Journal port Science Research
https://jpsr.uruk.edu.iq/index.php/jport
Volume 9, issue 2. 2026

GRU-based Federated Learning for Privacy-Preserving Intrusion Detection in SDN-Enabled IoT Networks

Zaid Ali Alsarray [a]

Department of Intelligent Medical System, University of Information Technology and Communication, Baghdad, Iraq.

ARTICLEI NFO

Keywords:

Federated Learning; Intrusion Detection System; Software-Defined Networking; Gated Recurrent Unit; IoT Security

ABSTRACT

Conventional security measures have become ineffective against advanced cyber threats due substantial attack surface resulting from the quick growth of Internet of Things (IoT) devices. Software-Defined Networking (SDN) allows centralized control, yet it also engenders privacy issues and complicates scalability. This paper proposes a privacy-preserving Intrusion Detection System (IDS) by integrating Software-Defined Networking (SDN) with a Gated Recurrent Unit (GRU)-based Federated Learning (FL) framework. The proposed SDN-FL-GRU model is different from centralized approaches because it lets distributed IoT nodes work together to train a global intrusion detection model without sharing raw data. It does this by using the FedAvg algorithm to combine the data in an efficient way. Testing the proposed framework on the CICIDS2017dataset shows that it can accurately detect complex attacks while keeping data private, with a detection accuracy of 93.4% and an F1-score of 92.8%. The results obtained indicate that the GRU-based approach outperforms traditional distributed models regarding convergence speed and effectiveness in resource-constrained edge environments.

1. INTRODUCTION

1.1 Overview

The rapid expansion of the Internet of Things (IoT) is transforming the current digital ecosystem, offering a connection framework that encompasses various sectors, including smart cities and industrial automation. Nevertheless, this technological improvement has led to a rise in sophisticated cyber breaches, with predictions indicating that by 2030,the number of connected devices will reach billions (Gubbi et al., n.d.; Waheed et al., 2024). Creates a massive attack surface that traditional security measures can no longer adequately protect (S, 2025) . To manage this growing complexity, Software Defined Networking (SDN) introduced as a flexible architectural answer that separates control logic from data forwarding. Federated Learning (FL) has emerged as a decentralized approach that addresses these privacy and architectural bottlenecks, allowing user to collectively train a model by keeping their private data locally and only sharing intermediate outputs with a central node (Kreutz et al., n.d.; Raza et al., 2024). Federated Learning mitigates the privacy issue in distributed IoT networks by retaining data inside the local domain while also decreasing the communication burden of traditional SDN-based security methods (Chetouane & Karoui, 2025; Mcmahan & Ramage, 2017).

1.2 Problem Statement

Despite the control efficiency, the inherent centralization of SDN controllers introduces a widespread Single point of failure (SPOF) and severe security bottlenecks, because the need to accumulate raw traffic data for intrusion detection often exposes sensitive user data to potential breaches (Vyas et al., 2024). Nevertheless, while Federated Learning protects privacy, it lacks the comprehensive perspective needed to identify complex threats. Leveraging this unique combination, recent studies have begun to explore the integration of federated learning within the software-defined networking framework to develop more secure and privacy-preserving intrusion detection systems. However, in these hybrid environments, achieving high detection accuracy remains challenging because of the non-Independent and Identically Distributed (Non-IID) characteristics of data collected by IoT devices and the need for improved aggregation procedures (Khan et al., n.d.; Waheed et al., 2024). Although the decentralized method has enormous potential, it's clear that, as

E-mail address:
zaid.ali-bmic@uoitc.edu.iq [a]

Received 29th January 2026، Accepted 25th March 2026

10.36371/port.2026.2.12

we have seen in 2024,2025 studies, the data privacy issue remains challenging. In order to overcome these limitations, Khan et al. introduced a federated learning framework for an SDN-enabled IoT environment, demonstrating that centralized orchestration enhances the security of distributed learning (Khan et al., 2021). yet despite these enhancement the interaction between the controller and various IoT nodes generally result in much higher communication costs as revealed by Harchi et al (Harchi, 2025). Furthermore, the adaptation of these structures to evolving attack vectors remains a significant area for development. Also, Alkhamisi et al. (Networks et al., 2024). Emphasized that in complex multi-controller SDN systems, conventional detection algorithms inadequately scale, needing a robust federated learning approach to ensure synchronized security policies across the network. As Javeed et al. (2024) (Javeed et al., 2024) argued, ensuring integrity at the edge needs more than decentralized training; it demands a robust modification of the learning process. The strategic analysis by Tom et al. (Tom et al., 2025). Confirms that refining the weight aggregation process is the most effective factor in mitigating real-time threats in industrial IoT

1.3 Research Objective

The principal objective of this paper is to enhance a light-weight and privacy –preserving intrusion detection system for IoT networks. The specific objectives are to:

1. Develop A hybrid Architecture: combines software-defined networking with federated learning to achieve a balance between centralized network visibility and decentralized data privacy.
2. Build a lightweight model: build an engine based on the Gated Recurrent Unit (GRU) to efficiently detect saturation attacks without using additional resources
3. Assessment of Results: Evaluate the overall performance of the framework using the CICIDS2017 dataset and compare it with the performance of current centralized and decentralized techniques.

1.4 Contribution

This study advances IoT security by proposing a novel hybrid architecture that combines Software-Defined Networking with Federated Learning to guarantee data privacy while maintaining global governance. Furthermore, it develops a lightweight gated recurrent unit detection engine tailored for resource-limited devices. The system has been experimentally evaluated with the CICIDS2017 dataset, demonstrating superior detection accuracy and reduced latency compared to current centralized and decentralized solutions.

1.5. Paper Organization

The remaining parts of this work are structured as follows: Chapter 2 covers the related literature on SDN security and federated learning. Chapter 3 discusses the methodology and the suggested hybrid framework architecture. Chapter 4 will present the experimental setup, dataset description, and a comparative analysis of the results. At last, the Paper is concluded in Chapter 5, which also offers ideas for future research.

Chapter 2: Literature Review

2.1. Overview of the IoT Security Landscape

The enormous volume and variety of the IoT ecosystem make traditional security measures ineffective due to significant resource limitations (Sharma, 2025) . Since vulnerabilities threaten the availability of critical infrastructure beyond data theft (Pastorek et al., 2025). A systematic classification of threats is necessary. Figure 2.1 illustrates the taxonomy of IoT attacks across the physical, network, and application layers, Humayun et al. (Humayun et al., 2024). Identifies network layer anomalies, particularly DoS and DDoS, as the principal disruption in SDN environments, thus defining the scope of this research.

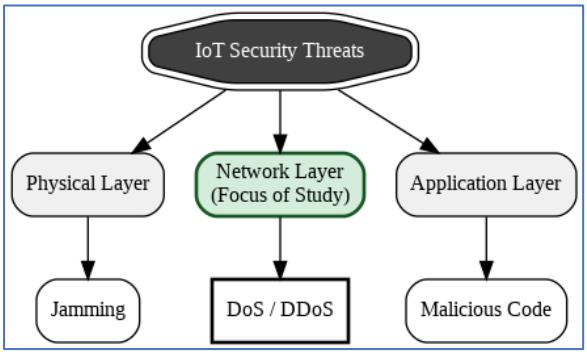

***Figure 1**: Taxonomy of IoT Security Threats (Physical, Network, and Application Layers)*

2.2. Evolution of IDS: From Centralized to Distributed Intelligence

Conventional signature-based detection techniques are becoming increasingly useless against zero-day attacks in a dynamic IoT ecosystem, as they fail to detect unfamiliar attack patterns. (Zhang et al., 2025). Although centralized machine learning enhances anomaly detection, it faces considerable latency and privacy concerns due to the need to transmit critical raw data to cloud servers (Hamad et al., 2025). Addressing those boundaries, Reis demonstrates that distributed “Edge AI” designs significantly reduce computational overhead while ensuring data privacy, thus encouraging a shift towards a federated protection framework (Reis, 2025).

2.3. Security Challenges in SDN-enabled IoT

Software-Defined Networking transforms IoT management by separating the control plane from the data plane, resulting in centralized programmability and dynamic scalability (Ahmadvand et al., 2023). The architectural centralization creates a significant risk, as the controller acts as a Single Point of Failure (SPOF). The authors claim that adversaries exploit this bottleneck by executing packet-saturation attacks (some form of DDoS), wherein the controller is overwhelmed with different massive flow requests. A lack of resources debilitates the entire network, requiring strong, distributed defense mechanisms to ensure the control plan availability(Andishmand et al., n.d.).

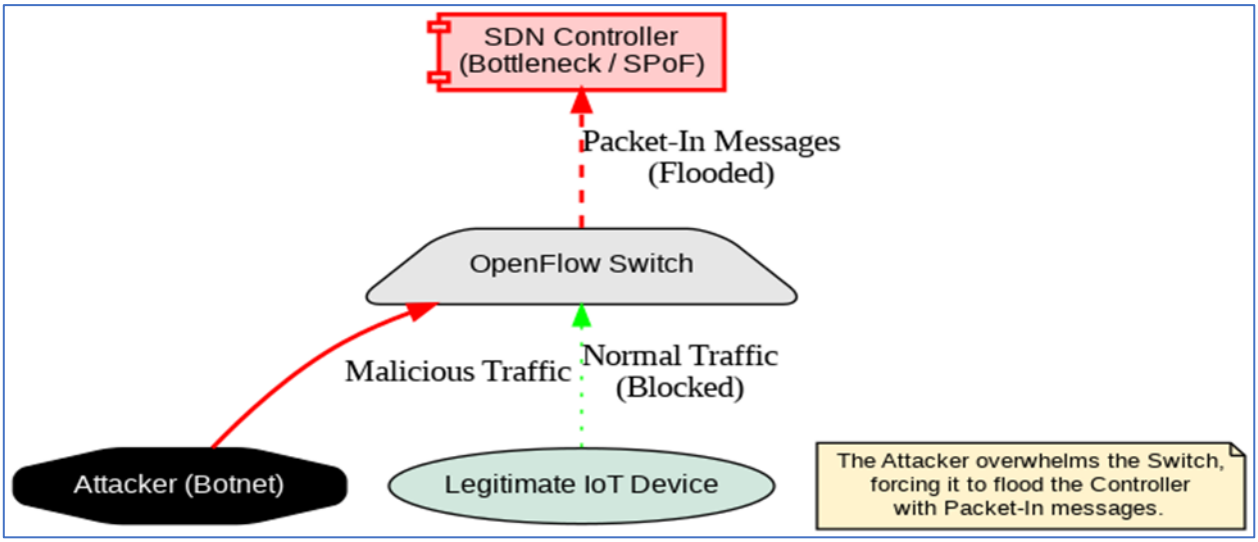

***Figure 2**: Representation of DDoS Saturation Attack targeting the SDN Controller.*

2.4. Federated Learning: Principles and Advantages

To mitigate the risk of privacy breaches and delays in response times linked to to centralized system; federated learning has evolved as a decentralized paradigm transformation. According to Adam and Baroudi FL facilities collaborative model trains among distributed IoT nodes without any transmission of raw data. Rather than sending sensitive traffic logs to a central server, devices conduct local training and share only model updates (gradients) to a global aggregator (Adam, 2024). Moreover, Vyas and Kim show that the 'Model-to-Data' approach not just maintains data control but also significantly reduces the usage of network bandwidth, making it an ideal safety precaution for an extensive, resource-limited IoT ecosystem (Kim & Lin, 2024; Vyas et al., 2024).

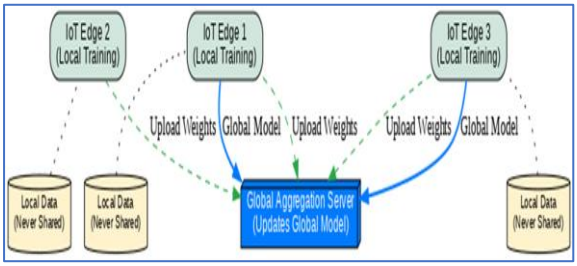

***Figure 3**: FL Architecture: Local training vs Global Aggregation*

2.5. Critical Analysis of Related Work

The most recent research on IoT intrusion detection can be divided into three main groups: centralized SDN-based solutions, decentralized FL models, and hybrid architectures. SDN methods provide you with a global view, but they also have a single point of failure and privacy issues with sending the raw data. Despite that, the FL approach offers better privacy protection, yet it cannot detect more complex threats to global network information effectively. In recent times, there have been hybrid methods created that try to address this void, but they often present extremely high levels of computational complexity, which limit their usability and feasibility on edge devices with constrained available resources. Table 1 provides a comprehensive breakdown of these advanced methods in comparison with the proposed framework.

***Table 1:** Comparison of State-of-the-Art IoT Security Frameworks.*

Reference	Technique	Key Strengths	Limitations (Research Gap)
Zhang et al. (2025) [20]	Signature-based IDS	Low overhead; Simple deployment.	Fails against Zero-day attacks.
Ji et al. (2025) [24]	Centralized SDN + DL	High accuracy; Global visibility.	Single Point of Failure (SPoF); Privacy risks.

Vyas et al. (2024) [9]	Standard Federated Learning	Data privacy; Bandwidth efficiency.	Lacks network mitigation; Poisoning risks.
Reis (2025) [22]	Distributed Edge AI	Low latency; Reduced server load.	High complexity for edge nodes; No global control.
Ali et al. (2024) [27]	Blockchain + FL	Immutable logs; Decentralized trust.	High latency due to consensus; Energy-intensive.
Afifi et al. (2025) [28]	Transformer Models (DL)	Superior traffic classification.	Computationally expensive; Slow inference time.
Proposed Framework	SDN + FL + GRU	Privacy; Robust detection; Efficient.	(Addressed in this paper)

2.6. Research Gap and Motivation

A comparison of Table 1 presents the primary difference between existing IoT models. Centralized software-defined network (SDN) (Ji et al. 2025) is capable of giving total visibility of the data traffic to enhance its management. On the other hand, it leads to central vulnerabilities and threats to data privacy. Decentralized and standardized federated learning methods, in contrast, are very effective in protecting data privacy but still cannot provide the central control that is required to prevent attacks in real time (Kim & Lin, 2024; Raza et al., 2024). Besides the need for an up-to-date and accurate framework (Ali et al. 2025), there is a need to improve schemes based on blockchain technology. Additionally, Firdaus et al. proposed advanced deep learning techniques based on transformers, which demand much computation in the presence of resource-constrained IoT environments, where responses with low latency are necessary. The solutions are usually costly and not feasible (Firdaus et al. 2025). It leads to a significant research gap in deciding on a method that combines risk reduction side by side with computational efficiency and privacy. The present work introduces a basic combined method (SDN_FL_GRU) that identifies and fixes the saturation risks without slowing down the network.

Chapter 3: Proposed Methodology

3.1. Overview of the Framework

This paper designs a strong intrusion detection mechanism focusing on privacy for IoT scenarios. Combining the features of software-defined networking and federated learning, the proposed solution enables detecting attacks in the spread networks. Additionally, this technique gets rid of privacy issues that come from sending sensitive raw data to a central server.

3.2. System Architecture Design

3.3. Data Pre-processing and Scaling

3.3.1. Dataset Selection

The CICIDS2017 dataset was used to evaluate the performance of the proposed Federated Learning (FL)-based Software Defined Networking (SDN)-based intrusion detection system. The dataset includes native and categorized network traffic encompassing all attack methods, such as DoS and DDoS. Therefore, the dataset effectively simulates future attacks on devices and services through the Internet of Things (IoT). This study used a mixed dataset consisting of 25,000 records, of which 12,500 were legitimate records, and 12,500 were attack records. The data were distributed across all study nodes as shown in Table 2. This type of partitioning is important to mitigate the effects of class imbalance, since it allows each client to receive both legitimate and attack examples during local updates.

Table 2. *Statistical distribution of CICIDS2017 samples across federated learning nodes.*

Node	Benign Samples	Malicious Samples	Total Samples
Node 1	2,500	2,500	5,000
Node 2	2,500	2,500	5,000
Node 3	2,500	2,500	5,000
Node 4	2,500	2,500	5,000
Node 5	2,500	2,500	5,000
Total	**12,500**	**12,500**	**25,000**

3.3.2. Data Pre-processing

In order to enhance the CICIDS2017 dataset in a resource-constrained federated environment, Pearson's correlation analysis is used to reduce the number of dimensions or features at the beginning of the process while also providing computational efficiency. As presented in Table 3, there are ten flow-level attributes determined through correlation analysis that are the most discriminative and ultimately retained. Figure 4 contains a correlation heatmap that shows that these ten discriminative features are compact, have high predictive quality, and contain no redundancy, as verified by comparing them to other features in the correlation heatmap. The preprocessing stage resulted in having only ten features, as initially their goal was not just to reduce the number of dimensions but also to create a compact detection model for use in federated IoT nodes. The correlation heatmap provides further evidence of the selected attributes being highly discriminating and having a very low level of redundancy.

Table 3: *Flow-level features based on Pearson Correlation Analysis.*

Index	Feature Name	Description / Relevance
1	Flow Duration	Total duration of the flow in microseconds.
2	Protocol	Protocol used in the flow (e.g., TCP, UDP).
3	Fwd Packets Total	Total packets sent in the forward direction.
4	Total Bwd Packets	Total packets sent in the backward direction.
5	Total Length of Fwd Packets	Total size of payload bytes in forward packets.
6	Total Length of Bwd Packets	Total size of payload bytes in backward packets.
7	Flow Bytes/s	Number of bytes transmitted per second.
8	Flow Packets/s	Number of packets transmitted per second.
9	Average Packet Size	Mean size of packets in the flow.
10	Packet Length Variance	Variance in packet length (indicates jitter).

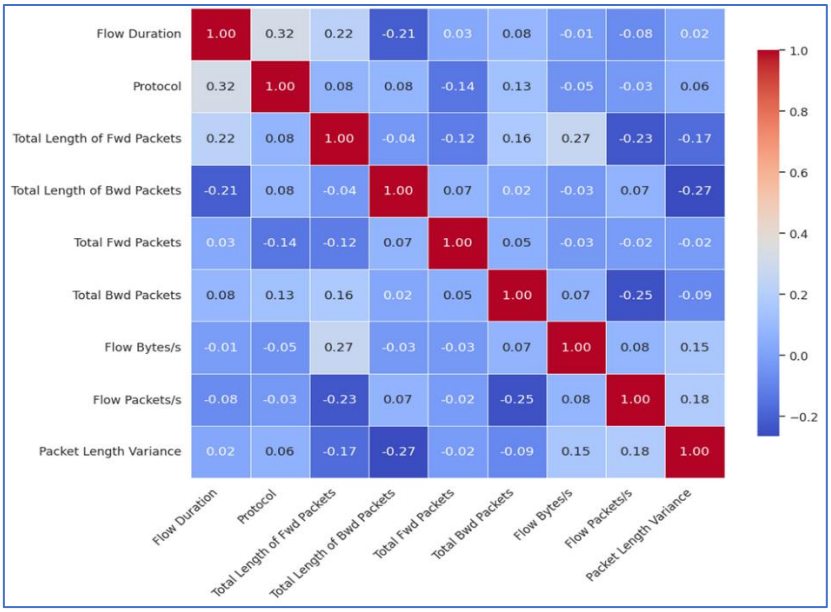

Figure 4: *Pearson correlation heatmap of the selected feature subset.*

3.3.3. Local GRU Model Design

To capture the time dependencies, present in network traffic while minimizing computational overhead on IoT edge devices, the proposed framework utilizes a Gated Recurrent Unit (GRU) model. Compared to Long Short-Term Memory (LSTM) networks, it requires less complexity in parameters and facilitates faster convergence because it has only two gates: Update and Reset gates.

Mathematical Formulation

Every time step t The GRU unit computes the hidden state h_t using the following operations:

1. Reset Gate (r_t) Controls the retention of past information.

$$r_t = \sigma(W_r . [h_{t-1}, x_t])$$

2. Update Gate (z_t): Set the information flow to the present condition.

$$z_t = \sigma(W_z . cdot [h_{t-1}, x_t])$$

3. Candidate State $(\tilde{h}_t)$:

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \odot h_{t-1}, x_t])$$

4. Final Hidden State (h_t): $h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$

Where σ defines the sigmoid function, $\odot$ signifies element-wise multiplication, and W denotes weight matrices

.

Table 4: *Architecture and hyperparameters of the GRU mode*

Configuration / Value	Parameter / Layer
Feature Dimension Input	64(Pre-processed) Features
Recurrent Layer Type	Gated Recurrent Unit (GRU)
Number of Units	Hidden Units 64
Recurrent Activation	Sigmoid(Gate),Tanh (State)
Regularization	Dropout (Rate = 0.2)
Fully Connected Layer	Dense (32 Neurons, ReLU)
Output Layer	Neurons(SoftmaxActivation)2
Optimizer Algorithm	(α = 0.001) Adam
Loss Function	Sparse Categorical Cross-Entropy
Batch Size	Samples 64
Training Epochs	with Early Stopping 50

3.4. Operational Logic and SDN-FL Protocol

The Federated Averaging (FedAvg) method governs the core logic of the proposed system. The SDN controller combines local updates using the following weighted average:

$$W_{t+1} = \sum_{k=1}^{K} \frac{n_k}{n} w_t^k$$

We designed the SDN-FL framework based on repeated communication cycles. Each round starts with the SDN controller sending the recent global model to the relevant IoT nodes. This forced synchronization keeps all distributed learners on the same baseline, stopping model divergence before it becomes an issue. The controller also makes sure to time these updates just right. This helps reduce network congestion, which is a common problem in IoT setups with many devices

The detailed operational flow of the SDN-FL Protocol is presented in Algorithm 1.

Algorithm 1: SDN-based Federated Learning Protocol for IDS

Inputs: Initial global model weights W_0, Number of IoT nodes K, Learning rate ηNumber of communication rounds T.

Output: Optimized Global Model W_{Global}

1. Initialization:
- SDN Controller initializes global weights W_0
- Distribute $W_0$to all participating IoT nodes.

2. Federated Learning Process:
- For each communication round t = 1, 2,… , T do:
  - Step A: Node Selection & Distribution
    - Controller selects K active IoT nodes.
    - Selected nodes download current global weights W_0
  - Step B: Local Training (at each IoT Node k in parallel)
    - Compute local gradient: $\nabla L\ w_k$
    - Update local weights: $w_k^{t+1} \leftarrow w_k^t - \eta \nabla L(w_k)$
  - Step C: Weight Uploading
    - Nodes transmit only updated parameters $\{w_k^{t+1}\}$ to the SDN Controller.
    - Step D: SDN-Based Global Aggregation

a. The SDN controller collects local updates w_k^{t+1} from all participating nodes.

b. The controller computes the new global weight $W_{\{t+1\}}$ using the

Federated Averaging (FedAvg) rule:

$$W_{t+1} \leftarrow \sum_{k=1}^{K} \left(\frac{n_k}{N}\right) w_k^{t+1} \quad \ldots \quad (1)$$

The global model parameters are updated using the standard Federated Averaging (FedAvg) rule, where the contribution of each IoT node is weighted by the size of its local dataset:

$$w_{global} = \sum_{k=1}^{K} \left(\frac{n_k}{N}\right) k_w \ \ldots (2)$$

c. The updated global parameters are re-broadcast to the perception layer.

3. Finally, the optimized global model is deployed at the SDN controller for real-time traffic monitoring and anomaly detection.

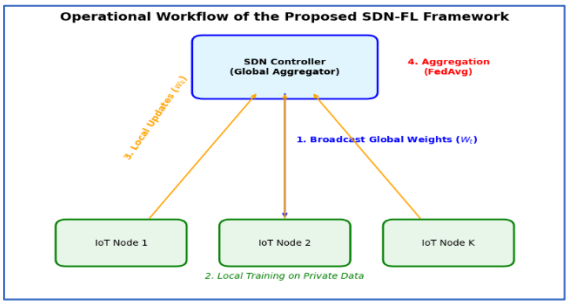

Figure 5: *Systematic communication workflow and iterative weight aggregation in the SDN-FL framework*

.

Figure 5 illustrates the SDN aggregator and local IoT nodes' interaction, enabling iterative parameter updates that preserve privacy and network efficiency by concealing raw traffic data.

4: Results and Discussion

4.1. Experimental Setup

The proposed SDN-FL-GRU framework executed on the cloud-based testbed, specifically Google Colab Pro+, using an

NVIDIA Tesla T4 GPU (16 GB VRAM). Python 3.12 was used with the support of PyTorch to create these models, and the Flower framework managed the federated learning rounds. The Ray Engine instantiated five independent IoT nodes with strictly partitioned datasets to simulate a realistic edge-computing environment that could ensure data isolation.

Table 5: *Global model performance metrics across federated communication rounds*

Round	Accuracy	Precision	Recall	F1-Score	Loss
1	62.50	61.10	59.40	58.20	0.6842
2	76.50	75.80	79.20	78.20	0.4120
3	84.10	83.50	86.40	85.15	0.2850
4	90.50	89.90	92.10	91.80	0.1980
5	**92.80**	**92.50**	**94.10**	**93.40**	**0.1450**

Rapid convergence of the model is evident from the large decrease in loss value from 0.68 to 0.14, as shown in Table 5. Improvements in both Precision and Recall simultaneously indicate that the use of a GRU produces an accurate model that successfully balances high detection rates and low false positive rates for a variety of complex threats in the Internet of Things (IoT).

4.2. Performance Evaluation and Analysis

4.2.1. Accuracy Analysis

The detection capabilities of the proposed SDN-FL framework were assessed by monitoring the global model correctness across five communication rounds. Figure 6 depicts the evolution of global accuracy as the Federated Learning process advances. The graph illustrates that the model achieves an optimal accuracy of 93.4% by the last round. The initial round 1 accuracy of 62.5% indicates inadequate prior exposure of the global model to the characteristics of decentralized traffic. The subsequent increase in round 2 to (78.8%) demonstrates the efficacy of the FedAvg algorithm in effectively integrating diverse local insights into a unified global representation.

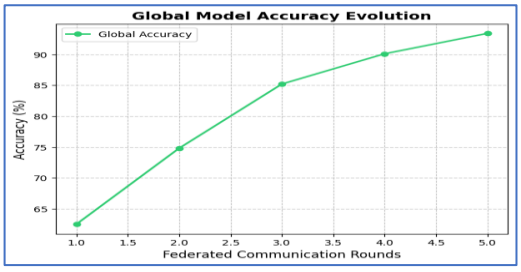

Figure 6: *Evolution of global model accuracy throughout FL*

By the last round, the model converged to a peak accuracy 93.4%

4.2.2. Convergence Analysis

The learning process stability is demonstrated by the reduction in loss across rounds. The Binary Cross-entropy (BCE) loss was computed at the SDN controller after each aggregation phase.

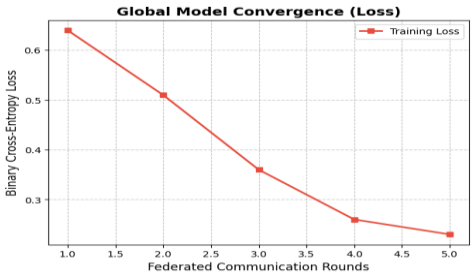

Figure 7: *Convergence Behavior of the BCE Loss Function*

The steady downward trend of the loss curve shows the robustness of FedAvg weight aggregation.

4.2.3. Classification Performance

A confusion matrix was created to evaluate the model's accuracy in its ability to separate the normal and attack traffic classes during the final evaluation stage beside overall accuracy.

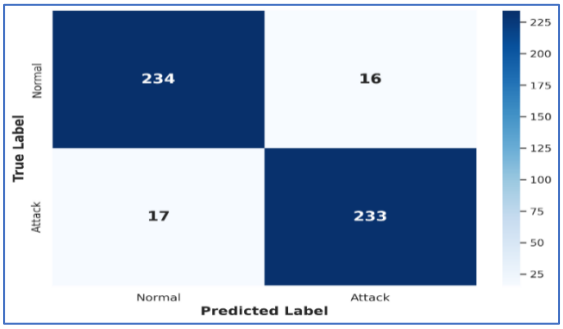

Figure 8. *Confusion matrix depicting intrusion detection accuracy*

The matrix demonstrates the model's exceptional accuracy in distinguishing between normal traffic and malicious attacks.

4.2.4. Computational Efficiency and Network Overhead

The use of the GRU-based classifier resulted in a 15% decrease in the required amount of time needed for training as compared to LSTM-type classifiers, proving their suitability for edge environments with limited resources.

4.3 Comparison to Current State of the Art

The effectiveness of the proposed SDN-FL-GRU architecture was evaluated against a distributed central detection system. This analysis provided an explicit evaluation of the performance of such systems in terms of the authenticity of identity, privacy, and architectural efficiency.

Table 6: *Comparison of the proposed framework with existing IDS approaches*

Model	Architecture	Privacy	Accuracy	Limitation
Centralized DNN	Centralized	Low	94.10	Single Point of Failure, Privacy risk
Standard FL-ANN	Distributed	High	91.20	Struggles with temporal dependencies
FL-SVM	Distributed	High	86.50	Low accuracy on complex attacks
Proposed FL-GRU	Distributed	High	93.40	alanced Accuracy & Privacy

Table 6 illustrates that the proposed FL-GRU achieves 93.40 accuracy, surpassing distributed baselines, including FL-ANN (91.20%) and FL-SVM (86.50%), while there is a slight performance gap compared to the centralized DNN. The proposed framework offers a critical advantage: it preserves the privacy of data at the edge without significantly compromising detection capabilities.

Chapter 5: Conclusion and Future Work

5.1. Conclusion

A privacy-preserving intrusion detection system utilizing software defined network and a gated recurrent unit has been introduced. The proposed SDN-FL-GRU model addresses the fundamental privacy-utility trade-off in the Internet of Things (IoT) networks through the provision of decentralized training on the basis of the framework, without the need to exchange raw data. The results demonstrate that the proposed framework achieves an overall classification accuracy of 93.4%, while also minimizing the communication overheads associated with the FedAvg aggregation process, thus confirming that low-complexity deep learning models can be successfully executed at the device edge level for protecting IoT networks and their associated critical infrastructure.

5.2. Future Work

Future research will focus on making the model more robust against adversarial poisoning attacks to ensure security in hostile environments. Further, we intend to deploy the framework on physical edge hardware, such as Raspberry Pi, in order to test real-time inference latency and energy consumption in live deployment scenarios.

Conflicts of Interest

The authors should pledge that they don't have any conflict of interest regarding their research. If there are no conflict of interest, then authors can declare the following: "The authors declare no conflicts of interest".

Funding

The funding section of your journal paper template should provide a concise and transparent declaration of the financial support received to carry out the research presented in your paper.

Acknowledgment

The preferred spelling of the word "acknowledgment" in America is without an "e" after the "g". Avoid the stilted expression "one of us (R. B. G.) Thanks ...". Instead, try "R. B. G. thanks..." Put sponsor acknowledgments in the unnumbered footnote on the first page.

References

Adam, M. (2024). Federated Learning for IoT : Applications , Trends , Taxonomy , Challenges , Current Solutions ,. *IEEE Open Journal of the Communications Society*, *5*(October), 7842–7877. https://doi.org/10.1109/OJCOMS.2024.3506214

Ahmadvand, H., Lal, C., Hemmati, H., Sookhak, M., & Member, S. (2023). *Privacy-Preserving and Security in SDN-Based IoT : A Survey*. *11*(May).

Ali, M., Hu, Y., & Li, J. (2025). *Federated Learning Augmented Cybersecurity for SDN-Based Aeronautical Communication Network*.

Andishmand, R., Mohammdi, H., & Mostafavi, S. (2023). *Detection and Analysis of DDoS Attacks in Software-defined Networks*.

Chetouane, A., & Karoui, K. (2025). New Continual Federated Learning System for Intrusion Detection in SDN-Based Edge Computing. *Concurrency and Computation: Practice and Experience*, *37*(2), e8332. https://doi.org/https://doi.org/10.1002/cpe.8332

Firdaus, A., Zaki, F., Hanif, H., Aqil, N., & Badrul, N. (2025). *Transformer-based tokenization for IoT traf fi c classi fi cation across diverse network environments*. 1–31. https://doi.org/10.7717/peerj-cs.3126

Gubbi, J., Buyya, R., & Marusic, S. (2023). *Internet of Things (IoT): A Vision , Architectural Elements , and Future Directions*. *1*, 1–19.

Hamad, N. A., Azmi, K., & Bakar, A. B. U. (2025). Systematic Analysis of Federated Learning Approaches for Intrusion Detection in the Internet of Things Environment. *IEEE Access*, *13*(May), 95410–95444. https://doi.org/10.1109/ACCESS.2025.3574672

Harchi, A. (2025). *SDN-Cloud Incident Detection & Response with Segmented Federated Learning for the IoT*. *5*(2), 671–684.

Humayun, M., Tariq, N., Alfayad, M., Zakwan, M., Alwakid, G., & Assiri, M. (2024). *Securing the Internet of Things in Artificial Intelligence Era : A Comprehensive Survey*. *February*.

Javeed, D., Shahid, M., Adil, M., & Kumar, P. (2024). Ad Hoc Networks A federated learning-based zero trust intrusion detection system for Internet of Things. *Ad Hoc Networks*, *162*(March), 103540. https://doi.org/10.1016/j.adhoc.2024.103540

Ji, Z., Cui, Y., Guo, Y., & Shen, G. (2025). *Towards saturation attack detection in SDN : a multi-edge representation learning-based method*.

Khan, L. U., Saad, W., Han, Z., & Hossain, E. (2021). *Federated Learning for Internet of Things : Recent Advances , Taxonomy , and Open Challenges*. 1–30.

Khan, L. U., Saad, W., Han, Z., Hossain, E., & Hong, C. S. (2021). Federated Learning for Internet of Things: Recent Advances, Taxonomy, and Open Challenges. *IEEE Communications Surveys & Tutorials*, *23*(3), 1759–1799. https://doi.org/10.1109/COMST.2021.3090430

Kim, T., & Lin, J. (2024). *A Survey on Heterogeneity Taxonomy , Security and Privacy Preservation in the Integration of IoT , Wireless Sensor Networks*.

Kreutz, D., Ramos, F. M. V, Verissimo, P., Rothenberg, C. E., Azodolmolky, S., Member, S., & Uhlig, S. (2015). *Software-Defined Networking : A Comprehensive Survey*. 1–49.

Mcmahan, H. B., & Ramage, D. (2017). *Communication-Efficient Learning of Deep Networks from Decentralized Data*. *54*.

Networks, M. S., Alkhamisi, A., & Katib, I. (2024). *Federated Learning-Based Security Attack Detection for*.

Pastorek, A., Pastorek, A., & Tundis, A. (2025). *Navigating the landscape of IoT security and associated risks in critical infrastructures*. *July 2024*. https://doi.org/10.1145/3664476.3669979

Raza, M., Saeed, M. J., & Sattar, M. A. (2024). Federated Learning for Privacy-Preserving Intrusion Detection in Software-Defined Networks. *IEEE Access*, *12*(April), 69551–69567. https://doi.org/10.1109/ACCESS.2024.3395997

Reis, M. J. C. S. (2025). Scalable Intrusion Detection in IoT Networks via Property Testing and Federated Edge AI. *IEEE Access*, *13*(August), 153244–153262. https://doi.org/10.1109/ACCESS.2025.3603937

S, B. B. (2025). *Federated learning in intrusion detection : advancements , applications , and future directions*. https://doi.org/10.1007/s10586-025-05325-w

Sharma, N. (2025). A survey on IoT security : challenges and their solutions using machine learning and blockchain technology. In *Cluster Computing*. Springer US. https://doi.org/10.1007/s10586-025-05208-0

Tom, A. K., Khraisat, A., Jan, T., Nguyen, T. D., & Alazab, A. (2025). *Survey of Federated Learning for Cyber Threat Intelligence in Industrial IoT : Techniques , Applications and Deployment Models*. 1–25.

Vyas, A., Lin, P.-C., Hwang, R.-H., & Tripathi, M. (2024). Privacy-Preserving Federated Learning for Intrusion Detection in IoT Environments: A Survey. *IEEE Access*, *12*, 127018–127050. https://doi.org/10.1109/ACCESS.2024.3454211

Waheed, S., Hanif, S., Hafeez, R., Sharif, M. I., Siddique, K., & Akhtar, Z. (2024). *A Comprehensive Survey on Applications , Challenges , Threats and Solutions in IoT Environment and Architecture*. https://doi.org/10.3844/jcssp.2024.310.332

Zhang, H., Ye, J., Huang, W., Liu, X., & Gu, J. (2025). Survey of federated learning in intrusion detection. *Journal of Parallel and Distributed Computing*, *195*, 104976. https://doi.org/https://doi.org/10.1016/j.jpdc.2024.104976