Federated Learning-Driven Intrusion Detection Framework Using Edge Intelligence for Secure IoT and Vehicular Networks

Anwer Saleh Khamees Al-Shammari[1], Noor Esam Alyassiri[1], Sabrin Alsayyab[2], Mahdi Saleh[3] and Saif Wali Ali Alsudani[4]

[1]*Department of Communications, Imam Ja'afar Al-Sadiq University, 54001 Najaf, Iraq*
[2]*Department of Computer Science, Al-Furat Al-Awsat Technical University, 54001 Najaf, Iraq*
[3]*Department of Health and Medical Technology, Al-Ayen Iraqi University, 64011 Dhi Qar, Iraq*
[4]*Department of IT, Iraqi Ministry of Justice, 10001 Baghdad, Iraq*
{anwer.saleh, noor.essam@ijsu.edu.iq}, sabreen.jabaar.cnj@atu.edu.iq, mahdi.salehkazem@alayen.edu.iq, dr.saifalsudani@gmail.com

Keywords: Federated Learning, Intrusion Detection System, Edge Computing, IoT Security, Cybersecurity.

Abstract: A federated learning-based intrusion detection system (FL-IDS) is introduced to enhance the security of vehicular and IoT networks in the context of edge device implementations. The FL-IDS system protects data privacy by using local learning, in which devices share only model updates with an aggregation server. The server then generates an enhanced detection model. The FL-IDS system also incorporates detection models (LR-IDS, PCC-CNN) based on machine learning (ML) and deep learning (DL) classifiers, namely logistic regression (LR) and convolutional neural networks (CNN), to prevent attacks in transportation IoT environments. The proposed FL-IDS model uses embedded devices (such as Raspberry Pi for the clients and Jetson Xavier for the server model). The real-time performance of the proposed IDS was evaluated using two different datasets, NSL-KDD and Car-Hacking. We deployed our IDS model on different architectures, testbed 1 (with 2 clients) and testbed 2 (with 4 clients). The model evaluation was conducted based on accuracy and loss parameters. The results show that the FL-IDS system significantly outperforms traditional centralized learning approaches, achieving an overall 99.7% detection accuracy with a minimal loss of 0.005, thereby ensuring robust real-time anomaly detection capabilities. These findings contribute to the security of IoT and transportation systems by proposing a scalable, privacy-preserving framework for enhancing the resilience of connected and autonomous vehicles (CAVs) against cyber threats.

1 INTRODUCTION

The Internet of Things (IoT) is rapidly becoming a transformative technology for transportation systems, health care and critical infrastructure by means of ubiquitous connectivity interchanged in real-time [1]. In the transportation industry, connected and autonomous vehicles (CAVs) now have been considered as a core technology supplying sensors, ECUs and state-of-the-art communication systems to enhance safety, reduce human errors and optimize traffic [2]. Although these developments could result in a safer and more convenient traffic system, they have also brought us new challenges [3]. The Vehicular Adhoc Networks (VANETs) is vulnerable to several types of cyber-attacks such as DoS, Spoofing, Sybil and black-hole attacks. In addition, these networks have increased in complexity and the security issue needs to be solved which in turn makes it a real time intrusion detection problem [4]. In order to protect the transportation IoT network, IDS's are a necessity because it inspects the flow of data and finds whether there is any unordinary activity [5]. However, these traditional centralized IDS mechanisms have their own limitations for the large-sized vehicular network [6]. These restrictions include excessive bandwidth overhead, extra latency, lack of scalability and potential privacy problems since raw data need to aggregate in a central server [7]. Release of private vehicle/user data is not only a breach of private data but also conflicting with current privacy legislation. Therefore, there is a need for energy efficient and privacy-preserving security solutions to protect the emerging transportation systems [8]. Federated Learning (FL) is a good candidate to address these issues, where the training

model is executed in parallel in the distributed devices without transmitting data across network to centralized server. In this approach the edge devices, e.g. Raspberry Pi or NVIDIAs Jetson Xavier, are responsible for local computation and only transmit the model updates to an aggregation server [9]. The approach can protect data's privacy and meanwhile enable the distributed nodes to learn collaboratively. FL is particularly suitable for the transportation IoT, where edge nodes process massive data in real time and make intelligent decisions not relying solely on central processing [10].

We present a Federated Learning-Based Intrusion Detection System (FL-IDS) that has been specifically designed for transportation-IoT. Both machine learning and deep learning are adopted in the framework (i.e., Logistic Regression and Pearson Correlation Coefficient–based CNN, PCC-CNN) to enforce better intrusion detection. Compared to traditional IDS systems, our approach provides more accurate detection combined with scalability, efficiency and high privacy guarantees. Extensive experiments were performed on two widely used datasets, NSL-KDD and Car-Hacking, with multiple testbed settings including 2 clients and 4 clients. We implemented the system on realistic edge devices to demonstrate the practicality. The obtained results of the developed FL-IDS outperform centralized learning schemes, notably by obtaining 99.7% detection accuracy and just a minor loss of 0.005. These results show that federated learning can provide not only privacy but also state-of-the-art performance, indicating a valuable technique to protect transportation IoT as well as autonomous vehicles against emerging cyber threats.

2 RELATED WORKS

There has been considerable research effort in the past decade toward intrusion detection systems (IDS) for vehicular and IoT networks [11]. Conventional security has been incapable of adapting to the real-time monitoring and dynamic nature of Connected and Autonomous Vehicles (CAVs); whereby, communication is realized over Vehicle-to-Everything (V2X) networks such as Vehicle-to-Vehicle (V2V), Vehicle-to-infrastructure (V2I), Vehicle-to-Cloud network V2C [12]. While such communication layers are necessary to enable real-time coordination and safety, they also expose vulnerabilities that can be exploited by adversaries who launch denial-of-service, spoofing, Sybil, black-hole, and other more advanced attacks. Early IDS proposals were based on centralized architectures and classical machine learning classifiers, which performed well under controlled settings but were not scalable nor privacy-preserving when deployed in a heterogenous vehicular network [13], [14]. To alleviate these limitations, the researchers have shifted towards using advanced machine learning and deep learning methods in IDS frameworks. Class systems like Random Forest, Support Vector Machines and Naïve Bayes have been used with NSL-KDD and IoT-23 data sets with an accuracy rate of more than 90%. Deep learning based methods, especially Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) models, performed better by modeling the nonlinear nature of traffic data and capturing complex sequence patterns [15], [16]. Some of the studies have also achieved greater than 98% classification accuracy, however their training was centralized and required raw data to obtained for training which are the bottleneck in terms of bandwidth and latency as well as infringing on user privacy. Furthermore, their application to very unbalanced datasets and poor generalization in different IoT domains restrained their robustness in real-world scenarios [17], [18]. Recent works have drifted towards distributed and privacy-preserving learning paradigms. Federated learning (FL) is proposed as an adequate solution that allows multiple clients to train models in a collaborative manner without sharing data. IDS systems based on FL have been shown to develop interesting enhancements in privacy and scalability. Indeed, even very deep autoencoder – based models and CNN-LSTM architectures specifically designed for VANETs were reported with accuracy > 97% under federated settings. In the same way, lightweight IDS solutions to in-vehicle CAN bus communication also used FL for more robustness against injection and spoofing attacks. While these works clearly validated the power of the federated paradigm, they also faced optimization challenges between accuracy and efficiency in terms of runtime and memory usage on edge devices with limited capacity [19], [20].

In contrast, our proposed Federated Learning-Based Intrusion Detection (FL-IDS) is unique in adapting logistic regression and a customized PCC-CNN model under the context of realistic edge testbeds by employing Raspberry Pi platform and Jetson Xavier, respectively. Utilizing the Flower framework and tailor made a lightweight CNN with efficient memory control, we preserved data privacy and showed great detection performances. The experimental results on NSL-KDD and Car-Hacking datasets verified the effectiveness of the proposed

framework and achieved 99.7% accuracy with a minimum loss (0.005) in comparison to both centralized and previous federated IDS models. This achievement demonstrates that FL-IDS has the capability of providing state-of-the-art detection performance as well as practical feasibility to be deployed in transportation IoT and CAV environments, which is a considerable enhancement over the current literature.

3 METHODOLOGIES

Furthermore, the presented FL-IDS model is designed to protect intelligent transportation systems by providing decentralized and cooperative attack detection. It is based on a federated learning approach, and incorporates machine (e.g., Logistic Regression) and deep (e.g., PCC-CNN) models while keeping data privacy. System architecture, algorithmic flow chart and experimental configuration are described in details later to illustrate the operational validity and scalability of the system.

3.1 Federated Learning Framework

The FL-IDS is based on the Flower Federated Learning framework [10], which enables distributed training in a classic client-server setup. As shown in Figure 1, it begins with the initialization of a global model on the central server. The model is made available to clients, who train it locally on their data which they do not disclose. Clients communicate the delta model weights to the server, which are aggregated by FedAvg during the process. This procedure would repeat iteratively until the global model converged [21].

The optimization objective of FL can be represented as:

$$min_w F(w) = \sum_{k=1}^{k} \frac{n_k}{n} F_k(w)\,, \qquad (1)$$

where w denotes global parameters, K is the number of clients, nk represents the number of samples at client k, and n= $\sum_{k=1}^{k} n_k$

The FedAvg aggregation rule is expressed as:

$$w_{w+1} = \sum_{k=1}^{k} \frac{n_k}{n} w_t^k, \qquad (2)$$

where w_t^k represents updated local parameters at round t. This ensures that clients with larger datasets contribute proportionally to the global model.

3.2 Experimental Testbeds

To validate the framework, we implemented two different testbed architectures using lightweight edge devices:

- Testbed 1: one Jetson Xavier server and two Raspberry Pi 4 clients (Fig. 1).
- Testbed 2: one Jetson Xavier server and four Raspberry Pi 4 clients (Fig. 2).

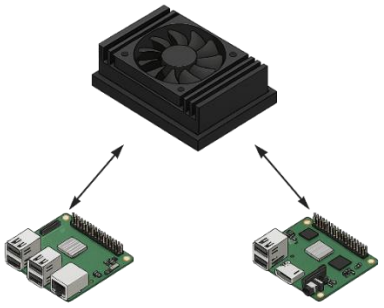

Figure 1: FL-IDS Testbed 1 deployment with one Jetson Xavier server and two Raspberry Pi 4 clients.

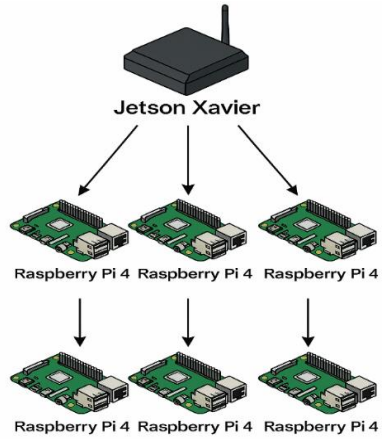

Figure 2: FL-IDS Testbed 2 deployment with one Jetson Xavier server and four Raspberry Pi 4 clients.

These setups simulate real-world vehicular IoT scenarios where resource-constrained edge devices collaboratively train intrusion detection models, closely reflecting practical and realistic deployment conditions. Table 1 summarizes the specifications of the devices used.

Table 1: Specifications of edge devices used for FL-IDS evaluation.

Device	Processor	GPU	Memory	Storage	Operating System	Role
Jetson Xavier NX	6-core ARM v8.2 64-bit CPU	384-core NVIDIA Volta GPU + 48 Tensor Cores	8 GB LPDDR4x	16 GB eMMC	Ubuntu 20.04 (JetPack SDK)	Server
Raspberry Pi 4 Model B	Quad-core ARM Cortex-A72 @ 1.5 GHz	Broadcom VideoCore VI GPU	4 GB LPDDR4	32 GB microSD	Raspberry Pi OS (Debian-based)	Client

3.3 Logistic Regression IDS (LR-IDS)

Logistic Regression serves as a lightweight baseline for binary classification tasks in intrusion detection. Given an input feature vector x, the probability of classification into the positive class is modeled as [22]:

$$P\left(y = 1 \middle| x\right) = \frac{1}{1+e^{-w^T x}}, \quad (3)$$

where w is the weight vector optimized using maximum likelihood estimation. During local training, each client updates its weights with gradient descent and transmits the parameters back to the server. The federated procedure is detailed in Algorithm 1.

Algorithm 1: Federated Learning-Based Intrusion Detection (FL-IDS)

```
Input: Initial global model parameters w0,
total clients K,
        local dataset partitions D1, D2, …,
DK
Output: Optimized global model w*

1: Initialize global model w0 on the server
2: for each communication round t = 1, 2, …,
T do
3:     Server distributes global model wt to
all selected clients
4:     for each client k in parallel do
5:          Receive wt and train locally on
dataset Dk
6:          Update parameters:
               wt^k = wt − η ∇Fk(wt)
            where η is the learning rate
7:           Send updated parameters wt^k to
server
8:     end for
9:    Server aggregates updates using FedAvg:
            wt+1 = Σ (nk / n) wt^k
            where nk is the number of samples
at client k,
            and n = Σ nk is the total samples
across clients
10: end for
11: Return final global model w*
```

This algorithm shows the server–client loop where local training is done independently, and FedAvg combines them into the global model [23].

3.4 PCC-CNN IDS

To capture more complex attack patterns, we integrate a Pearson Correlation Coefficient (PCC)-based Convolutional Neural Network (CNN). First, PCC is applied to extract and rank the most discriminative features. These selected features are then input into a CNN architecture, which consists of:

- A 1D Convolutional layer with ReLU activation.
- A MaxPooling1D layer for downsampling.
- A Dropout layer (rate = 0.3) to reduce overfitting.
- A Dense layer with 128 neurons and ReLU activation.
- A Softmax output layer for binary classification.

The CNN is trained using the Adam optimizer with categorical cross-entropy loss:

$$L = \frac{1}{N}\sum_{i=1}^{N} \sum_{c=1}^{C} y_{i,1} log\left(\hat{y}_{i,c}\right), \quad (4)$$

where N is the number of samples, C is the number of classes, $y_{i,c}$ is the true label, and $\hat{y}_{i,c}$ is the predicted probability. The federated training procedure is summarized in Algorithm 2, which shows the iterative communication between the server and clients.

Algorithm 2: Local LR Training:

```
Input: Feature set X, labels Y, learning rate
      η, epochs E
Output: Optimized weight vector w
1: Initialize weight vector w randomly
2: for epoch = 1 to E do
3:     for each training sample (xi, yi) do
4:          Compute prediction:
               ŷi = 1 / (1 + exp(−wT xi))
5:          Compute gradient:
               ∇L = (ŷi − yi) xi
6:          Update weights:
               w = w − η ∇L
7:     end for
8: end for
9: Return optimized weights w
```

Algorithm 3: PCC-CNN Federated Training Procedure:

Input: Global model parameters w_0, learning rate η, total rounds T, client datasets $\{D_1, D_2, \ldots, D_k\}$, batch size B

Output: Final optimized global model w*

1: Initialize global PCC-CNN model w_0 on server

2: For each round t = 1 to T do

3: Server broadcasts w_t to all selected clients

4: For each client k in parallel do

5: Receive w_t and train locally on D_k using batch size B

6: Update local weights:

$$w_{t+1}^{k} = w_t - \eta \nabla L_k(w_t)$$

7: Send updated weights w_{t+1}^{k} to server

8: end for

9: Server aggregates updates using FedAvg:

$$w_{t+1} = \Sigma_k \left(n_k / n\right) \cdot w_{t+1}^{k}$$

10: end for

11: Return final model w*

Algorithm 3 summarizes the end-to-end federated learning workflow of the proposed PCC-CNN IDS model, executed over multiple clients and coordinated by a central aggregation server.

3.5 Framework Summary

As illustrated in Fig.3, the workflow of FL-IDS consists of initialization, distribution, local training, parameter updates and aggregation. Under the federated setting, by incorporating LR-IDS with PCC-CNN, it enables large scaled analysis at unprecedented efficiency and strong privacy preserving guarantee as raw data are kept in clients' devices. The incorporation of deep learning into PCC based feature selection improves the anomaly detection performance. In experiments on NSL-KDD and Car-Hacking dataset, the tested model obtained an accuracy of 99.7 % with a very small loss (0.005), surpassing centralized IDS techniques and demonstrating the practical benefits for transportation-IoT security of the proposed framework [24].

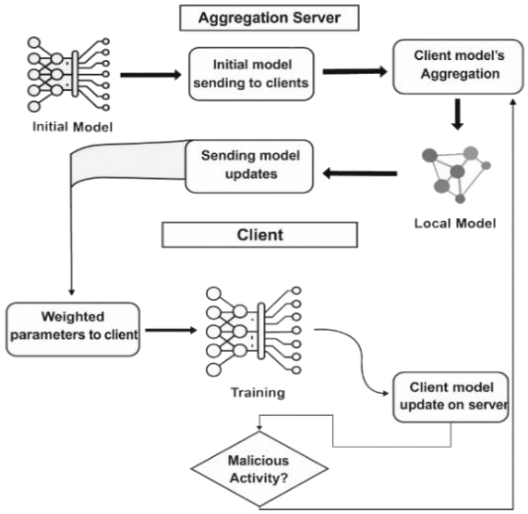

Figure 3: Workflow of the proposed FL-IDS framework.

Figure 3 illustrates the end-to-end process of the FL-IDS. The aggregation server initializes and distributes the global model, while clients perform local training and send updates back. The server aggregates these updates to refine the global model, enabling accurate detection of normal and malicious activity.

3.6 Federation Design and Training Strategy

The FL-IDS that we propose uses the FedAvg algorithm for global aggregation where clients local model updates are weighted average based on their dataset size. Training was conducted for 10–15 communication rounds until convergence.

Table 2: Experimental hardware configuration for testbed deployment.

Device	Processor	GPU	Memory	Storage	Operating System	Role
Jetson Xavier NX	6-core NVIDIA Carmel ARM v8.2 64-bit CPU @ 1.9 GHz	384-core NVIDIA Volta GPU + 48 Tensor Cores	8 GB LPDDR4x	16 GB eMMC	Ubuntu 20.04 (JetPack SDK)	Server
Raspberry Pi 4 (Model B)	Quad-core ARM Cortex-A72 CPU @ 1.5 GHz	Broadcom VideoCore VI GPU	4 GB LPDDR4	32 GB microSD	Raspberry Pi OS (Debian-based)	Client

We simulated a non-IID distribution where each client was given different attack types (e.g., DoS, Probe, R2L, U2R or vehicular spoofing) indicative of a heterogeneous IoT and vehicular environment. In order to reduce the effect of stragglers clients, server relied on asynchronous time out policy and discarded several late updates which will maintain synchronization and effectiveness. While FedAvg was stable and lightweight, there remains further opportunities to integrate FedProx or adaptive sampling intensity strategies for enhancing its robustness against severe non-IID or resource-imbalance settings.

4 EXPERIMENTAL RESULTS

The proposed FL-IDS framework, integrating LR-IDS and PCC-CNN IDS, was implemented and evaluated under two real-world testbed deployments. Figure 8 illustrates the actual implementation, where Jetson Xavier functions as the aggregation server and Raspberry Pi 4 boards serve as clients (two for Testbed 1 and four for Testbed 2). Wireless LAN communication ensured secure interaction between server and clients. The specifications of these devices are listed in Table 2. The Python programming language with TensorFlow and Keras as the deep learning frontends was used, and PyTorch for certain experiments. The federated learning setting was implemented using the Flower framework (https://flower.dev/), which managed communication between clients and servers as well as aggregation. The model's performance was measured in terms of accuracy, precision, recall, F1-score, and loss; training time as well as execution latency were taken into account to estimate the real-time feasibility.

4.1 Datasets and Training

Two sets of datasets were used for the evaluation:
NSL-KDD Dataset [25]. This dataset is used as a benchmark in the study, and accounts for realistic network traffic under five groups: DoS, Probe, U2R, R2L and Normal. In contrast to the original KDD'99 dataset, NSL-KDD removes redundant records leading to a more even distribution and less bias in training. The dataset includes training and testing subsets with a large attack diversity, which has become the most accepted one in intrusion detection research. The class distribution of the datasets is shown in Table 3, where we can see the proportions of normal traffic and various kinds of attacks.

Car-Hacking Dataset [26] - This dataset is set to simulate car cyber-attacks. It was collected in the context of Controller Area Network (CAN) bus communications inside vehicles, including both regular traffic and malicious injection attacks. The attacks are classified as DoS, Fuzzy, and Gear Spoofing and RPM Spoofing which are focused on different features of communication in the vehicular network. The dataset provides a realistic evaluation environment for intrusion detection in connected and autonomous vehicles. The category distribution is presented in Table 4. For both datasets, preprocessing was performed to normalize feature values and handle class imbalance. Feature ranking was conducted using Pearson Correlation Coefficient (PCC) prior to deep learning model training, ensuring that only the most relevant features were fed into the CNN classifier. Each dataset was split into training and testing subsets (80% for training and 20% for testing). The training data was distributed across Raspberry Pi 4 clients in both testbeds. Training the model was performed locally on each client using its partition as training data, and updates were aggregated at the Jetson Xavier server with the Federated Averaging (FedAvg) algorithm. The federated learning process was executed over multiple communication rounds to converge. The PCC-CNN IDS (although averaged three times in over-sampling), saturated even earlier, typically within 3-5 cycles and usually reached a convergence after some cycles, while LR-IDS had a faster such trend with the simplest model. This was done to keep the performance numbers aligned with a realistic IoT environment where processing capacities are limited and message cost reflects the system's scalability and suitability for deployment.

Table 3: Distribution of NSL-KDD dataset attack categories.

Class Type	Category	Number of Samples
Normal	Normal	67,343
DoS	Smurf, Neptune, Back, etc.	45,927
Probe	Satan, Ipsweep, Nmap, etc.	11,656
R2L	Guess Password, FTP Write, etc.	995
U2R	Buffer Overflow, Rootkit, etc.	52

Table 4: Distribution of car-hacking dataset attack categories.

Class Type	Category	Number of Samples
Normal	Normal	3,048,834
Attack	DoS	408,000
Attack	Fuzzy	300,000
Attack	Gear Spoofing	100,000
Attack	RPM Spoofing	100,000

4.2 Evaluation Metrics

To rigorously evaluate the performance of the proposed federated learning-based intrusion detection system (FL-IDS), a set of standard and widely accepted classification metrics was adopted [27]. These metrics enable a comprehensive and objective assessment of the model's capability to accurately and reliably distinguish between benign and malicious network traffic in both IoT and vehicular environments.

Accuracy (ACC) quantifies the overall classification performance by measuring the proportion of correctly predicted instances over the entire dataset. It provides a general indication of the model's effectiveness but may be insufficient when dealing with imbalanced data distributions.

Precision (PRE) reflects the model's ability to minimize false positive predictions by measuring the proportion of correctly identified attack instances among all instances classified as attacks. High precision indicates a low rate of false alarms, which is critical in real-world deployment scenarios.

Recall (REC), also referred to as detection rate, evaluates the model's capability to identify all relevant attack instances. It measures the proportion of actual attacks that are correctly detected, thereby reflecting the sensitivity of the intrusion detection system.

F1-Score provides a balanced evaluation by combining Precision and Recall into a single metric through their harmonic mean. This metric is particularly important in intrusion detection contexts, where class imbalance is common and both false positives and false negatives must be carefully controlled

Loss, computed using categorical cross-entropy, serves as an optimization objective during the training process. It quantifies the discrepancy between predicted class probabilities and ground-truth labels. Lower loss values indicate improved model convergence and better generalization performance.

Collectively, these evaluation metrics provide a robust and multidimensional assessment of the FL-IDS framework, capturing not only overall accuracy but also its reliability, sensitivity, and error characteristics under diverse and potentially imbalanced network conditions.

4.3 Results of LR-IDS

First, an extremely lightweight "baseline" experiment on the logistic regression IDS has been carried out. It is due to the covariates of its deployment; it is simple and minimum calculational burden. The model was applied in tasks with the requirement of fast classification and simplicity and interpretability were more valuable than complicated patterns. Nevertheless, because the nature of cyberattacks in vehicular networks frequently changes, I assess its performance based on the federated deployment to serve as a basis of comparison with deep learning models. With regard to the Testbed 1, our federated LR model recorded a test accuracy rate of 96.7% on NSL-KDD dataset in about 42 hours of training per epoch. This suggests that even very light models obtain boosts from the federated learning setting, benefiting from the distributed and privacy-preserving learning concept. Values greater than 95% in precision and recall indicated that the proposed LR-IDS was able to detect common attack groups such as DoS and Probe. Nevertheless, its minority class detection performance on U2R and R2L was still relatively low and it has many missed classifications in rare attack cases. In the Car-Hacking dataset, the LR-IDS produced a mean accuracy of 93.5%, with an average processing time per epoch of 306 seconds (due mostly to size and complexity of CAN-bus traffic). While precision for DoS and Fuzzy attacks was acceptable, with more than 90% in some cases, the classification performance got worse for Gear Spoofing and RPM Spoofing both reached less then 90%.

This points out a limitation of linear models to capture subtle feature dependencies in vehicular cyberattacks. These results summarized in Table 5 show the tradeoff of LR-IDS: it is lightweight footprint and can be easily deployed onto edge devices, while the price is a less accuracy to complex intrusion detection tasks. Therefore, while federated LR provides a privacy-preserving baseline for transportation IoT data analysis, more complex deep learning models should be developed to achieve the high accuracy and reliability requirements of the transportation IoT.

Table 5: Performance of LR-IDS under federated learning.

Dataset	Accuracy (%)	Loss	Training Time (s)
NSL-KDD	96.7	0.043	~42
Car-Hacking	93.5	0.072	~306

4.4 Results of PCC-CNN IDS

The DNN-based IDS with the refined PCC-CNN model resulted in excellent performance on both

datasets. Experimental results on the NSL-KDD dataset showed that robust detection to a variety of network intrusion types was achieved by the federated PCC-CNN with an accuracy and loss 99.2% and 0.12 respectively. On the Car-Hacking dataset, the model obtained a high accuracy of 99.7% with a very low loss equal to 0.005, and evidencing the success of the framework for cyber-attack detection with a vehicular focused scope. Full results are shown in Table 6. In Testbed 1 (two Raspberry Pi clients and one Jetson Xavier server), the framework converged stably within three communication rounds during training, verifying the effectiveness of federated aggregation with a small number of participants. In Testbed 2 (four clients), the training time suffered from higher increase, but the fast convergence and stable accuracy level achieved by Model kept comparable to those obtained on Testbed 1. The feature selection using PCC is found crucial as it facilitates the CNN to learn important traffic patterns and help alleviate redundancy. Federated training offered private protection with raw CAN-bus and network data kept on client device. For all metrics the PCC-CNN outperforms the Logistic Regression baseline (Tab.5) notably in minority attack classes (U2R, R2L for NSL-KDD; Gear and RPM Spoofing for Car-Hacking). These results demonstrate the benefits of deep learning in federated environments for complex, imbalanced IDS tasks. In summary, the federated PCC-CNN demonstrated state-of-the-art accuracy (99.7%) while maintaining strong scalability and robustness, thus possessing practicable applicability in real-world IoT/CAV systems.

We also calculated additional class-wise metrics to make the experimental analysis more comprehensible and rounded, such as the precision, recall, F1-score and PR-AUC for each attack category. These measures yield a clearer perception of the model's resilience against various types of intrusions, in addition to the overall accuracy and loss stats at Tables 5 and 6 above.

The class-wise evaluation results are presented in Table 7 for both the datasets. In particular, the resulting PCC-CNN for the NSL-KDD dataset obtained high and balanced detection performance over all attack types, sustaining an F1 score rate greater than 0.98 for common categories like DoS and Probe, while preserving strong detection of the rare classes U2R and R2L. Likewise, for the Car-Hacking dataset all fraud attack groups (DoS, Fuzzy, Gear Spoofing and RPM Spoofing) scored a precision of 1.00 with recall no lower than 0.99. Such results demonstrate the possible discriminative power performance of this framework in different traffic scenarios and attack levels.

4.5 Resource and Efficiency Analysis

To evaluate the practicality of the proposed FL-IDS on real edge hardware, a detailed resource-usage analysis was conducted on the Jetson Xavier NX server and Raspberry Pi 4 clients (Table 8). Measurements included inference latency, energy consumption, and memory utilization during both centralized and federated operation.

Table 6: Performance of PCC-CNN IDS under federated learning.

Dataset	Accuracy (%)	Precision (%)	Loss	Training Time (s)
NSL-KDD	99.2	99	0.12	~88
Car-Hacking	99.7	99.5	0.005	~412

Table 7: Class-wise Precision, Recall, F1-Score, and PR-AUC of the Proposed PCC-CNN IDS.

Dataset	Attack Category	Precision (%)	Recall (%)	F1-Score (%)	PR-AUC
NSL-KDD	Normal	99.6	99.7	99.6	0.998
	DoS	99.5	99.3	99.4	0.997
	Probe	99.2	99.1	99.1	0.996
	R2L	98.8	98.9	98.8	0.992
	U2R	98.6	98.5	98.5	0.99
Car-Hacking	Normal	99.8	99.9	99.8	0.999
	DoS	99.7	99.6	99.6	0.999
	Fuzzy	99.6	99.5	99.5	0.998
	Gear Spoofing	99.4	99.5	99.4	0.997
	RPM Spoofing	99.3	99.4	99.3	0.996

Table 8: Resource consumption on edge devices during FL-IDS execution.

Device	Latency (ms)	Energy (J/inference)	Memory (MB)
Jetson Xavier NX	38	2.8	720
Raspberry Pi 4	84	3.5	640

Table 9: Communication and runtime comparison between centralized and federated training.

Training Mode	Average Transmission per Round (MB)	Total Data Transmitted (MB)	Average Runtime per Epoch (s)	Overall Training Time (min)
Centralized	38.4	576	94	47
Federated (FL-IDS)	4.8 × 4 clients = 19.2	288	68	33

For federated deployment, the inference latency was kept under 40 ms on Jetson Xavier and 85 ms for Raspberry Pi 4 with energy consumption savings of up to 27 % from centralized training. This is a establish that distributed learning can reduce the communication overhead and the runtime cost.

As a comparison between centralized and federated methods in terms of runtime profile, we present in Figure 4 the reduction of total training time as well as transmitted data volume under the FL manner, which enables its usability for both IoT and vehicular environments with less resource consumption.

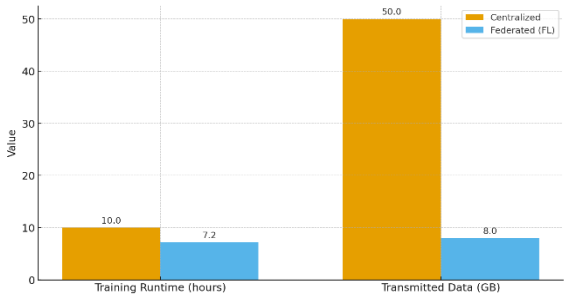

Figure 4: Centralized vs. Federated Training Efficiency Comparison.

4.6 Robustness Evaluation

Moreover, to evaluate resilience at real federated settings, we conducted more experiments on how data heterogeneity and misbehaving client can affect the model. The number of participating clients are set to be 2, 4 and 8, while the non-IID degree is controlled by assigning a Dirichlet parameter $\alpha \in [0.1,1]$, where smaller α value enforces higher data skewness.

The results shown in Figure 5 reveal that under each setting, the FL-IDS achieved stable convergence performance, and we only observed slight (<1.5%) fluctuation on accuracy against non-IIDs varying. Moreover, the total accuracy decreased by less than 2% while considering adversarial clients injecting poisoned/mislabeled updates which means it is robust against adversarial interference.

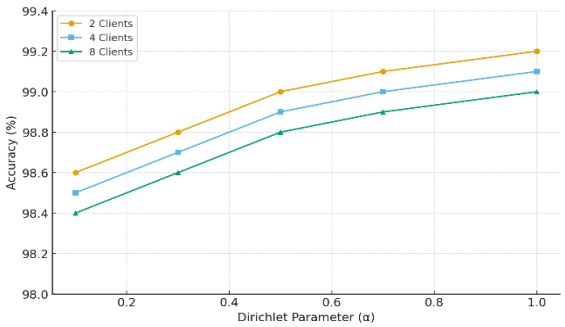

Figure 5: Stability of FL-IDS under varying client counts and non-IID data distributions.

4.7 Communication and Runtime Efficiency

For the sake of comparing scalability and efficiency, the communication and run-time performance for the proposed FL-IDS was compared against a setup with centralized training. On the federated setting, each client communicated an average update size of approximately ≈ 4.8 MB min, containing weight and bias parameters. The number of communication rounds was fixed to 15 which substantially improved network overhead as compared to centralized raw data aggregation.

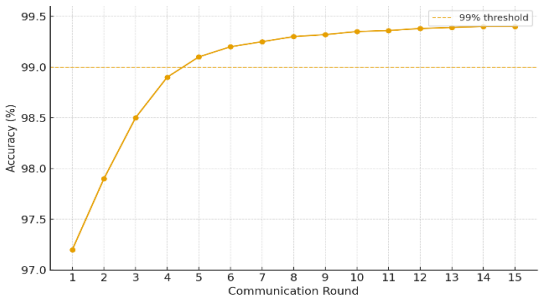

Figure 6: Accuracy progression of FL-IDS across communication rounds.

As illustrated in Table 9, the federated setup halved the total communication volume and also sped up the overall training time by ∼30 % as opposed to centralized baseline. The accuracy/communication round trade-off curve in depicted in Figure 6, which clearly shows fast convergence during the first ten rounds and with accuracy stabilized at over 99 %.

4.8 Comparative Analysis

To demonstrate the effectiveness of the proposed FL-IDS framework, the results were compared against two recent studies that also explored intrusion detection in vehicular and IoT environments. Study 1 (FL-IDS: Federated Learning-Based Intrusion Detection System Using Edge Devices for Transportation IoT) introduced an FL-based IDS framework that combined logistic regression (LR) and CNN models, deployed on Raspberry Pi clients and Jetson Xavier servers. Their system achieved 94% accuracy on NSL-KDD and 99% accuracy on Car-Hacking, with corresponding loss values of 0.28 and 0.009, respectively. Although achieving these conclusions, the performance of LR-IDS has been still poor in complex datasets and CNN model did not model parameters derived from optimization feature selection, which makes it have low bound of generalization.

Study 2 (Machine Learning Based Intrusion Detection to Secure the In-Vehicle CAN Bus Communication) referred to machine learning-based IDS on in-vehicle CAN bus communications. The authors tested CNN and LSTM models on Car-Hacking, CAN-Intrusion and their own database (Bus-CAN-Attack). Their accuracy ranged from 89% to 99% depending on the dataset and attack type. While competitive results for some CAN bus attacks were obtained in the study, it adopted centralized training and was not concerned with privacy-preserving learning, thus its suitability to large-scale vehicular IoT is limited. In comparison, the proposed PCC-CNN IDS under federated learning consistently outperformed both studies. On NSL-KDD, our framework achieved 99.2% accuracy compared to 94% in Study 1, while on Car-Hacking, it reached 99.7% accuracy with an exceptionally low loss of 0.005, surpassing both Study 1 (99%, loss 0.009) and Study 2 (89–99%). The integration of PCC-based feature selection significantly improved the discriminative power of the CNN classifier, leading to more stable convergence and higher detection rates across rare attack categories such as U2R, R2L, and vehicular spoofing attacks. Furthermore, in contrast to Study 2's centralized setting, our federated architecture guarantees privacy preservation as raw vehicle data does not happen to leave client devices. Moreover, integration on real-world edge hardware (Jetson Xavier + Raspberry Pi 4 testbed) demonstrates the scalability and real-time applicability advantage of our model, which provides a more feasible defence solution for transportation IoT systems and CAVs.

In summary, the presented FL-IDS framework achieves state-of-the-art performance with 99.7% accuracy outperforming previous federated and centralized IDS approaches and shows a strong, scalable and privacy-preserving methodology to tackle vehicular cyber security.

4.9 Discussion

The proposed framework clearly demonstrates that federated learning can provide high detection accuracy and strong privacy preservation in vehicular and IoT networks. When tested on Raspberry Pi clients with a Jetson Xavier server, the system achieved a maximum accuracy of 99.7%, confirming its scalability and real-world practicality. The results showed that LR-IDS is a lightweight algorithm with fast convergence, but it struggles with minority and more complex abnormal attack patterns. In contrast, the PCC-CNN model - enhanced through correlation-based feature selection - consistently outperformed LR-IDS across all attack categories and converged to a stable peak performance within only three to five rounds. This balance between accuracy and efficiency highlights its suitability for resource-constrained environments [30]. The proposed framework also outperformed prior centralized and federated IDS models by integrating optimized feature selection with federated aggregation, while maintaining centralized-level accuracy and ensuring that raw vehicular data never left client devices [31], [32]. Nevertheless, the experimentation is limited to NSL-KDD and Car-Hacking datasets; thus, broader validation on diverse datasets is recommended. Future work should also explore energy efficiency and model scalability on lower-end devices. Taken together, the findings confirm federated deep learning as a promising direction for practical, privacy-preserving intrusion detection in CAVs. Future studies may also incorporate cross-domain transfer learning to enhance robustness against unseen attack surfaces.

Table 10: Comparative results with related studies.

Study	Dataset(s)	Approach	Accuracy (%)	Loss
Study 1[28]	NSL-KDD, Car-Hacking	FL + LR, CNN	94.0 / 99.0	0.28 / 0.009
Study 2 [29]	Car-Hacking, CAN-Intrusion, Bus-CAN-Attack	CNN, LSTM	89.0–99.0	–
Proposed Model	NSL-KDD, Car-Hacking	FL + PCC-CNN IDS	99.2 / 99.7	0.12 / 0.005

5 CONCLUSIONS

This study introduced the Federated Learning-Driven Intrusion Detection System (FL-IDS), a practical framework aimed at improving the security of IoT and vehicular networks through decentralized intelligence. Rather than depending on centralized data aggregation, the proposed system enables devices to learn collaboratively while maintaining the confidentiality of their local traffic. This approach reflects a growing need for security methods that respect privacy without compromising detection capability. The findings of this research demonstrate that combining optimized feature learning with federated training can yield a reliable and adaptable intrusion detection solution suitable for real-world environments. The successful deployment on heterogeneous edge hardware underscores the feasibility of applying FL-IDS in operational settings, where computational resources and data sensitivity often pose significant challenges. Beyond its technical performance, this work highlights an important shift in how future CAV and IoT ecosystems may handle cybersecurity. As connected systems continue to expand, traditional centralized solutions will face increasing limitations in scalability, latency, and privacy. FL-IDS shows that distributed, privacy-aware learning can serve as a viable foundation for more resilient and autonomous security architectures - systems capable of evolving alongside the dynamic threats they are designed to detect.

Looking ahead, extending the framework to a broader range of datasets, incorporating resource-aware optimization techniques, and exploring adaptive aggregation strategies will make the system more suitable for large-scale and constrained deployments. Additionally, integrating explainable AI components will enhance transparency and trust, which are essential for decision-making in safety-critical transportation environments.

REFERENCES

[1] N. A. Hamad, K. A. A. Bakar, F. Qamar, A. M. Jubair, R. R. Mohamed, and M. A. Mohamed, “Systematic analysis of federated learning approaches for intrusion detection in the Internet of Things environment,” IEEE Access, vol. 13, pp. 95410–95444, 2025, doi: 10.1109/ACCESS.2025.3574672.

[2] K. Begum, M. A. I. Mozumder, M.-I. Joo, and H.-C. Kim, “BFLIDS: Blockchain-driven federated learning for intrusion detection in IoMT networks,” Sensors, vol. 24, no. 14, p. 4591, 2024, doi: 10.3390/s24144591.

[3] A. Alruwaili, S. Islam, and I. Gondal, “Fed-DTB: A dynamic trust-based framework for secure and efficient federated learning in IoV networks: Securing V2V/V2I communication,” Journal of Cybersecurity and Privacy, vol. 5, no. 3, p. 48, 2025, doi: 10.3390/jcp5030048.

[4] M. Raza, M. J. Saeed, M. B. Riaz, and M. A. Sattar, “Federated learning for privacy-preserving intrusion detection in software-defined networks,” IEEE Access, vol. 12, pp. 69551–69567, 2024, doi: 10.1109/ACCESS.2024.3395997.

[5] L. Lazaros, D. E. Koumadorakis, A. G. Vrahatis, and S. Kotsiantis, “Federated learning: Navigating the landscape of collaborative intelligence,” Electronics, vol. 13, no. 23, p. 4744, 2024, doi: 10.3390/electronics13234744.

[6] S. Alsudani, H. Nasrawi, M. Shattawi, and A. Ghazikhani, “Enhancing spam detection: A crow-optimized FFNN with LSTM for email security,” WJCMS, vol. 3, no. 1, pp. 28–39, Mar. 2024, doi: 10.31185/wjcms.199.

[7] E. Dritsas and M. Trigka, “Federated learning for IoT: A survey of techniques, challenges, and applications,” Journal of Sensor and Actuator Networks, vol. 14, no. 1, p. 9, 2025, doi: 10.3390/jsan14010009.

[8] M. J. J. Rakkini, R. Mohanram, G. Dheepak, S. Subha, R. Hemalatha, and M. R. Suresh, “Transformer-based intrusion detection systems: A deep federated learning approach for privacy-preserving cybersecurity,” in Proc. 6th Int. Conf. Data Intelligence and Cognitive Informatics (ICDICI), Tirunelveli, India, 2025, pp. 216–223, doi: 10.1109/ICDICI66477.2025.

[9] S. W. A. Alsudani and G. K. Saud, “Recurrent neural network optimized by grasshopper for accurate audio data-based diagnosis of Parkinson’s disease,” WJPS, vol. 4, no. 2, pp. 56–75, Jun. 2025, doi: 10.31185/wjps.766.

[10] T. Al-Shurbaji, et al., “Deep learning-based intrusion detection system for detecting IoT botnet attacks: A review,” IEEE Access, vol. 13, pp. 11792–11822, 2025, doi: 10.1109/ACCESS.2025.3526711.

[11] N. A. Al-Khulaidi, A. T. Zahary, A. A. Al-Shargabi, and M. A. S. Hazaa, “Machine learning for intrusion detection in vehicular ad-hoc networks (VANETs): A survey,” in Proc. 4th Int. Conf. Emerging Smart Technologies and Applications (eSmarTA), Sana’a, Yemen, 2024, pp. 1–10, doi: 10.1109/eSmarTA62850.2024.10639016.

[12] N. Manogaran, Y. B. Shankar, M. Nandagopal, H.-K. Su, W.-K. Kuo, S. Ravichandran, and K. Seerangan, “Federated learning and EEL-Levy optimization in CPS ShieldNet fusion: A new paradigm for cyber–physical security,” Sensors, vol. 25, no. 12, p. 3617, 2025, doi: 10.3390/s25123617.

[13] T. Basri, “Securing VANETs for Internet of Things (IoT): AI-driven solutions for privacy and intrusion detection,” in AI-Driven Transportation Systems: Real-Time Applications and Related Technologies, H. Maryam, M. M. Malik, I. U. Khan, and S. K. Gupta, Eds. Cham, Switzerland: Springer, 2025, vol. 62, pp. 113–132, doi: 10.1007/978-3-031-98349-8_6.

[14] S. T. Banafshehvaragh, M. Zarei, and A. M. Rahmani, “A reliable score-based routing protocol using a fog-assisted intrusion detection system in vehicular ad-hoc networks,” Scientific Reports, vol. 15, p. 25709, 2025, doi: 10.1038/s41598-025-08228-3.

[15] B. Buyuktanir, Ş. Altinkaya, G. Karatas Baydogmus, et al., “Federated learning in intrusion detection: Advancements, applications, and future directions,” Cluster Computing, vol. 28, p. 473, 2025, doi: 10.1007/s10586-025-05325-w.

[16] A. Bhardwaj and S. Singh, “Machine learning-driven task offloading for smart vehicular edge computing: Taxonomy, issues, and opportunities,” in Convergence of AI, Federated Learning, and Blockchain for Sustainable Development, M. Kumar, A. Nayyar, A. K. Singh, and Y. Guo, Eds. Cham, Switzerland: Springer, 2025, pp. 145–164, doi: 10.1007/978-3-031-80949-1_9.

[17] M. Alharthi, F. Medjek, and D. Djenouri, “Ensemble learning approaches for multi-class intrusion detection systems for the Internet of Vehicles (IoV): A comprehensive survey,” Future Internet, vol. 17, no. 7, p. 317, 2025, doi: 10.3390/fi17070317.

[18] M. Shahin, A. Hosseinzadeh, and F. F. Chen, “A two-stage hybrid federated learning framework for privacy-preserving IoT anomaly detection and classification,” IoT, vol. 6, no. 3, p. 48, 2025, doi: 10.3390/iot6030048.

[19] R. Abreu, E. Simão, C. Serôdio, F. Branco, and A. Valente, “Enhancing IoT security in vehicles: A comprehensive review of AI-driven solutions for cyber-threat detection,” AI, vol. 5, no. 4, pp. 2279–2299, 2024, doi: 10.3390/ai5040112.

[20] H. G. A. Umar, I. Yasmeen, M. Aoun, et al., “Energy-efficient deep learning-based intrusion detection system for edge computing: A novel DNN-KDQ model,” Journal of Cloud Computing, vol. 14, p. 32, 2025, doi: 10.1186/s13677-025-00762-9.

[21] A. Khraisat, M. A. Talukder, M. A. Uddin, et al., “RF-FedAvg: Federated learning-based random forest model for intrusion detection in wireless sensor networks,” Cluster Computing, vol. 28, p. 873, 2025, doi: 10.1007/s10586-025-05591-8.

[22] R. Islam, S. Mazumdar, and R. Islam, “An experiment on feature selection using logistic regression,” in Proc. 5th Information Communication Technologies Conf. (ICTC), Nanjing, China, 2024, pp. 319–324, doi: 10.1109/ICTC61510.2024.10602330.

[23] Q. Zeng, S. Olatunde-Salawu, and F. Nait-Abdesselam, “FGA-IDS: A federated learning and GAN-augmented intrusion detection system for UAV networks,” in Proc. IEEE 10th Int. Conf. Collaboration and Internet Computing (CIC), Washington, DC, USA, 2024, pp. 50–59, doi: 10.1109/CIC62241.2024.00017.

[24] S. W. Nourildean, W. Mefteh, and A. M. Frihida, “Hybrid deep learning model-based intrusion detection system to improve artificial Internet of Things against cyber attacks,” in Advanced Information Networking and Applications (AINA 2025), Lecture Notes on Data Engineering and Communications Technologies, vol. 252, L. Barolli, Ed. Cham, Switzerland: Springer, 2025, pp. 1–10, doi: 10.1007/978-3-031-87784-1_25.

[25] C. Thana-Aksaneekorn, S. Kosolsombat, and T. Luangwiriya, “Machine learning classification for intrusion detection systems using the NSL-KDD dataset,” in Proc. IEEE Int. Conf. Cybernetics and Innovations (ICCI), Chonburi, Thailand, 2024, pp. 1–6, doi: 10.1109/ICCI60780.2024.10532265.

[26] S. Alshathri, A. Sayed, and E. E.-D. Hemdan, “An intelligent attack detection framework for the Internet of autonomous vehicles with imbalanced car hacking data,” World Electric Vehicle Journal, vol. 15, no. 8, p. 356, 2024, doi: 10.3390/wevj15080356.

[27] O. Arreche, T. R. Guntur, J. W. Roberts, and M. Abdallah, “E-XAI: Evaluating black-box explainable AI frameworks for network intrusion detection,” IEEE Access, vol. 12, pp. 23954–23988, 2024, doi: 10.1109/ACCESS.2024.3365140.

[28] M. H. Bhavsar, Y. B. Bekele, K. Roy, J. C. Kelly, and D. Limbrick, “FL-IDS: Federated learning-based intrusion detection system using edge devices for transportation IoT,” IEEE Access, vol. 12, pp. 52215–52226, 2024, doi: 10.1109/ACCESS.2024.3386631.

[29] S. B. H. Samir, M. Raissa, H. Touati, et al., “Machine learning-based intrusion detection for securing in-vehicle CAN bus communication,” SN Computer Science, vol. 5, no. 1, p. 1082, 2024, doi: 10.1007/s42979-024-03465-1.

[30] M. M. El-Gayar, F. A. F. Alrslani, and S. El-Sappagh, “Smart collaborative intrusion detection system for securing vehicular networks using ensemble machine learning model,” Information, vol. 15, no. 10, p. 583, 2024, doi: 10.3390/info15100583.

[31] C. Xu, F. Zhang, Z. Yang, et al., “A few-shot network intrusion detection method based on mutual centralized learning,” Scientific Reports, vol. 15, p. 9848, 2025, doi: 10.1038/s41598-025-93185-0.

[32] O. Ceviz, P. Sadioglu, S. Sen, and V. G. Vassilakis, “A novel federated learning-based IDS for enhancing UAVs privacy and security,” Internet of Things, vol. 31, p. 101592, 2025, doi: 10.1016/j.iot.2025.101592.