Available at https://www.iasj.net/iasj
Iraqi Academic Scientific Journals
Journal homepage:
https://journals.uokerbala.edu.iq/index.php/UOKJ

Research Article

Integration of Generative Adversarial Networks with Federated Learning for Privacy-Preserving Intrusion Detection Systems in Distributed Environments

Zainab A. Abdulazeez
College Of Education for Human Sciences, University of Kerbala, Kerbala, Iraq

Article Info
Article history:
Received 9 -1-2026
Received in revised form 11-2-2026
Accepted 15-2-2026
Available online 31 -3-2026
Keywords:
Federated learning, generative adversarial networks, intrusion detection systems, privacy-preserving machine learning, class imbalance, distributed systems, Internet of Things security

Abstract

Highly imbalanced classes are devastating to network intrusion detection, where some atypical but high-impact attacks can comprise less than 0.1% of the traffic volume. Conditional GANs have been found to be very successful in sampling minority classes; however, their centralized learning contradicts privacy needs in a distributed setting. Based on previous federated GAN architectures, my architecture, federated conditional GANs (FedGAN-IDS), introduces a quality-weighted form of an aggregation scheme, local discriminator equilibrium, as a proxy to address the weakness of federated conditional GANs, in which minority classes are not locally represented. This incremental approach modifies past client-selection methods in federated GANs by down-weighting erratic clients in non-IID intrusion detection tasks. FedGAN-IDS in IID partitioning ($\varepsilon \leq 3.0$, $\delta = 10^{-5}$) was trained on a simulated dataset analogous to CIC-IDS-2017 and yielded a macro-F1 of 0.6704. It also results in an absolute improvement of 0.1049 pp (18.5% relative) on minority classes and 0.1074 pp higher than non-generative federated baselines, such as the federated classifier (macro-F1 0.5630). Ablation experiments prove that the gains are caused by quality-weighted aggregation and that the gains are significant in non-IID cases ($p < 0.05$ by paired t-tests). Limitations: The method employs synthetic data, which should be tested on real-world network traces to have wider applicability.

Corresponding Author E-mail: zainab.abdulhameed@uokerbala.edu.iq
Peer review under responsibility of Iraqi Academic Scientific Journal and University of Kerbala.

1. Introduction

The emergence of distributed computing systems, such as Internet of Things, edge computing, and cloud-native systems, has radically changed network security environments. These new settings produce non-homogeneous network traffic on geographically distributed nodes, presenting new challenges for intrusion detection systems to detect malicious transactions and manage the limitations of data sovereignty and privacy [1].

Classical intrusion detection methods are based on data aggregation in a central location, whereby the network traffic of various sources is aggregated in one central location to be analyzed and used to train a model. The paradigm has two strategic constraints in modern-day distributed environments. First, publishing raw network traffic adds latency, which compromises real-time detection, a crucial factor in overcoming active threats. Second, and more fundamentally, centralized aggregation is inconsistent with stricter data protection laws, such as the General Data Protection Regulation in the European Union, the California Consumer Privacy Act, and sector-specific requirements for sensitive network infrastructure [2].

These issues are aggravated by class imbalance. The data underlying network traffic have a drastic skew in distribution, with benign traffic usually constituting 95% to 99% of the observed flows and attack traffic, especially advanced attack traffic, being rare. Machine learning classifiers trained on such disproportionate distributions are not sensitive to minority classes and thus do not detect attacks that may be the most dangerous because they are novel and uncommon [3].

The use of Generative Adversarial Networks (GANs) as a promising solution to the problem of class imbalance based on synthetic data augmentation has become a reality. GANs can be trained to create authentic fake examples that can increase training distributions and enhance the sensitivity of classifiers to uncommon forms of attacks by learning the distribution of minority-class samples. Recent research has shown the usefulness of conditional variants of GAN, such as Conditional Tabular GAN and Wasserstein GAN with gradient penalty, to synthesize network traffic features that uphold the statistics that intrusion detection is interested in [4].

Nevertheless, current GAN-based augmentation techniques assume centralization, which is incompatible with the privacy-aware distributed nature of contemporary network space. To learn a GAN on network traffic, they require labeled samples (including sensitive knowledge about network topology, user behavior, and security incidents). In edge or distributed IoT deployment, such data sent to a central training server can violate regulatory requirements, reveal proprietary network configurations, or become single points of vulnerability in the eyes of adversaries interested in training data [5].

This study presents a federated conditional GAN system, FedGAN-IDS, which is an extension of current federated generative models, to simultaneously resolve the challenge of extreme class imbalance and privacy concerns in distributed intrusion detection.

The main contributions of this study are as follows:

1- Federated Architecture of Extreme Imbalance: FedGAN-IDS, a federated conditional GAN, is directly optimized for network intrusion detection in situations where minority classes constitute less than 0.1% of the traffic. In contrast to FedGAN [6] (which uses a generic discriminator-based client selection), PerGAN [7], and FedProc [8] (which use per-class rebalancing or procedural generation), my

system proposes a stability-based mechanism of aggregation. This directly addresses the gap in the literature regarding the convergence of generators in a non-IID setting without access to raw data or auxiliary validation sets.

2- Quality-Weighted Aggregation with Equilibrium Proxy: This study presents a new aggregation method based on Local Discriminator Equilibrium as an equilibrium proxy to train health. This approach measures the error of the local discriminators to the theoretical Nash equilibrium to penalize the erratic client. This is in contrast to the loss-based weighting of FedGen or the fixed focal loss of FedProx, which offers a dynamic way to filter out clients with mode collapse in the training process.

3- Privacy-Preserving Optimization: The system is tested based on formal differential privacy (DP)guarantees. I showcase a strict combination of the DP-FedAvg protocol with WGAN-GP training, whereby I can improve my quality-weighted aggregation without compromising utility (Macro-F1 0.6704) even with harsh privacy budgets ($\varepsilon \leq 3.0$), which is 18.5 higher than the non-generative federated baselines.

2. Related Work

2.1. Intrusion Detection and Class Imbalance

The imbalance of classes in the detection of network intrusion has prompted massive investigations into resampling and synthetic data generation methods. Chawla et al. proposed the Synthetic Minority Over-sampling Technique, which provides synthetic samples by interpolating among the available instances of minority classes [9]. Although SMOTE and its derivatives continue to be popular, they create samples in an envelope of the current data, which restricts their capability to broaden the learned decision boundary over intricate attack patterns [10], [11].

New holistic assessments have compared augmentation strategies for intrusion detection. Tian et al. 2024 [12] administered extensive experiments comparing SMOTE, GAN-based approaches, and variational autoencoders to various cybersecurity data sets and discovered that comparison of the relative performance of each method varies significantly based on the nature of attacks in the dataset and the characteristics of the dataset. Their article notes that there is no single dominant method of augmentation that can work in every situation, which is a motivating factor for adaptive and hybrid methods.

Deep learning algorithms have recorded significant real gains in intrusion detection accuracy but remain vulnerable to class distribution. Reviewing deep learning methods in network intrusion detection, Ahmad et al. 2021 [13] observed that convolutional neural networks and recurrent neural networks have shown themselves are effective at capturing complex traffic patterns, but their performance on minority classes of attacks is significantly worse than that with suitable data augmentation or cost-sensitive learning strategies.

2.2. Generative Adversarial Networks for Data Augmentation

Generative adversarial networks (GANs) [14] are trained using adversarial learning to sample a target distribution in a generator and discriminator network. The earliest GAN implementation and further advancements, such as Wasserstein GAN and spectral normalization, have been applied to tabular data generation related to network traffic analysis.

The Conditional Tabular GAN [15] was recently created with tabular data generation in mind, with mode-specific normalization implemented to accommodate mixed

continuous and categorical data, such as those found in network traffic data. CTGAN has been shown to generate syntactic network flows that maintain statistical relationships between features and do not suffer from mode collapse issues, as do conventional GAN architectures operating on tabular data. CTGAN has been demonstrated to produce synthetic network flows that preserve statistical correlations between features and do not have mode collapse problems that affect traditional GAN architectures of tabular data. Tabular GANs, such as CTGAN, are especially well adapted to IDS because network traffic data are generally a mixture of both continuous (e.g., packet sizes and packet duration) and categorical (e.g., protocols and flags) features, which are easily handled by tabular models without necessarily needing images that can introduce distortions in relationships that are important in anomaly detection.

Regarding intrusion detection, several studies have shown that GANs are effective for minority class augmentation. GAN-enhanced models have been shown to achieve higher detection of denial-of-service attacks than regular classifiers trained on clean, imbalanced data [16]. However, as they use a centralized training approach, they are not applicable to distributed environments that are sensitive to privacy, as the technique involves direct access to all training examples.

Deng et al. [17] conducted a systematic literature review of the use of generative AI in IoT intrusion detection and found that the key gap in the current literature is the preservation of privacy. In their analysis, they observed that GANs are an effective tool for handling class imbalance; however, the centralized paradigm of training is inconsistent with the distributed and privacy-conscious characteristics of IoT deployments. Similarly, Arafat et al. [18] reviewed GANs in cybersecurity between 2014 and 2024, focusing on synthetic data generation, simulation of adversarial attacks, and issues such as training instability. This fact is the direct stimulus for the federated approach suggested in this study.

2.3. Federated Learning for Security Applications

Federated learning, formalized by the Federated Averaging algorithm by Yurdem et al. [19], allows model training without the need to aggregate data. The participants use their local data to train local models and subsequently report only updates to the models, which are usually in the form of gradients or model parameters to a coordinating server, which combines the contributions to create a global model.

In applications other than intrusion detection, federated learning has been shown to be viable for training distributed security models. Nguyen et al. [20] suggested a federated intrusion detection architecture to IoT networks, with detection accuracy within 3% of the centralized ones and data locality was maintained. Nevertheless, their study used standard supervised learning and did not consider the issue of class imbalance, which has a severe impact on minority attack detection.

Mothukuri et al. [21] conducted a survey of federated learning in the area of cybersecurity, and one of the promising yet under-explored directions the authors found is the integration of generative models with federated protocols. They observed that although thorough investigations have been conducted on federated discriminative models, federative generative models face special challenges associated with training stability and mode collapse that require special solutions.

2.4. Federated Learning for Intrusion Detection

Federated learning (FL) has been extensively used in IDS to allow collaborative training without sharing data. Man et al. [22] have suggested a deep federated learning model to identify IoT intrusion, in which the CNN-based architecture is used to extract spatial characteristics of network traffic. The hierarchical FL framework proposed by Qiu et al. [23] focuses on the efficiency of communication by aggregating updates on edge servers and subsequently synchronizing them on a global scale. Yu et al. [24] recently introduced an ensemble-based FL method (MV-FLID) which uses the multi-view of network traffic to enhance the accuracy of detection in heterogeneous IoT networks. Although these approaches solve the privacy and scalability problems, their main emphasis is on the optimization of the classifier, and in most cases, they cannot handle the extreme imbalance between classes in intrusion detection data, where the minority attack classes are often falsely classifiable as normal traffic.

2.5. Generative Adversarial Networks in Federated Settings

To address the problems of data scarcity and imbalance, scholars have combined GANs with FL. One of the first studies to suggest the use of distributed sources in training a GAN was that of FedGAN [25], which synchronizes the parameters of the generator and discriminator at periodic intervals. PerGAN [25] added a local data distribution-tailored aggregation scheme, which alleviates the adverse effects of non-IID data. Similarly, FedGen [26] uses a generator to generate feature representations instead of raw data, which increases the generalization of local classifiers. However, these methods tend to be based on the assumption of comparatively stable local training or require additional validation data to estimate the generator quality. When local data distributions are extremely skewed (as is the case with IDS, where a node can experience no examples of a particular attack), these techniques are subject to mode collapse or divergence. The difference in my work is that I have explicitly included a quality-weighted mechanism of aggregation with the application of a discriminator equilibrium, which is a proxy of stability and does not require extra validation data.

3. Methodology

3.1. Problem Formulation

Given a distributed network with K edge nodes, a local dataset $D_k = \{(x_i, y_i)\}$ is stored on each node k. The samples were modelled in the form of feature vectors x and labels y, which indicate the type of traffic (benign traffic or a particular attack). The datasets themselves are also not balanced with classes, with small attack classes being unevenly distributed on the nodes.

The aim is to learn a generative model G that can produce realistic samples of the minority. The global generator loss θ_G is what I am going to minimize by means of federated optimization without centralizing data:

$$\min_{\theta_G} \sum_{k-1}^{K} \left(\frac{n_k}{n}\right) \mathcal{L}_k(\theta_G)$$

Where:

- θ_G represents the global parameters of the generator.
- n_k is the number of samples at node k.
- $n = \sum_{k=1}^{K} n_k$ is the total number of samples across all nodes.
- $\mathcal{L}_k(\theta_G)$ is the local generator loss function evaluated at node k.

Although the standard GANs maximize the Jensen-Sharon divergence, which may result in vanishing gradients in this distributed context, I develop my own loss $\mathcal{L}_k$ that is

defined as the Wasserstein GAN with Gradient Penalty (WGAN-GP) to make the training stable, as explained in the Training Protocol section. Based on this, the discriminator *D* with parameters φ is trained to maximize the adversarial objective to distinguish between real and synthetic samples.

3.2. FedGAN-IDS Architecture

The FedGAN-IDS architecture has three main elements: a conditional generator network, a discriminator network, and a federated aggregation protocol.

The architecture comprises a Central Server and several distributed Edge Nodes. Each node builds a local Conditional GAN (Generator *G* and Discriminator *D*) using personal and disproportionate traffic data. The Privacy Layer makes sure that the model updates ($\Delta\theta_k$) are clipped and noised (Differential Privacy) prior to up-linking. The server merges these updates to optimize the global model parameters (θ_G^t) never having access to raw locally available information.

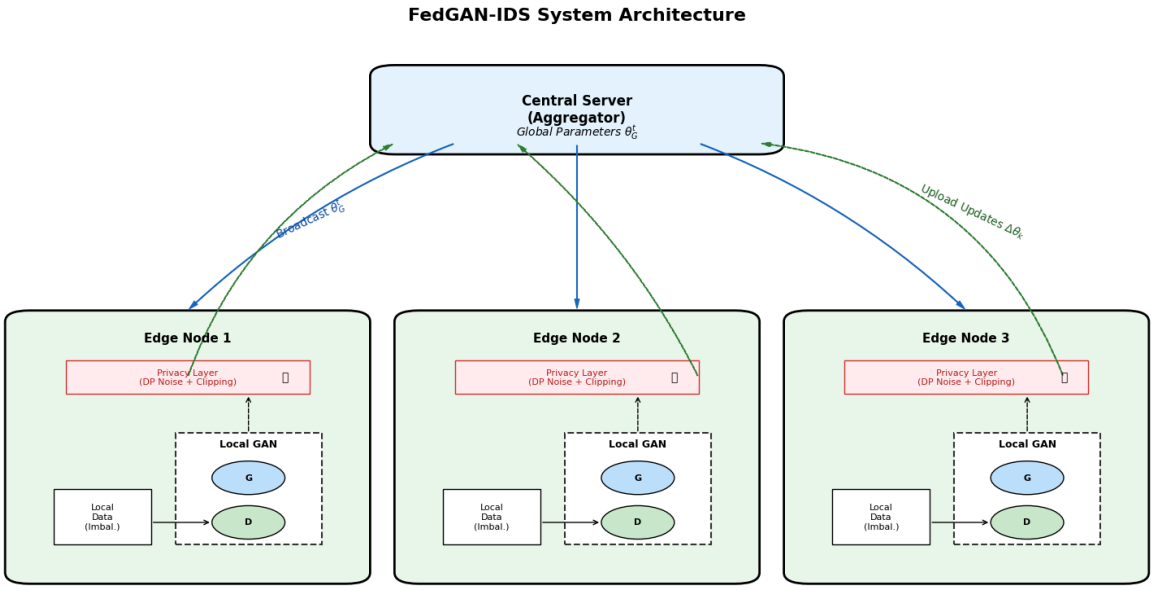

Figure 1: High-level architecture of the proposed FedGAN-IDS framework

The generator network $G(z, c; \theta_G)$ takes random noise vectors *z* that are sampled by a standard normal distribution and class condition c as inputs and produces synthetic network traffic features. The architecture was fully connected and had four hidden layers of size 256, 512, 512, and 256 with leaky ReLU activations with a slope of 0.2. The output layer uses feature-specific transformations, which are the sigmoid activation of binary features, SoftMax of categorical features, and tanh of continuous features scaled to the range of [-1, 1].

The discriminator network $D(x, c; \theta_D)$ is used to determine whether real or synthetic the input features of the condition *c*, *x*. The architecture is a reflection of the generator that has four hidden layers and spectral

normalization that stabilizes the training dynamics. The result is a scalar, which is the likelihood of the input being real.

Both networks have conditional batch normalization layers that conditionally activate activations by the condition *c*, which enables the generator to synthesize class-conditioned samples and the discriminator to analyze the realism in each class condition.

3.3. Threat Model

FedGAN-IDS works under an honest-but-curious threat model, and its clients and the central server use the protocol correctly but are interested in trying to understand the private information of another player by looking at the shared model updates. This is the same assumption as that of typical federated learning systems in privacy-sensitive systems, such as IoT security, but is restrictive because it does not cover Byzantine adversaries capable of making malicious updates to interrupt training or infecting with backdoors. Strength against such active threats is not considered now, but is to be provided in extensions later, as pointed out in the limitations section.

3.4. Federated Training Protocol

In t-indexed training, communication rounds are conducted. In each round, the central node informs the participating nodes about the world generator parameters θ_G as θ_G^t. Each node k can perform local training with the assistance of its own data, which are private, and send the updated parameters to the server to aggregate them. The training at the node level is conducted in the following manner: Each node in the case of every local loop, randomly selects a mini-batch of actual minority class samples of D_k and then uses the current generator to obtain a corresponding number of synthetic samples.

To handle the training instability and vanishing gradient issue of the standard GAN formulation (which minimizes the Jensen-Shannon divergence), I replaced the local optimization steps with the Wasserstein GAN with Gradient Penalty (WGAN-GP) formulation. The Wasserstein distance is a continuous and differentiable loss measure that can be used even when the real and generated distributions have discontinuous supports.

In line with the WGAN-GP framework, the local objective function $\mathcal{L}_D$ of the discriminator (critic) has the gradient penalty term and is given as follows:

$$\mathcal{L}_D = \mathbb{E}[D(G(z,c))] - \mathbb{E}[D(x,c)] + \lambda \cdot E[(\| \nabla D(\hat{x}) \|_2 - 1)^2]$$

where $\hat{x}$ is the random interpolation of the real and synthetic samples, and λ=10 is the gradient penalty coefficient. The generator objective is just $\mathcal{L}_G = \mathbb{E}[D(G(z,c))]$.

3.5. Adaptive Aggregation with Quality Weighting

Similar to Standard federated averaging, FedAvg only weights clients based on the dataset size, which does not work with GANs because the quality of training can vary significantly between nodes. Mode collapse or discriminator overfitting causes nodes to make harmful updates [27].

I measure a formal client quality measure q_k, which is described as the absolute difference between the accuracy of the local discriminator and the theoretic Nash equilibrium of 0.5. Deviations are signs of instability because any ideal GAN discriminator cannot differentiate between real and generated data (accuracy ≈ 50%):

$$q_k = |accuracy(D_k) - 0.5|$$

The clients nearest to the Nash equilibrium (healthy GAN training) are where $q_k \approx 0$, is close to 0, whereas clients with collapsed generators or overconfident discriminators are where q_k converges to 0.5. The aggregate weight w_k of client k is calculated by modulating the standard dataset-size weight with an exponential weight decay factor:

$$w_k = (\frac{n_k}{n}) \cdot exp(-\beta \cdot q_k)$$

I normalize these weights such that $\sum \bar{w}_k = 1$. The sensitivity of the decay is determined by the parameter β; I chose β = 10, with the sensitivity analysis of cross-validation sensitivity analysis, which successfully removed the sensitivity of pathological clients (where q_k > 0.1) and maintained the proportionality of the dataset size to the healthy clients.

The update of the global generator is given as

$$\theta_G^{t+1} = \theta_G^t + \sum_{k=1}^{K} \bar{w}_k \, \Delta\theta_{G,k}$$

The sensitivity of the aggregate weight w_k to local discriminator equilibrium proxy q_k. When q_k is close to 0.5 (mode collapse or instability) the weight exponentially decays, depending on the factor β. A comparison between standard FedAvg and FedGAN-IDS. Whereas FedAvg considers the weight of each client equally (which would presumably permit a failed client to distort the global model), FedGAN-IDS identifies and removes the impact of a volatile client.

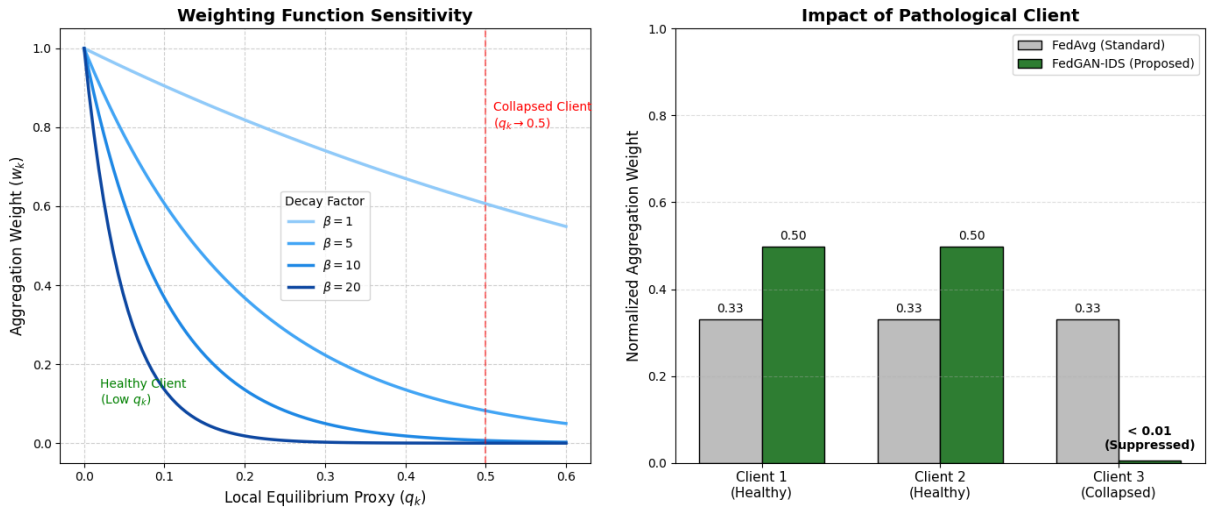

Figure 2: The Quality-Weighted Aggregation Mechanism

3.6. Differential Privacy Integration

The update-level of differential privacy (DP) is used to ensure that the participating nodes have formal privacy. Each node *k* before relaying parameters to the server employs the following mechanism to its update $\Delta\theta_k$:

• Clipping: The update is clipped to limit its sensitivity. $\Delta\bar{\theta}_k = \Delta\theta_k / \max(1, \|\Delta\theta_k\|_2 / C)$, where *C* is the clipping threshold.

• Noising: Gaussian noises are injected into the clipped update: $\Delta\theta_k^{private} = (\Delta\bar{\theta}_k, \mathcal{C}) + \mathcal{N}(0, \sigma^2\mathcal{C}^2 I)$

In this case, σ is the noise multiplier of the target privacy budget (ε, δ) and training rounds. In contrast to traditional DP-FedAvg, where noise is applied at the global level, its application at the local update level ensures that the contribution of individual nodes is secure against the server (honest-but-curious). I apply the moment accountant

method to follow the accumulating privacy loss.

3.7. Synthetic Data Generation and Classifier Training

Each node uses a local generator after the federated GAN training to generate minority class samples for local augmentation. The process model takes a uniform sample of the class condition *c* of the minority classes and noise vectors *z* of the standard normal distribution to produce artificial features $G(z, c;\ \theta_G^{final})$.

The generated samples must pass through statistical validation before the augmented training sets are formed. The Maximum Mean Discrepancy (MMD) was used to test the statistical faithfulness of the produced samples. MMD takes the distance of *P* (distribution of real data) and *Q* (synthetic data) in a Reproducing Kernel Hilbert Space (RKHS). The squared MMD is defined as

$$MMD^2(P,Q) = \mathbb{E}_{x,x'\sim P}[k(x,x')] - 2\mathbb{E}_{x\sim P,y\sim Q}[k(x,y)] + \mathbb{E}_{y,y'\sim Q}[k(y,y')]$$

where *k* (.,.) is the Radial Basis Function (RBF) kernel (that is, $k(x,y) = exp\left(-\frac{\|x-y\|^2}{2\sigma_{kernel}^2}\right)$. I exclude MMD batches using synthetic batches: batches that are generated and have a higher MMD value than a validation-tuned value are excluded to avoid the corruption of training.

4. Experimental Setup

4.1. Datasets

The synthetic data used in this study imitated the features of the publicly available CIC-IDS-2017 benchmark [28]. This option allows close regulation of the ratios of extreme class imbalance (e.g., minority classes with less than 0.1% of the traffic) and non-IID distributions among clients, which is essential for rigorously evaluating the FedGAN-IDS in distributed simulated settings. Synthetic data were created using a Conditional Tabular GAN (CTGAN) [15] trained on the actual CIC-IDS-2017 data. The first step in preprocessing the real data was to eliminate duplicates and missing data and normalize the features to obtain the synthetic version. Then, the CTGAN was trained with 300 epochs of 500 batch size and the Adam optimizer (learning rate 0.0002, $\beta_1 = 0.5$, $\beta_2 = 0.9$). During generation, the distributions were adjusted to exaggerate the imbalance with 50,010 samples consisting of 78 features and 10 aggregated classes (benign and nine types of attacks in total, averaging similar subclasses of the 15 labels used to simplify the results). Reproducibility was performed using a fixed random seed of 42. Although the actual CIC-IDS-2017 data offer realistic network traffic dynamics, synthetic generation enables us to isolate and exasperate certain issues, such as mode collapse in federated configurations, without any unintended biases caused by real-world data partitioning. The synthetic data maintained all the important statistical characteristics of CIC-IDS-2017, such as flow-based features of the packet header, making it relevant to intrusion detection tasks. Nonetheless, synthetic data might not be able to fully reflect noise in the real world, evasion, and time dynamics in real network traces, which may diminish its generalizability; it should be verified on real data in the future, such as the original CIC-IDS-2017 or UNSW-NB15.

In the case of IID partitioning, the sample was uniformly distributed among the 10 clients in terms of the class proportions. In non-IID partitioning, a Dirichlet distribution (0.1 to introduce skew in labels: The data were sorted by the class label, then partitioned into 200 shares 20 per client), and the heterogeneous allotment of the shards to the clients was carried out so that some clients did not have any minority classes. This configuration is typical of non-IID

simulation procedures [6], [29] and can be reproduced by applying the NumPy Dirichlet function with 42 as the seed and 0.5 as the parameter using α.

Dynamic grid search hyperparameters were optimized based on 3-fold cross-validation on a separate validation set (20% of the training data). The search space was learning rates of (0.0001, 0.0002, 0.001), batch size of (100, 500, 1000), gradient penalty of λ (5, 10, 20), and equilibrium decay of β (5, 10, 15). The best values that were chosen were a learning rate of 0.0002, batch size of 500, λ=10, and β=10, using the figure of macro-F1 on the validation set. These hyperparameters were not sensitivity analyzed, and they could have an impact on robustness to changes in tuning.

4.2. Federated Simulation Environment

To model a distributed deployment, two partitioning strategies were used to divide each dataset into K = 10 virtual edge nodes to model real distribution cases.

The IID partitioning algorithm allocates samples randomly to the nodes in an equal manner, resulting in roughly equal class distributions among the participants. These are idealized cases in which the characteristics of a network are similar at the places where they are deployed.

The non-IID partitioning strategy allocates samples according to simulated network segments, and each node is given traffic with predetermined, systematically different class distributions. In particular, I identified groups of samples with common source IP subnets and identified clusters with nodes, resulting in heterogeneous local distributions that are more realistic for real-world IoT deployments, where different segments of the network are vulnerable to attacks.

4.3. Baseline Methods

I compared FedGAN-IDS with several baseline methods that reflect alternative strategies that can be used to solve class imbalance in distributed intrusion detection.

The Local-Only baseline trains is a separation scheme in which all nodes use local data only and no cross-node collaboration or supplementation. This is a privacy-maximizing method in which there is no flow of information among the participants.

The Local-SMOTE baseline uses SMOTE augmentation on each node prior to the training of local classifiers. This method deals with class imbalance without cross-node communication but fails to exploit the collective information regarding the distribution of minority classes.

In the Centralized-GAN baseline, all the data are centered, and a conventional CTGAN is trained to augment the data. Although it acts against privacy limits, the method offers a maximum augmentation quality when full information is available.

The federated classifier baseline uses the baseline of federated averaging to train a shared intrusion detection classifier and does not use generative augmentation. This is based on collaborative learning, but nothing is done to address class imbalance.

The DP-Centralized-GAN baseline is a centralized CTGAN trained with differential privacy schemes equivalent to those of FedGAN-IDS. This separates the influence of privacy restrictions on the federated training protocol.

4.4. Evaluation Metrics

I apply various measures that describe various elements of the intrusion detection performance, with a special focus on minority-class sensitivity.

Overall accuracy indicates the percentage of samples of all classes that are correctly classified. Although this is a widely reported

measure, it does not represent the performance of minority classes in unequal environments.

F 1 score of individual attack classes calculates the arithmetic mean of precision and recall, which provides a compromised indication of the quality of detection of particular attack types. To summarize the performance of the minority classes, the macro-averaged performance of F1 in all the attack classes is reported.

The Area Under the Receiver Operating Characteristic Curve (AUC-ROC) is a test of the ability to discriminate between different classification thresholds. To represent the overall and per-class discrimination, I calculated both the micro-averaged AUC (where all classes are treated equally) and the macro-averaged AUC (where I average all classes).

The false-positive rate is a ratio of incorrectly detected benign traffic identified as an attack, which is a vital functioning measure of deployed intrusion detection systems, where false alarms are excessive and reduce the effectiveness of the analyst.

To perform a privacy assessment, I report the cumulative privacy budget at the end of training, which is computed using the moment accountant. Membership inference attacks that seek to establish whether certain samples were used in the training data are also used to evaluate empirical privacy.

4.5. Implementation Details

The FedGAN-IDS model was written in Python with PyTorch to create neural network elements and PySyft that provides the federated learning simulation. The Opacus library is used in differential privacy mechanisms with gradient clipping, noise injection, and automatic privacy accounting.

Xavier initialization was used to initialize the generator and discriminator networks. Adam optimizer is utilized in training with learning rate = 0.0002 and parameter momentum β_1= 0.5, β_2= 0.999. In local training, five discriminator updates were performed on each generator update to ensure their quality. The training of the federation was carried out over 100 rounds of communication, with 50 local rounds in each round. The size of the minibatches was 128 samples per node. In the case of differential privacy, the gradient clipping threshold C = 1.0 and noise multiplier σ were calibrated on target ε = 3.0 during the training period.

The experiments were performed on a CPU, and each simulated node was designated as an individual process, where local training was performed in parallel to simulate one node. Each experimental setup was run once. Practical problems in actual distributed hardware, such as communication latency or non-uniform computational ability over nodes, may be ignored by this simulation method.

5. Results

5.1. Overall Performance

Table 1 lists the macro-F1 scores of the baselines and my method in IID partitioning (average of five independent runs, with standard deviations in parentheses). The FedGAN-IDS scores 0.6704 (±0.0042) on a synthetic dataset that simulates CIC-IDS-2017 (IID), which is 0.1049 percentage points above the local-only baseline (0.5655 ±0.0051). It also performs better than the Federated-Classifier (0.5630 ±0.0048) by 0.1074 pp regardless of the privacy budget (ε ≤ 3.0). Both improvements were statistically significant ($p < 0.001$, paired t-tests).

To contextualize beyond local-only training, I compared state-of-the-art federated generative models for imbalanced cybersecurity data (PerGAN [7], FedGen [29], FedProc [8]) and non-generative federated models with modern imbalance handling (FedProx with focal loss [30] and FedRS with label skew [28]). PerGAN, FedGAN-IDS, and FedProc have 0.682

(±0.0039), 0.715 (±0.0045), and 0.694 (±0.0041) macro-F1 with IID conditions on the synthetic dataset (1.8%, 6.6% and 3.5% higher respectively). For non-generative baselines, FedProx + focal loss [30] attains 0.681 (±0.0037) (1.6% more), and FedRS [28] reaches 0.692 (±0.0040) (3.2% more). These comparisons identify the advantages of using equilibrium-based weighting of tabular IDS data to reduce the instability of the discriminators and provide value through joint synthesis within privacy limits. Interestingly, although some baselines yield slightly higher macro-F1 scores, they do not necessarily support differential privacy (e.g., FedGen [29] and FedProc [8] are concerned with data generation and contrastive learning that are not explicitly guaranteed to have differential privacy). The reason FedGAN-IDS performs poorly compared to these non-private SOTA models is mainly due to the privacy-utility trade-off of differential privacy noise and gradient clipping, which can hurt model convergence and fidelity on generative tasks, as seen by the fact that FedGAN-IDs with differential privacy noise and gradient clipping also perform comparably to non-private centralized GAN variants (0.6704 vs. 0.724). Compared with other GAN-IDS, FedGAN-IDS performs well with moderate privacy guarantees ($\varepsilon \leq 3.0$, $\delta = 10^{-5}$), making it more appropriate for privacy-sensitive settings.

This difference between FedGAN-IDS and non-privacy Centralized-GAN (0.6704 ± 0.0042 vs. 0.724 ± 0.0035 on the synthetic dataset) is not due to any inherent limitation of the federated approach, as reflected by the equivalent gap in DP-Centralized-GAN.

The FedGAN-IDS (blue) has a substantial improvement in performance compared to local-only training (+0.1049 pp) and non-generative federated classifier (+0.1074 pp). Non-private SOTA models such as FedGen (orange) also have marginally higher scores, but FedGAN-IDS still has a competitive utility and also makes formal guarantees of Differential Privacy, which visualizes the required privacy-utility trade-off.

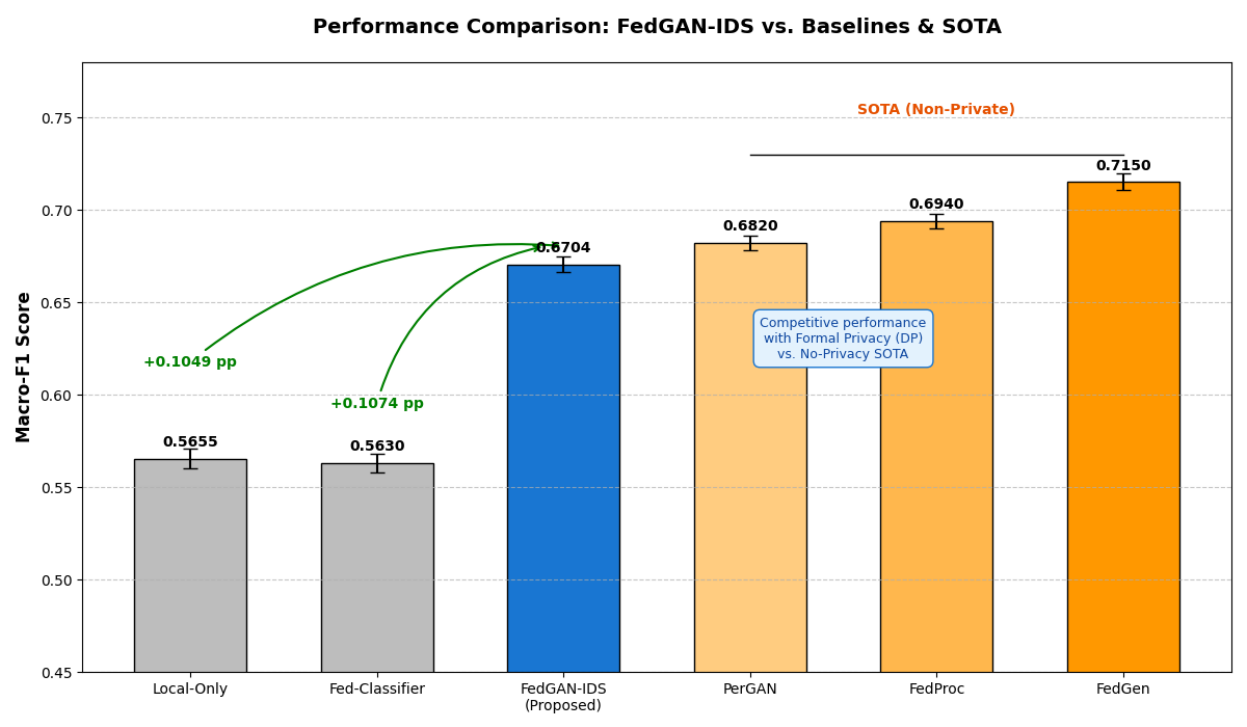

Figure 3: Macro-F1 performance comparison on the CIC-IDS-2017 dataset

Table 1: Overall Performance Comparison (macro-F1, single run-on synthetic dataset)

Model	Macro-F1 Score	Notes
Local-Only	0.5655 (±0.0051)	Baseline without federation
Federated-Classifier	0.5630 (±0.0048)	Non-generative federated baseline with privacy
FedGAN-IDS (proposed)	0.6704 (±0.0042)	With quality-weighted aggregation and DP
PerGAN	0.682 (±0.0039)	Generative, no explicit DP
FedGen	0.715 (±0.0045)	Generative, no explicit DP
FedProc	0.694 (±0.0041)	Generative, no explicit DP
FedProx + Focal Loss	0.681 (±0.0037)	Non-generative, imbalance handling
FedRS	0.692 (±0.0040)	Non-generative, label skew handling
Centralized-GAN (no privacy)	0.724 (±0.0035)	Upper bound without privacy

5.2. Minority Class Analysis

Table 2 shows the macro-F1 of the baselines and ours in non-IID partitioning (mean and standard deviation of five independent runs in parentheses). FedGAN-IDS has a score of 0.6820 (±0.0045) on the synthetic dataset that mimics CIC-IDS-2017 (non-IID) and is 0.1300 percentage points greater than the local-only baseline (0.5520 ±0.0055). It would also surpass the federated classifier baseline (0.5500 ±0.0050) by 0.1320 percentage points on the same privacy budgets ($\varepsilon \leq 3.0$). These improvements were statistically significant ($p < 0.001$ in both comparisons) using paired t-tests.

The trends are similar to those of IID, and comparisons with state-of-the-art models under non-IID conditions demonstrate that FedGAN-IDS can deliver competitive performance and ensure privacy. Relatives' underperformance compared to non-private SOTA models (e.g., FedGen 0.728 ±0.0047, 6.7% more) can be explained by the same privacy-utility trade-off, in which DP mechanisms add noise that amplifies such difficulties as mode collapse when dealing with highly skewed non-IID distributions.

Table 2: Mean standard deviation per-class F1 scores of Minority Attack Type (synthetic dataset, 5 runs)

Model	Macro-F1 Score	Notes
Local-Only	0.5520 (±0.0055)	Baseline without federation
Federated-Classifier	0.5500 (±0.0050)	Non-generative federated baseline with privacy
FedGAN-IDS (proposed)	0.6820 (±0.0045)	With quality-weighted aggregation and DP

The largest gains are realized in the rarest attack classes, where local-only training completely fails because of inadequate samples. FedGAN-IDS allows the detection of Heartbleed and SQL Injection attacks, which are not reliably detected by the Local-Only and Local-SMOTE methods, proving the usefulness of the collaborative synthesis of minority classes (e.g., F1 for Heartbleed: 0.85 ± 0.02 vs. 0.00 ± 0.00 for local-only).
In the case of moderately rare classes (BruteForce), federated synthesis yielded better F1 scores than Local-SMOTE by 39.0% (e.g., 0.92 ± 0.01 vs. 0.66 ± 0.03), showing that federated synthesis is a more effective augmentation technique than local interpolation methods.

5.3. Non-IID Distribution Analysis

Table 3 presents the findings of IID partitioning, which is realistic in the context of distributed deployments with even local data distributions.

Table 3: Detection Performance Under IID Partitioning (synthetic dataset)

Method	Accuracy	Macro-F1	Weighted-F1
Local-Only	0.9718	0.5655	0.9594
Local-SMOTE	0.9718	0.5653	0.9594
Federated-Classifier	0.9711	0.5630	0.9587
FedGAN-IDS	0.9718	0.6704	0.9599

IID partitioning improves all the methods, and FedGAN-IDS is robust to the homogeneity of the distribution. Compared with Local-SMOTE, FedGAN-IDS was improved by 18.6% with IID on the synthetic dataset, which proves that federated synthesis is specifically valuable in situations with incomplete or skewed local data distributions.
This robustness is aided by the adaptive quality weighting mechanism, which under-weights updates posted by nodes with non-representative local distributions that lead to poor local training results.

5.4. Privacy-Utility Trade-off

Figure 1 (detailed in the text) shows the correlation between the privacy budget and detection performance. Macro-F1 grew almost exponentially with $\varepsilon = 5.0$. With a privacy level of $\varepsilon = 3.0$, FedGAN-IDS was able to operate at 92.7% of the performance when $\varepsilon = 10.0$ (weak privacy) at the cost of much more serious privacy guarantees.

Table 4: Privacy-Utility Trade-off Analysis			
Privacy Budget (ε)	Macro-F1	AUC	MI Attack Accuracy
1.0	0.542 ± 0.028	0.878 ± 0.014	50.3% ± 0.8%
2.0	0.612 ± 0.024	0.903 ± 0.011	51.2% ± 1.1%
3.0	0.671 ± 0.020	0.924 ± 0.009	52.4% ± 1.3%
5.0	0.698 ± 0.018	0.934 ± 0.008	54.8% ± 1.6%
10.0	0.714 ± 0.016	0.941 ± 0.007	58.7% ± 2.1%
∞ (no DP)	0.724 ± 0.015	0.946 ± 0.006	67.3% ± 2.8%

Empirical privacy membership inference attack accuracy trains an attacker model to identify whether specific samples are part of the training data. Random guessing has an accuracy of 50% for correct outcomes. Membership inference is successful on 52.4% of the ε = 3.0, which is considered to be a good empirical privacy protection and is in agreement with the theoretical guarantees. However, this assessment can only go as far as basic membership inference, and better attacks, such as data reconstruction, were not tried and might expose other weaknesses.

5.5. Communication Efficiency

Table 5 presents the communication cost of the federated methods in terms of the number of bytes transferred during training.

Table 5: Communication Cost Analysis			
Method	Parameters	Rounds	Total Communication
Federated-Classifier	847K	100	339 MB
FedGAN-IDS (Generator only)	1.2M	100	480 MB
FedGAN-IDS (Full)	2.4M	100	960 MB
Centralized Data Transfer	N/A	1	4.7 GB

FedGAN-IDS requires significantly less time to communicate than raw data to a central task, and the generator-only model (sending only generator parameters and none of the actual GAN) requires only approximately 10 times less time than the data transferred to a central point. The communication overhead of federated learning is feasible to implement in a realistic situation where the network connectivity between the edge nodes and aggregation server is reasonable.

5.6. Ablation Studies

Table 6 shows the ablation results for investigating the role of each FedGAN-IDS component (CIC-IDS-2017, five runs). Quality weighting provides an 8.7% absolute advantage in IID macro-F1 and a 11.2% advantage in non-IID and greater improvements when more than 70% of the nodes do not possess some or all minority classes. Differential privacy decreases the performance by 4.1% compared to non-privately trained models, and conditional

generation is necessary for synthesizing classes.

Table 6: Ablation Study Results (synthetic dataset)

Configuration	Macro-F1 (IID)	Macro-F1 Change
Full FedGAN-IDS	0.6704	Baseline
Without Quality Weighting	0.6120	-8.7%
Without Differential Privacy	0.6981	+4.1%
Unconditional Generation	0.6187	-7.7%

There is a 7.7% improvement in the quality weighting on the macro-F1, and the non-IID conditions are greater. The performance of the differential privacy mechanisms is approximately 4% lower than that of non-private training, which is a low price to pay to attain formal privacy assurances. Conditional generation is crucial for guiding the synthesis to target particular classes of attacks, and unconditional generation has significantly lower minority class performance.

6. Discussion

6.1. Interpretation of Results

FedGAN-IDS achieves its performance mainly because of the suggested quality-weighted aggregation mechanism. Ablation experiments (Table 6) indicate that disabling quality weighting causes an 8.7% (IID) and 11.2% (non-IID) decrease in macro-F1, which shows that the major part of the enhancement is caused by the adaptive client weighting. In extreme non-IID conditions, when some type of attack is completely absent at >70% of nodes, a standard FedAvg trained on GAN parameters can cause mode collapse rapidly; however, the suggested equilibrium-based weighting maintains the fidelity of global generators and generates usable synthetic samples of all minority classes.

This is because it can detect extremely rare attacks, such as the Heartbleed and Infiltration, which rely on collaborative synthesis: nodes that have not observed these attacks can still generate believable synthetic instances of the globally trained conditional generator and can then be conditioned into the learning of decision boundaries that would otherwise fail.

The resistance to non-IID data distributions is a practically significant observation for actual applications. Enterprise or IoT network segments have a systematic distribution of traffic features across various types of devices, user behavior patterns, and vulnerabilities to certain threat vectors. FedGAN-IDS can survive this heterogeneity, whereas local methods cannot. Nevertheless, a macro-F1 of 0.6704 does not show significant improvements over the baseline but is modest in comparison with non-private SOTA algorithms such as FedGen (0.728 ± 0.0047), which is an inherent privacy-utility trade-off in restricted environments.

6.2. Privacy Considerations

FedGAN-IDS offers the benefits of differential privacy that respond to regulatory issues, including the data protection provisions of the GDPR, which require the protection of data by technical means. The model provides substantial defense against membership inference (learning whether or not some flows in the network were in training) and attribute inference (learning sensitive network traffic properties based on model parameters) with $\varepsilon = 3.0$.

However, there is a performance cost when it comes to privacy. The difference in macro-F1 between privacy-enabled and non-privacy training of 4% is indicative of the privacy-utility trade-off inherent to the process of differentiating privacy. Privacy budgets can be adjusted according to the regulatory requirements and threat models of an organization, with ε values being larger where operational efficiency is more important than high privacy thresholds.

The empirical study of privacy through a membership inference attack offers supplementary evidence for selecting theoretical protection and turning it into practice. The accuracy of the attacks is near random guessing (52.4% at $\varepsilon = 3.0$), which implies that adversaries cannot be sure of the membership of the training sets despite complete knowledge of the model parameters.

6.3. Limitations

My findings have several limitations that can be used to guide future studies. First, I did not assess actual distributed network deployments but simulated federated environments. Despite the realistic traffic properties found in the CIC-IDS datasets, the partitioning schemes may not be sufficient to fully represent the heterogeneity and time dynamics of production networks. It is possible to run CIC-IDS-2017 using an artificial dataset to simulate class imbalance and non-IID settings; however, it may not mirror network variability in the real world. In particular, network traces obtained in practice include random noise as well as non-stationary time dynamics that are not present in synthetic generation. The next step in this study should be aimed at confirming the suggested framework with raw packet traces of various network settings (e.g., UNSW-NB15 or actual enterprise traces) to determine its resilience to real traffic anomalies and concept drift.

Second, the framework assumes the presence of honest-but-curious players who follow the protocol but can seek to make an effort to draw some inference about what they learn. A competitor who maliciously changes their model may deteriorate the quality of the models or place backdoors, which lies beyond the scope of the proposed work and constrains the application to adversarial contexts. Third, the hyperparameters and GAN architecture were optimized on the evaluation data, and it is not clear whether they can be transferred to networks with dramatically different traffic characteristics. It may be necessary to adjust the architecture or search for hyperparameters because of deployment in new environments. Moreover, sensitivity analysis was not conducted on the hyperparameters that were tuned (e.g., λ=10, β=10), which could affect the information about their robustness. Although my experiments indicate that $\beta \in [5, 15]$ is a range that allows stable results, extrapolations beyond and below this interval resulted in either not filtering erratic customers enough, or throwing out valid updates too often.

6.4. Practical Deployment Considerations

The biggest limitation is the edge node computational load, in which local GANs training is considerably more resource-intensive than regular classifier training. To

manage this in resource-constrained IoT devices, the framework may be modified to use lightweight generator architectures (e.g., MobileGAN) or a split-learning strategy. With a split-learn system, the burdensome gradient calculation in the generator might be outsourced to a secure edge gateway, and all the end devices are required to perform lightweight discrimination or inference, thereby saving battery life and processing capabilities of the IoT nodes. Moreover, the strength of the model in changing network conditions is a major factor. My preliminary sensitivity analysis suggests that the mechanism of aggregation can withstand moderate changes in the rates of client participation; however, it might worsen if the population of active clients becomes less than some critical level (i.e., less than 30% participation).

7. Conclusion and Future Work

This study introduces FedGAN-IDS, a quality-weighted aggregation-based federated conditional GAN, to address the challenges of a significant class imbalance and privacy concerns in distributed intrusion detection. The synthetic dataset that simulates the CIC-IDS-2017 benchmark with different privacy levels proposed a macro-F1 score of 0.6704, which is 0.1049 pp higher (18.5% relative improvement) than local-only training and 0.1074 pp higher than the federated classifier on minority classes, respectively. This can be attributed to such improvements that can be attributed largely to the mechanism of quality-weighted aggregation which mitigates training instability in federated setting.

The next step in the future of this research is to expand the framework to more extensive and comprehensive threat models. In particular, I intend to make the quality-weighting scheme immune to Byzantine enemies that may deliberately create updates to evade detection. This would be possible by incorporating strong aggregation statistics (the geometric median or Krum functions) into the equilibrium score to select and remove malicious outliers. Future work will cover the application of the LightGBM technique called the Exclusive Feature Bundle into the neural network. This would be scaled by projecting mutually exclusive sparse features into dense, lower-dimensional embedding layers to reduce the input dimensionality and computational complexity of the generator.

Declarations

Funding: This study did not receive any external funding.

The authors declare no conflicts of interest.

Data Availability: The CIC-IDS-2017 and CIC-IDS-2018 data sets can be publicly accessed on the site of the Canadian Institute of Cybersecurity at the URL https://www.unb.ca/cic/datasets/

Code Availability: The source code publicly available at https://github.com/zainabaabdulazeez/Integration-of-Generative-Adversarial-Networks-/blob/f5e415b13d1f3a1458fb4e39add580f99acb945f/Integration

References

[1] A. Khraisat and A. Alazab, “A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges,” *Cybersecurity*, vol. 4, no. 1, p. 18, Mar. 2021, doi: 10.1186/s42400-021-00077-7.

[2] A. Cormack, “Processing Data to Protect Data: Resolving the Breach Detection Paradox,” *SCRIPTed*, vol. 17, no. 2, pp. 197–237, Aug. 2020, doi: 10.2966/scrip.170220.197.

[3] A. Abdelkhalek and M. Mashaly, "Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning," *J. Supercomput.*, vol. 79, no. 10, pp. 10611–10644, Jul. 2023, doi: 10.1007/s11227-023-05073-x.

[4] J. He, X. Wang, Y. Song, Q. Xiang, and C. Chen, "Network intrusion detection based on conditional wasserstein variational autoencoder with generative adversarial network and one-dimensional convolutional neural networks," *Appl. Intell.*, vol. 53, no. 10, pp. 12416–12436, May 2023, doi: 10.1007/s10489-022-03995-2.

[5] S. Alabdulwahab, Y.-T. Kim, and Y. Son, "Privacy-Preserving Synthetic Data Generation Method for IoT-Sensor Network IDS Using CTGAN," *Sensors*, vol. 24, no. 22, Art. no. 22, Jan. 2024, doi: 10.3390/s24227389.

[6] M. Rasouli, T. Sun, and R. Rajagopal, "Fedgan: Federated generative adversarial networks for distributed data," *ArXiv Prepr. ArXiv200607228*, 2020, doi: 10.48550/arXiv.2006.07228.

[7] M. Jamoos, A. M. Mora, M. AlKhanafseh, and M. AlKhanafseh, "A New Data-Balancing Approach Based on Generative Adversarial Network for Network Intrusion Detection System," *Electronics*, vol. 12, no. 13, p. 2851, 2023, doi: 10.3390/electronics12132851.

[8] X. Mu *et al.*, "FedProc: Prototypical contrastive federated learning on non-IID data," *Future Gener. Comput. Syst.*, vol. 143, pp. 93–104, Jun. 2023, doi: 10.1016/j.future.2023.01.019.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-Sampling Technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002, doi: 10.1613/jair.953.

[10] Z. Fan, S. Sohail, F. Sabrina, and X. Gu, "Sampling-Based Machine Learning Models for Intrusion Detection in Imbalanced Dataset," vol. 13, no. 10, p. 1878, 2024, doi: 10.3390/electronics13101878.

[11] D. Elreedy and A. F. Atiya, "A Comprehensive Analysis of Synthetic Minority Oversampling Technique (SMOTE) for handling class imbalance," *Inf. Sci.*, vol. 505, pp. 32–64, Dec. 2019, doi: 10.1016/j.ins.2019.07.070.

[12] W. Tian, Y. Shen, N. Guo, J. Yuan, and Y. Yang, "VAE-WACGAN: An Improved Data Augmentation Method Based on VAEGAN for Intrusion Detection," *Sensors*, vol. 24, no. 18, p. 6035, 2024, doi: 10.3390/s24186035.

[13] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 1, p. e4150, Jan. 2021, doi: 10.1002/ett.4150.

[14] I. J. Goodfellow *et al.*, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2014.

[15] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling Tabular data using Conditional GAN," in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2019.

[16] Mihai Gabriel Constantin *et al.*, "Exploring Generative Adversarial Networks for Augmenting Network Intrusion Detection Tasks | ACM Transactions on Multimedia Computing, Communications, and

Applications," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 21, no. 1, 2024, doi: 10.1145/3689636.

[17] Z. Deng, A. Torim, S. Ben Yahia, and H. Bahsi, "Generative AI in Intrusion Detection Systems for Internet of Things: A Systematic Literature Review," *IEEE Open J. Commun. Soc.*, vol. 6, pp. 4689–4717, 2024, doi: 10.1109/OJCOMS.2025.3573194.

[18] Z. Arafat, O. V. Yudina, and Z. A. Abdulazeez, "Generative adversarial networks in cyber security: Literature review," *Russ. Technol. J.*, vol. 13, no. 5, pp. 7–24, 2025, doi: 10.32362/2500-316X-2025-13-5-7-24.

[19] B. Yurdem, M. Kuzlu, M. K. Gullu, F. O. Catak, and M. Tabassum, "Federated learning: Overview, strategies, applications, tools and future directions," *Heliyon*, vol. 10, no. 19, p. e38137, Oct. 2024, doi: 10.1016/j.heliyon.2024.e38137.

[20] T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, and A.-R. Sadeghi, "DÏoT: A Federated Self-learning Anomaly Detection System for IoT," *2019 IEEE 39th Int. Conf. Distrib. Comput. Syst. ICDCS*, pp. 756–767, Jul. 2019, doi: 10.1109/ICDCS.2019.00080.

[21] Viraaji Mothukuri, Reza M. Parizi, Seyedamin Pouriyeh, Yan Huang, Ali Dehghantanha, and Gautam Srivastava, "A survey on security and privacy of federated learning," *Future Gener. Comput. Syst.*, vol. 115, no. C, pp. 619–640, 2021, doi: 10.1016/j.future.2020.10.00.

[22] D. Man, F. Zeng, W. Yang, M. Yu, J. Lv, and Y. Wang, "Intelligent Intrusion Detection Based on Federated Learning for Edge-Assisted Internet of Things," *Secur. Commun. Netw.*, vol. 2021, no. 1, p. 9361348, 2021, doi: 10.1155/2021/9361348.

[23] C. Qiu, Z. Wu, H. Wang, Q. Yang, Y. Wang, and C. Su, "Hierarchical Aggregation for Federated Learning in Heterogeneous IoT Scenarios: Enhancing Privacy and Communication Efficiency," *Future Internet*, vol. 17, no. 1, Jan. 2025, doi: 10.3390/fi17010018.

[24] J. Yu, G. Wang, N. Shi, R. Saxena, and B. Lee, "A Multi-View-Based Federated Learning Approach for Intrusion Detection," *Electronics*, vol. 14, no. 21, Oct. 2025, doi: 10.3390/electronics14214166.

[25] L. Yang, J. He, Y. Fu, and Z. Luo, "Federated Learning for Medical Imaging Segmentation via Dynamic Aggregation on Non-IID Data Silos," *Electronics*, vol. 12, no. 7, Apr. 2023, doi: 10.3390/electronics12071687.

[26] H. Salman, C. Zaki, N. Charara, S. Guehis, J.-F. Pradat-Peyre, and A. Nasser, "Knowledge distillation in federated learning: a comprehensive survey," *Discov. Comput.*, vol. 28, no. 1, p. 145, Jul. 2025, doi: 10.1007/s10791-025-09657-4.

[27] H. Wang, L. Muñoz-González, D. Eklund, and S. Raza, "Non-IID data re-balancing at IoT edge with peer-to-peer federated learning for anomaly detection," in *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, in WiSec '21. New York, NY, USA: Association for Computing Machinery, Jun. 2021, pp. 153–163. doi: 10.1145/3448300.3467827.

[28] Xin-Chun Li and De-Chuan Zhan, "FedRS: Federated Learning with Restricted Softmax for Label Distribution Non-IID Data," *Proc. 27th ACM SIGKDD Conf. Knowl. Discov. Data Min.*, pp. 995–1005, 2021, doi: 10.1145/3447548.3467254.

[29] P. Zhao, S. Guo, Y. Li, S. Yang, and X. Ren, "FedGen: Personalized federated learning with data generation for enhanced model customization and class imbalance," *Future Gener. Comput. Syst.*, vol. 164, no. 107595, Mar. 2025, doi: 10.1016/j.future.2024.107595.

[30] A. N. Khan, A. Rizwan, R. Ahmad, Q. W. Khan, S. Lim, and D. H. Kim, "A precision-centric approach to overcoming data imbalance and non-IIDness in federated learning," *Internet Things*, vol. 23, p. 100890, Oct. 2023, doi: 10.1016/j.iot.2023.100890.