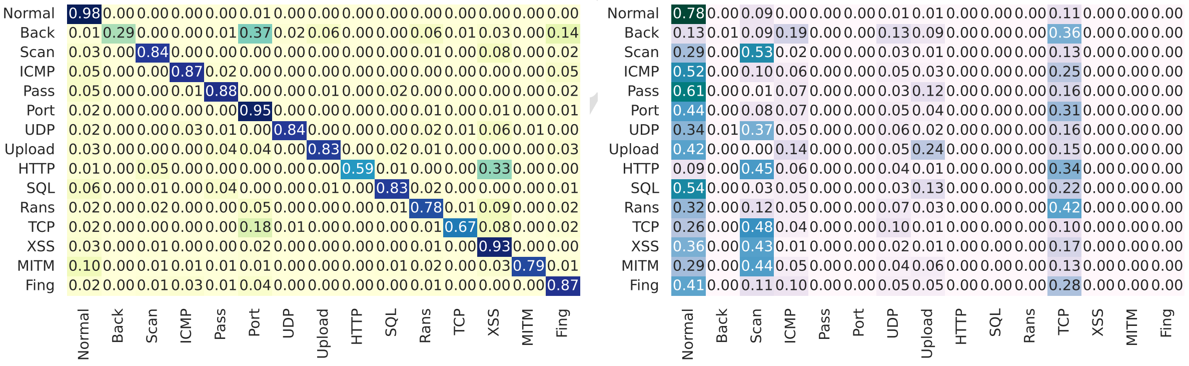
Fig. 7: Confusion matrix depicting the class distribution of generated traffic. The classes are labeled using the FedID classifier

After the fine-tuning of clients' critics, table V presents a performance comparison of the proposed individual detector against three evaluated sophisticated adversarial attacks. The results demonstrate that individual critics exhibit varying performance levels against different adversarial attacks, with some clients achieving higher accuracy and better detection rates while keeping false positive rates relatively low. For instance, in the case of the FGSM attack, the best-performing client (Client 8) achieved a detection rate of 97.29% and a false positive rate of 4.57%. Conversely, the worst-performing client (Client 2) achieved a false positive rate of 15.76%. Client 9 generally emerges as the top-performing client across the evaluated attacks, displaying impressive detection rates and accuracy. These findings emphasize the potential of the proposed fine-tuning individual critics to discern sophisticated adversarial attacks effectively instead of relying on a single

TABLE VI: Per-class performance using different evaluation aspects

		Original TestSet				Augmented TestSet			
Classes	Metrics	Precision		Detection rate		Precision		Detection rate	
	Settings	FedID	FedGenID	FedID	FedGenID	FedID	FedGenID	FedID	FedGenID
Normal	No-DP	1.00	0.99	1.00	1.00	0.91	0.99	0.96	1.00
	DP	1.00	1.00	1.00	1.00	0.86	1.00	0.99	1.00
Backdoor	No-DP	0.63	0.65	0.96	0.98	0.62	0.65	0.93	0.96
	DP	0.72	0.73	0.89	0.89	0.72	0.73	0.86	0.89
Vulnerability_scan	No-DP	0.89	0.60	0.70	0.98	0.33	0.60	0.64	0.92
	DP	0.58	0.49	0.94	0.99	0.57	0.49	0.58	0.99
DDoS_ICMP	No-DP	1.00	0.97	1.00	1.00	0.83	0.97	0.76	0.97
	DP	0.97	1.00	0.98	0.99	0.90	1.00	0.73	0.99
Password	No-DP	0.00	0.94	0.00	0.07	0.00	0.94	0.00	0.50
	DP	0.00	1.00	0.00	0.07	0.00	1.00	0.00	0.07
Port_Scanning	No-DP	0.00	0.86	0.00	0.00	0.00	0.86	0.00	0.70
	DP	0.00	0.58	0.00	0.03	0.14	0.58	0.00	0.03
DDoS_UDP	No-DP	0.98	1.00	1.00	1.00	0.83	1.00	0.98	0.99
	DP	0.96	0.97	1.00	1.00	0.96	0.97	0.98	1.00
Uploading	No-DP	0.54	0.76	0.39	0.38	0.36	0.76	0.34	0.54
	DP	0.45	0.57	0.42	0.37	0.17	0.57	0.41	0.37
DDoS_HTTP	No-DP	0.64	0.79	0.97	0.30	0.64	0.79	0.95	0.31
	DP	0.75	0.87	0.52	0.25	0.49	0.87	0.51	0.25
SQL_injection	No-DP	0.41	0.48	0.90	0.91	0.41	0.48	0.65	0.89
	DP	0.40	0.41	0.82	0.90	0.40	0.41	0.59	0.90
Ransomware	No-DP	0.00	0.85	0.00	0.11	0.00	0.85	0.00	0.57
	DP	0.00	0.30	0.00	0.06	0.00	0.30	0.00	0.06
DDoS_TCP	No-DP	0.71	0.71	0.99	0.99	0.25	0.71	0.92	0.97
	DP	0.69	0.69	1.00	1.00	0.57	0.69	0.92	1.00
XSS	No-DP	0.00	0.92	0.00	0.03	0.00	0.92	0.00	0.81
	DP	0.99	1.00	0.02	0.02	0.56	1.00	0.01	0.02
MITM	No-DP	0.00	0.92	0.00	1.00	0.00	0.92	0.00	0.81
	DP	0.00	1.00	0.00	1.00	0.00	1.00	0.00	1.00
Fingerprinting	No-DP	0.00	0.90	0.00	0.00	0.00	0.90	0.00	0.86
	DP	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.00

FedID: Federated Intrusion detection; **FedGenID** : Federated Generative Intrusion detection;
No-DP : No differentially private training; **DP** : with differentially private training.



	Normal	Back	Scan	ICMP	Pass	Port	UDP	Upload	HTTP	SQL	Rans	TCP	XSS	MITM	Fing
Normal	0.98	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Back	0.01	0.29	0.00	0.00	0.01	0.37	0.02	0.06	0.00	0.00	0.06	0.01	0.03	0.00	0.14
Scan	0.03	0.00	0.84	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.08	0.00	0.02
ICMP	0.05	0.00	0.00	0.87	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05
Pass	0.05	0.00	0.00	0.01	0.88	0.00	0.00	0.01	0.00	0.02	0.00	0.00	0.00	0.00	0.02
Port	0.02	0.00	0.00	0.00	0.00	0.95	0.00	0.00	0.00	0.00	0.01	0.00	0.01	0.00	0.01
UDP	0.02	0.00	0.00	0.03	0.01	0.00	0.84	0.00	0.00	0.00	0.02	0.01	0.06	0.01	0.00
Upload	0.03	0.00	0.00	0.00	0.04	0.04	0.00	0.83	0.00	0.02	0.01	0.00	0.00	0.00	0.03
HTTP	0.01	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.59	0.01	0.00	0.00	0.33	0.00	0.00
SQL	0.06	0.00	0.01	0.00	0.04	0.00	0.00	0.01	0.00	0.83	0.02	0.00	0.00	0.00	0.01
Rans	0.02	0.00	0.02	0.00	0.00	0.05	0.00	0.00	0.00	0.01	0.78	0.01	0.09	0.00	0.02
TCP	0.02	0.00	0.00	0.00	0.00	0.18	0.01	0.00	0.00	0.00	0.01	0.67	0.08	0.00	0.02
XSS	0.03	0.00	0.01	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.01	0.00	0.93	0.00	0.00
MITM	0.10	0.00	0.01	0.01	0.01	0.01	0.00	0.00	0.00	0.01	0.02	0.00	0.03	0.79	0.01
Fing	0.02	0.00	0.01	0.03	0.01	0.04	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.87

(a) Federated Generative Intrusion Detection (FedGenID)

	Normal	Back	Scan	ICMP	Pass	Port	UDP	Upload	HTTP	SQL	Rans	TCP	XSS	MITM	Fing
Normal	0.78	0.00	0.09	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.11	0.00	0.00	0.00
Back	0.13	0.01	0.09	0.19	0.00	0.00	0.13	0.09	0.00	0.00	0.00	0.36	0.00	0.00	0.00
Scan	0.29	0.00	0.53	0.02	0.00	0.00	0.03	0.01	0.00	0.00	0.00	0.13	0.00	0.00	0.00
ICMP	0.52	0.00	0.10	0.06	0.00	0.00	0.05	0.03	0.00	0.00	0.00	0.25	0.00	0.00	0.00
Pass	0.61	0.00	0.01	0.07	0.00	0.00	0.03	0.12	0.00	0.00	0.00	0.16	0.00	0.00	0.00
Port	0.44	0.00	0.08	0.07	0.00	0.00	0.05	0.04	0.00	0.00	0.00	0.31	0.00	0.00	0.00
UDP	0.34	0.01	0.37	0.05	0.00	0.00	0.06	0.02	0.00	0.00	0.00	0.16	0.00	0.00	0.00
Upload	0.42	0.00	0.00	0.14	0.00	0.00	0.05	0.24	0.00	0.00	0.00	0.15	0.00	0.00	0.00
HTTP	0.09	0.00	0.45	0.06	0.00	0.00	0.04	0.01	0.00	0.00	0.00	0.34	0.00	0.00	0.00
SQL	0.54	0.00	0.03	0.05	0.00	0.00	0.03	0.13	0.00	0.00	0.00	0.22	0.00	0.00	0.00
Rans	0.32	0.00	0.12	0.05	0.00	0.00	0.07	0.03	0.00	0.00	0.00	0.42	0.00	0.00	0.00
TCP	0.26	0.00	0.48	0.04	0.00	0.00	0.10	0.01	0.00	0.00	0.00	0.10	0.00	0.00	0.00
XSS	0.36	0.00	0.43	0.01	0.00	0.00	0.02	0.01	0.00	0.00	0.00	0.17	0.00	0.00	0.00
MITM	0.29	0.00	0.44	0.05	0.00	0.00	0.04	0.06	0.00	0.00	0.00	0.13	0.00	0.00	0.00
Fing	0.41	0.00	0.11	0.10	0.00	0.00	0.05	0.05	0.00	0.00	0.00	0.28	0.00	0.00	0.00

(b) Federated Intrusion Detection (FedID)

Fig. 8: The effectiveness of FedGenID over FedID in the context of zero-day attacks

model to defend against all attacks. Furthermore, our design differs from conventional methods by utilizing another classifier detection model to investigate undetected adversarial inputs. Consequently enhancing the overall system robustness.

Figure 9 illustrates a comparative analysis of validation accuracy between FedGenID and FedID over time, considering both scenarios with and without DP. The evaluation is performed on the Original Real-TestSet. While our synthetic FL approach offers a degree of privacy preservation, the introduction of DP training shows promise for further enhancing privacy protection despite its potential negative impact on model performance. Our results demonstrate that incorporating DP incurs a training overhead for both frameworks, with FedGenID displaying a comparatively lower increase in computational time attributable to its data augmentation approach. Furthermore, our analysis reveals that FedGenID achieves performance levels nearly comparable to those of FedID without DP while outperforming it under DP training conditions. This underscores the potential of FedGenID as a more cost-effective solution when considering privacy

TABLE VII: Comparison with related GAN-based IDS

Study	Main Idea	**Dataset**			Features			
		Name	Settings	Features Diversity	Federated Learning	Differential Privacy	Adversarial Attacks	Data Curation
Kaplan et al. 2020 [24]	Two methods for improving BiGAN training include minimizing the mean squared error between the generator input and output and starting with a pre-trained generator	KDD99	IID	Low	No	No	No	No
Aabdalgawad et al. 2021 [25]	BiGAN model to detect unknown anomalies, as a form of defense against novel or zero-day	IoT-23	IID	Low	No	No	No	No
Wu et al. 2021 [26]	Feature reduction and a deep convolutional GAN, addressing limited resources and optimizing the discriminative CNN network with synthetic samples.	CIC-DDOS2018 CIC-DDOS2019	IID	Medium	No	No	No	No
Xie et al. 2021 [27]	a GAN-discriminator to detect potential data tampering threats in controller area networks by incorporating enhanced synthetic attack data.	N/A	N/A	Medium	No	No	No	No
Tabassum et al. 2022 [29]	Federated GAN for IoT devices using gradient exchange and model updates.	KDD99, NSL-KDD, UNSW-NB15	IID	Medium	Yes	No	No	No
Gu et al. 2023 [36]	Self-attention WGAN, and focal loss DNN to improve the detection performance of rare and unknown attack	NSL-KDD, CIC-IDS-2018	IID	Medium	No	No	No	No
He et al. 2022 [31]	FL-GAN with differential privacy for enhanced security and data privacy in IDS	CIC-IDS2017	IID	Medium	Yes	Yes	No	No
Meenakshi et al. 2024 [37]	a self-attention-based conditional variational auto-encoder GAN, combining advanced techniques to adapt to network dynamics, and accurately identify intusions	WSN-DS	IID	Medium	No	No	No	No
Our Study	FL-GAN with a 3-model IDS framework comprising a generative model, local discriminator, and classifier models, for robust detection of zero-day and adversarial attacks	EdgeIIoTSet2022	Non-IID	High	Yes	Yes	FGSM, BIM, DeepFOOL	Yes

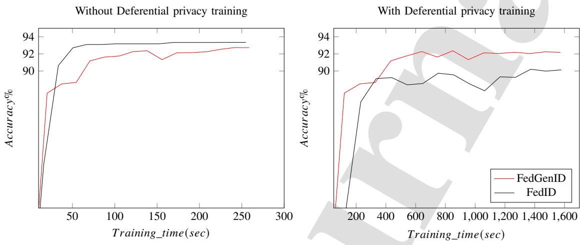


Fig. 9: An Examination of Validation Accuracy in FedGenID Compared to Standalone FedID with and without Differential Privacy Training on the Original Test Data

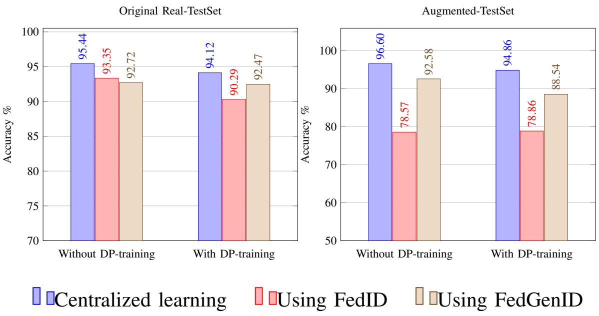


Fig. 10: Comparative analysis of cyber threat detection performance and robustness using our proposed FedGenID and Standalone FedID

preservation. Notably, even when both strategies are combined to boost privacy protection, FedGenID excels in mitigating the adverse effects of DP by enhancing the diversity and coverage of the data, and improving the model's robustness and generalization. These findings reinforce the robustness and efficiency of our proposed framework, emphasizing its practical applicability in privacy-sensitive settings.

Figure 10 illustrates a comparative study between FedGenID and FedID, when also considering the impact of DP training on the classification accuracy of both frameworks on both test sets. The results demonstrate the potential of our proposed FedGenID and its ability to maintain competitive accuracy levels across different TestSets, with FedGenID achieving 92.72% and 92.47% accuracy without and with DP-training in the original TestSet. Despite a slight drop with DP-training, FedGenID maintains commendable performance, surpassing FedID in the "Augmented-TestSet" by 14% and 10% without and with DP training, respectively. These findings position FedGenID as a robust and adaptable privacy-preserving IDS, addressing the evolving challenges of cyber threat detection in privacy-sensitive IoT environments. done Table VI demonstrates per-class performance results to evaluate the effectiveness of FedGenID in enhancing the precision and recall of detecting and identifying various types of cyber threats and robustness against zero-day attacks. Both FedID and FeGenID achieve high precision and recall without DP in detecting the 'Normal' traffic for threat detection. With DP, precision drops slightly, while recall remains competitive in all experiments. Regarding specific attack categories, the precision and recall scores of FedGenID and FedID exhibit notable distinctions across different privacy settings. In scenarios like 'DDoS_ICMP', 'DDoS_UDP', 'MITM', and 'Password', FedGenID achieves performance levels nearly equivalent to or better than FedID without DP. When both privacy-enhancing strategies are combined, FedGenID exhibits robust performance, especially in scenarios involving zero-day attacks. These results underscore the potential of FedGenID as a valuable tool for privacy-preserving FL in security-sensitive contexts. However, while FedGenID demonstrates promising results in accuracy, resilience, and generalization, there are specific classes where further refinement may enhance precision and recall.

TABLE VIII: Evaluating FedGenID and FedID against zero-day attacks

Classes	FedID			FedGenID		
	Recall	Fpr	Fnr	Recall	Fpr	Fnr
Normal	0.78	0.39	0.14	0.98	0.25	0.10
Back	0.01	0.00	0.00	0.29	0.00	0.01
Scan	0.53	0.20	0.03	0.84	0.05	0.09
ICMP	0.06	0.05	0.08	0.87	0.06	0.09
Pass	0.00	0.00	0.12	0.88	0.06	0.11
Port	0.00	0.00	0.13	0.95	0.17	0.05
UDP	0.06	0.04	0.00	0.84	0.01	0.00
Upload	0.24	0.05	0.03	0.83	0.02	0.06
HTTP	0.00	0.00	0.00	0.59	0.00	0.01
SQL	0.00	0.00	0.04	0.83	0.04	0.05
Rans	0.00	0.00	0.05	0.78	0.06	0.09
TCP	0.10	0.27	0.01	0.67	0.02	0.02
XSS	0.00	0.00	0.21	0.93	0.14	0.13
MITM	0.00	0.00	0.01	0.79	0.01	0.01
Fing	0.00	0.00	0.14	0.87	0.11	0.16
Accuracy	35.27%			92.17%		

Fpr: False Positive Rate, **Fnp :** False Negative Rate

Figure 8 and Table VIII compare the performance of FedGenID and FedID in detecting and identifying zero-day attacks. The results demonstrate that FedGenID exhibits significantly higher performance metrics, with high recall rates for most attack classes and lower false positive (FPR) and false negative rates (FNR). For instance, FedGenID achieves low FPR and FNR for the 'Normal' class, ensuring robust security against zero-day attacks by minimizing the misclassification of benign traffic as malicious. Additionally, FedGenID achieves low FNR across different attack classes, demonstrating its ability to detect and identify malicious traffic instances accurately, thereby reducing the risk of undetected attacks. These findings emphasize the importance of leveraging synthetic FL with data augmentation to enhance robustness against emerging and evolving cyber threats.

Overall, our proposed FedGenID framework presented a novel contribution to federated generative intrusion detection and demonstrated its efficiency in addressing challenges posed by privacy preservation, zero-day attacks, and emerging cyber threats in industrial IoT applications.

In table VII, we compare our FedGenID framework and recent state-of-the-art GAN-based security frameworks. The scope of the comparison covers the GAN-based intrusion detection application, specifically tailored to enhance cyber threat detection performance. We distinguish our FedGenID framework by opting for a recent and real-world industrial IoT dataset with various network traffic features, providing a more realistic representation of challenges than previous studies. In addition, we distinctively opt for a federated generative model that undergoes training without the exchange of local critics to enhance the adaptability of FL against data challenges, followed by a synthetic federated classifier learning approach to improve robustness against zero-day cyber threats. Furthermore, our research underscores the significance of data curation in assessing the consistency of the generated traffic data, a crucial factor often missed in earlier works. Also included is the adversarial defense against three sophisticated attacks to boost the resilience of IDS in the face of adversarial attempts, a consideration often overlooked in prior works. Moreover, our study sets itself apart by incorporating DP-training, which provides an additional layer of data privacy during collaborative learning.

Overall, our comparative study offers novel contributions, including a recent dataset choice, the adoption of FL and DP, and meticulous attention to data curation, addressing gaps observed in prior research.

VI. Conclusions and Future Work

In this paper, we have introduced an innovative FL framework named FedGenID. Specifically, we proposed an improved federated generative framework to generate synthetic data and blend it with actual client data. Thus overcoming imbalanced and distributed data challenges while improving the efficiency and robustness against cyber threats. The results conducted on a recent industrial cyber security dataset demonstrated the efficiency of our proposed security framework while maintaining data privacy. However, secure aggregation and authentication of model sharing are still required to ensure the integrity and trustworthiness of our framework. Future studies will focus on addressing this limitation. Furthermore, we aim to explore ensemble learning approaches for collective decision-making and self-supervised learning methodologies to enhance generative model capabilities.

References

[1] N. Tuptuk and S. Hailes, "Security of smart manufacturing systems," *Journal of manufacturing systems*, vol. 47, pp. 93–106, 2018.

[2] M. Lezzi, M. Lazoi, and A. Corallo, "Cybersecurity for industry 4.0 in the current literature: A reference framework," *Computers in Industry*, vol. 103, pp. 97–110, 2018.

[3] B. Chen, Y. Tan, Z. Sun, and L. Yu, "Attack-resilient control against fdi attacks in cyber-physical systems," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 6, pp. 1099–1102, 2022.

[4] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: introduction and outlook," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.

[5] M. Agarwal, S. Purwar, S. Biswas, and S. Nandi, "Intrusion detection system for ps-poll dos attack in 802.11 networks using real time discrete event system," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 792–808, 2016.

[6] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40 281–40 306, 2022.

[7] D. Hamouda, M. A. Ferrag, N. Benhamida, and H. Seridi, "Intrusion detection systems for industrial internet of things: A survey," in *2021 International Conference on Theoretical and Applicative Aspects of Computer Science (ICTAACS)*. IEEE, 2021, pp. 1–8.

[8] ——, "Ppss: A privacy-preserving secure framework using blockchain-enabled federated deep learning for industrial iots," *Pervasive and Mobile Computing*, p. 101738, 2022. [Online]. Available: https://doi.org/10.1016/j.pmcj.2022.101738

[9] N. Rodríguez-Barroso, D. Jiménez-López, M. V. Luzón, F. Herrera, and E. Martínez-Cámara, "Survey on federated learning threats: Concepts, taxonomy on attacks and defences, experimental study and challenges," *Information Fusion*, vol. 90, pp. 148–173, 2023.

[10] Y. Cheriguene, W. Jaafar, H. Yanikomeroglu, and C. A. Kerrache, "Towards reliable participation in uav-enabled federated edge learning on non-iid data," *IEEE Open Journal of Vehicular Technology*, 2023.

[11] N. Martins, J. M. Cruz, T. Cruz, and P. H. Abreu, "Adversarial machine learning applied to intrusion and malware scenarios: a systematic review," *IEEE Access*, vol. 8, pp. 35 403–35 419, 2020.

[12] J. Zhang, L. Zhao, K. Yu, G. Min, A. Y. Al-Dubai, and A. Y. Zomaya, "A novel federated learning scheme for generative adversarial networks," *IEEE Transactions on Mobile Computing*, 2023.

[13] T. Chuenbubpha, T. Boonchoo, J. Haga, and P. Rattanatamrong, "Solving non-iid in federated learning for image classification using gans," in *2023 20th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. IEEE, 2023, pp. 333–338.

[14] S. Bond-Taylor, A. Leach, Y. Long, and C. G. Willcocks, "Deep generative modelling: A comparative review of vaes, gans, normalizing flows, energy-based and autoregressive models," *IEEE transactions on pattern analysis and machine intelligence*, 2021.

[15] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: A survey toward private and secure applications," *ACM Computing Surveys (CSUR)*, vol. 54, no. 6, pp. 1–38, 2021.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[17] R. Durall, A. Chatzimichailidis, P. Labus, and J. Keuper, "Combating mode collapse in gan training: An empirical analysis using hessian eigenvalues," *arXiv preprint arXiv:2012.09673*, 2020.

[18] S. B. Hulayyil, S. Li, and L. Xu, "Machine-learning-based vulnerability detection and classification in internet of things device security," *Electronics*, vol. 12, no. 18, p. 3927, 2023.

[19] Z. Liu, S. Li, Y. Zhang, X. Yun, and Z. Cheng, "Efficient malware originated traffic classification by using generative adversarial networks," in *2020 IEEE symposium on computers and communications (ISCC)*. IEEE, 2020, pp. 1–7.

[20] S. I. Popoola, R. Ande, B. Adebisi, G. Gui, M. Hammoudeh, and O. Jogunola, "Federated deep learning for zero-day botnet attack detection in iot-edge devices," *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3930–3944, 2021.

[21] J.-Y. Kim, S.-J. Bu, and S.-B. Cho, "Zero-day malware detection using transferred generative adversarial networks based on deep autoencoders," *Information Sciences*, vol. 460, pp. 83–102, 2018.

[22] M. A. Ferrag, O. Friha, B. Kantarci, N. Tihanyi, L. Cordeiro, M. Debbah, D. Hamouda, M. Al-Hawawreh, and K.-K. R. Choo, "Edge learning for 6g-enabled internet of things: A comprehensive survey of vulnerabilities, datasets, and defenses," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 4, pp. 2654–2713, 2023.

[23] M. A. Merzouk, F. Cuppens, N. Boulahia-Cuppens, and R. Yaich, "Investigating the practicality of adversarial evasion attacks on network intrusion detection," *Annals of Telecommunications*, vol. 77, no. 11-12, pp. 763–775, 2022.

[24] M. O. Kaplan and S. E. Alptekin, "An improved bigan based approach for anomaly detection," *Procedia Computer Science*, vol. 176, pp. 185–194, 2020.

[25] N. Abdalgawad, A. Sajun, Y. Kaddoura, I. A. Zualkernan, and F. Aloul, "Generative deep learning to detect cyberattacks for the iot-23 dataset," *IEEE Access*, vol. 10, pp. 6430–6441, 2021.

[26] Y. Wu, L. Nie, S. Wang, Z. Ning, and S. Li, "Intelligent intrusion detection for internet of things security: A deep convolutional generative adversarial network-enabled approach," *IEEE Internet of Things Journal*, 2021.

[27] G. Xie, L. T. Yang, Y. Yang, H. Luo, R. Li, and M. Alazab, "Threat analysis for automotive can networks: A gan model-based intrusion detection technique," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4467–4477, 2021.

[28] I. Siniosoglou, P. Radoglou-Grammatikis, G. Efstathopoulos, P. Fouliras, and P. Sarigiannidis, "A unified deep learning anomaly detection and classification approach for smart grid environments," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1137–1151, 2021.

[29] A. Tabassum, A. Erbad, W. Lebda, A. Mohamed, and M. Guizani, "Fedgan-ids: Privacy-preserving ids using gan and federated learning," *Computer Communications*, 2022.

[30] J. Zhang, B. Chen, X. Cheng, H. T. T. Binh, and S. Yu, "Poisongan: Generative poisoning attacks against federated learning in edge computing systems," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3310–3322, 2020.

[31] X. He, Q. Chen, L. Tang, W. Wang, and T. Liu, "Cgan-based collaborative intrusion detection for uav networks: A blockchain-empowered distributed federated learning approach," *IEEE Internet of Things Journal*, vol. 10, no. 1, pp. 120–132, 2022.

[32] M. Rasouli, T. Sun, and R. Rajagopal, "Fedgan: Federated generative adversarial networks for distributed data," *arXiv preprint arXiv:2006.07228*, 2020.

[33] B. Xin, W. Yang, Y. Geng, S. Chen, S. Wang, and L. Huang, "Private fl-gan: Differential privacy synthetic data generation based on federated learning," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 2927–2931.

[34] Y. Li, J. Li, and Y. Wang, "Privacy-preserving spatiotemporal scenario generation of renewable energies: A federated deep generative learning approach," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 4, pp. 2310–2320, 2021.

[35] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," *Advances in neural information processing systems*, vol. 30, 2017.

[36] Y. Gu, Y. Yang, Y. Yan, F. Shen, and M. Gao, "Learning-based intrusion detection for high-dimensional imbalanced traffic," *Computer Communications*, vol. 212, pp. 366–376, 2023.

[37] B. Meenakshi and D. Karunkuzhali, "Enhancing cyber security in wsn using optimized self-attention-based provisional variational auto-encoder generative adversarial network," *Computer Standards & Interfaces*, vol. 88, p. 103802, 2024.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: