I. J. Computer Network and Information Security, 2026, 2, 196-209
Published Online on April 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2026.02.11

Addressing Data Privacy Concerns in IoT Architecture with Federated Learning and TinyML

Hiba Kandil*
IPSS, Faculty of Sciences, Mohammed V University in Rabat, Avenue Ibn Battouta B.P. 1014 RP, Rabat, Morocco
E-mail: hiba_kandil@um5.ac.ma
ORCID iD: https://orcid.org/0009-0008-1165-2596
*Corresponding author

Hafssa Benaboud
IPSS, Faculty of Sciences, Mohammed V University in Rabat, Avenue Ibn Battouta B.P. 1014 RP, Rabat, Morocco
E-mail: hafssa.benaboud@fsr.um5.ac.ma
ORCID iD: https://orcid.org/0000-0002-5027-8217

Received: 28 September 2025; Revised: 08 November 2025; Accepted: 27 January 2026; Published: 08 April 2026

Abstract: The rapid extension of the Internet of Things (IoT) has introduced significant concerns, particularly in ensuring data security and safeguarding sensitive and private data. The integration of Federated Learning into IoT architecture has occurred as a covenanting solution to address the risks of data breaches, resource efficiency, and the challenges of data privacy and security. This paper presents a novel lightweight framework tailored for resource-constrained IoT devices that integrates Federated Learning and Tiny Machine Learning (TinyML) to deploy lightweight, reliable models on edge devices. Our experimental results show that the proposed approach can improve efficiency, reduce communication overhead, and enhance privacy preservation.

Index Terms: Internet of Things (IoT), Data Privacy, Machine Learning, Federated Learning, TinyML.

1. Introduction

The rapid evolution of the IoT network has changed the way we interact with smart devices. As billions of connected devices with embedded intelligence exchange and gather huge amounts of data. Data privacy, efficiency and security have become the main concerns of developers, researchers, and organizations [1,2]. IoT architecture is generally made up of interconnected layers with sensors, edge devices, communication networks, cloud services platforms, and data analytics systems. Each layer reveals concerns related to: data privacy, security breaches, and compliance with regulation. This is an expected consequence of the distributed and scalable nature of IoT systems, the heterogeneity of devices, and the huge amounts of generated data, causing identity theft, data breaches, unauthorized surveillance, erosion of user trust, or even damage to the entire system [3,4]. In order to face these challenges, IoT privacy frameworks [5] need to embrace mechanisms such as end-to-end encryption, data minimization, secure data processing techniques, and user consent management. Therefore, it is vital to overcome these issues while enhancing the operational performance of IoT systems. Traditional centralized solutions for data collection and processing in IoT often suffer from data breaches, latency issues, scalability, and data privacy concerns [6,7]. This is where innovative remedies like federated learning and TinyML emerge. Federated learning empowers decentralized machine learning model training across IoT architecture while raw and sensitive data are maintained localized [8,9]. This mechanism preserves data privacy and reduces the risks of unauthorized access, and data breaches [10]. Since with federated learning, IoT devices learn from their localized data and send only model updates to a central server without sharing raw data. Meanwhile, TinyML provides the means for the optimization and deployment of machine learning models on resource-constrained IoT devices. TinyML enables devices with limited computational resources to make real-time decisions while reducing latency, and conserving energy. This technology contributes to setting up intelligence on the edge level [11-13]. The combination of federated learning and TinyML offers the perfect solution to address data privacy and efficiency on IoT architecture, allowing industries to optimize the performance of IoT systems with the respect of ethical considerations [14,15].

This work is open access and licensed under the Creative Commons CC BY 4.0 License.

To fulfill these requirements, this work proposes a lightweight approach for data privacy tailored for resource-constrained IoT devices. The main contributions of this work are listed below.

- Investigating the effectiveness and viability of combining Federated Learning and TinyML for machine learning model deployment and optimization in IoT architecture.
- Proposing a lightweight architecture considers the Network traffic analysis of heterogeneous IoT devices at different communication levels.
- Conducting a comparative analysis of federated learning model performance on various heterogeneous IoT devices with unbalanced datasets.

This paper is organized as follows. Section 2 outlines recent research contributions related to our work. Section 3 provides the proposal methodology, implementation, and experimentation. It is divided into four parts. The first part describes the dataset, the second part presents the network environment, the third part covers the framework architecture, and the last part provides the implementation details. The results and discussion are illustrated in Section 4. Section 5 presents the proposal challenges and the future research directions to overcome these challenges, while Section 6 concludes the paper.

2. Related Works

In this section, we present some recent related contributions to our proposal. Authors of [16] presented a novel approach for implementing federated learning while addressing privacy issues in a tiny IoT environment. The proposed approach focused on the tiniest IoT devices used in federated learning research, such as ESP32, Teensy 4.0, and STM32 Nucleo H7. In addition, they used the transfer learning technique and proved the effectiveness of the proposed solution for learning while preserving privacy in real-world applications. Nonetheless, the proposed solution may not resolve all communication issues in federated learning environment. It does not tackle many details of real-world privacy issues. In [17] the authors presented an innovative model-agnostic meta-learning technique for TinyML devices named: TinyMetaFed. By performing regression, audio, and image classification experiments on Raspberry Pi and Arduino MCU devices. The framework enables privacy-preserving using Top_P% selective communication and partial local reconstruction. Furthermore, the framework grants energy consumption optimization and communication savings. Yet, as the experiments are done in restrained conditions. The proposed framework argues for: improving privacy and robustness, real-world deployment validation, and exploring hyperparameters. The authors in [18] designed a hybrid method combining advanced functional encryption with a Bayesian differential privacy technique for privacy-preserving federated learning. The advanced encryption function provided for federated learning secures the weight and the characteristics of the uploaded data of each participant. While the local privacy method in Bayesian differential privacy enables efficient data privacy accommodated on distributed datasets. Moreover, the authors added a sparse difference algorithm for differential gradient updates to enhance the transmission and the storage resources in a federated learning environment. Meanwhile, since no real-world deployment is tested. The model may face challenges to preserve efficiency in a scalable network, to train a unified model with the respect to data heterogeneity, to secure the model against advanced attacks, and to optimize the computational overhead required to ensure privacy with the proposed mechanisms. An asynchronous federated learning approach is proposed in [19]. The proposed model aims to preserve privacy in mobile edge IIoT networks by using the Iteration Model Design Update Strategy scheme. Resulting in reducing data shared with offline nodes, enabling real-time updates of the edge server model with online nodes, and securing data privacy from malicious nodes. Besides the authors included the Double-Weight Modification Strategy in the framework to enhance the model training and guarantee accurate and smoother model updates. A Gradient Compression was used for a more efficient model design. The framework tested with Python and EdgeCloudSim v4.0, revealing better efficiency, accuracy, and decreasing bandwidth consumption compared to existing solutions. Still, the approach has been assessed in simulation environment, which may not adequately represent the complexities, the model performance in scalable IoT networks, and adversarial attacks resistance. Another federated learning model is proposed in [20]. The authors provided a privacy-enhanced approach for heterogeneous data in IoT systems. A dual-model framework designed for IoT devices, where a Shared Model assists intercommunications is joined with a personalized model made from local aggregation for enhancing personalization while tacking data heterogeneity. Besides, to ensure privacy and secure data in training, a local Gradient Shielding algorithm is provided by performing local training of personalized models. Experiments with Hypernetworks and Benchmark Datasets demonstrate the feasibility of this solution. However, due to data heterogeneity and resource constraints of IoT devices, it's challenging to attain optimal performance and efficient solutions. FedPARL a tri-layer federated learning model is presented in [21]. It contributes to upgrading the training and learning in resource-constrained IoT devices by minimizing the model size with sample-based pruning. Besides, enabling according to resource availability variable local epochs. Furthermore, the proposed model uses a trust score to choose clients with better track records and sufficient resources. This article uses several tools such as Sample-Based Model Pruning, Trust Score Mechanism, and Federated Averaging Algorithm for lightweight and efficient model design. Although the proposed solution faces challenges related to handling the communication overhead of a scalable number of clients, and the complexity of the model implementation in a

resource-constrained environment.

3. Methodology

Most IoT researchers' contributions using Machine learning approaches are focused on Attack Detection solutions, IoT device profiling, and detecting Abnormal Behaviors of the IoT traffic. Addressing data privacy in IoT architecture is very challenging due to the massive data collected, the weak security measures, the lack of user awareness, and the risks of data breaches and cyberattacks. This section describes the methods used to design our lightweight optimized federated learning model tailored for data privacy in IoT architecture.

3.1. Dataset Description

In this part, we describe the dataset used in our proposal. To perform efficient machine learning model conception in an IoT environment. The datasets must be fulfilled with the following characteristics: a manageable number of features in order to reduce the computational costs, varied conditions of data collected, open and accessible datasets, small to moderate size of datasets to respect the resources constraints of IoT devices, and real-world traffic scenarios.

In this work, we use the CIC IoT dataset 2022 [22], a suitable dataset with the above characteristics, and recommended for data privacy analysis. The CIC IoT dataset 2022 (Canadian Institute for Cybersecurity) is an up-to-date dataset designed for IoT device identification and security analysis. This dataset is presented in CSV format to simplify the process of importation and analysis. In addition, it involves heterogeneous IoT devices with IP-based and non-IP-based devices, specifically: wearables, Smart Home devices, and industrial sensors. The dataset captured data over a real-world IoT environment in different sets of scenarios out of 60 IoT devices from various types: Audio, Camera, and home automation. For each IoT device, several datasets are provided, generated from various scenarios, and each dataset contains the same number of features: 48. The features set is presented in Table 1.

Table 1. Features set of CIC IoT dataset 2022

	Column in dataset		Column in dataset		Column in dataset
1	L4_tcp	17	source_port	33	var (Variance packet)
2	L4_udp	18	dest_port	34	q3 (third quartile packet)
3	L7_http	19	DNS_count	35	q1 (rst quartile packet)
4	L7_https	20	NTP_count	36	iqr (inter-quartile packet)
5	ip_padding	21	ARP_count	37	sum_e (Sum of frame)
6	ip_ralert	22	cnt (Number of Protocols)	38	min_e (Minimum ethernet)
7	port_class_src	23	ip_dst_count	39	max_e (Maximum ethernet)
8	port_class_dst	24	most_freq_dest_ip	40	med (Median frame)
9	pck_size	25	most_freq_prot	41	average (Average frame)
10	pck_rawdata	26	most_freq_sport	42	var_e (Variance frame)
11	ip_dst_new	27	most_freq_dport	43	q3_e (third quartile frame)
12	ethernet_frame_size	28	sum_et (Sum of packet size)	44	q1_e (rst quartile ethernet)
13	ethernet_frame_type	29	min_et (Minimum packet)	45	iqr_e (inter-quartile frame)
14	ttl	30	max_et	46	epoch_timestamp
15	total_length	31	med_et (Median packet)	47	inter_arrival_time
16	protocol	32	average_et	48	time_since_previously_displa yed_frame

3.2. Network Environment

The network configuration of the chosen dataset is described in this subsection. As shown in Figure 1, the network environment incorporates heterogeneous IoT devices involving end-user devices, firewalls, switches, and routers while supporting multiple communication protocols akin to IEEE 802.11(WI-FI), Zig-Bee, Z-Wave, and Bluetooth, in addition to numerous sub-networks, including both internal and external networks. A Cloud-based service can be displayed to simulate hybrid and modern network ecosystems. The configuration of the lab network is built with a 64-bit Windows machine. This machine has two network interface cards (NIC). The first one is linked to the network gateway offering access to the internet (WAN), and the other is connected to a plug-and-play network switch providing the local network for the IoT devices (LAN). For capturing and analyzing the network traffic, Wireshark is settled on the Windows machine to listen to each network interface and save the output traffic as a pcap file. As IoT devices need wired connections, Vera Plus, a smart automation hub, is linked to the plug-and-play switch, providing a coherent wireless IoT network with: Bluetooth, Z-Ware, Zig-Bee, Z-Wave, and IEEE 802.11(WI-FI).

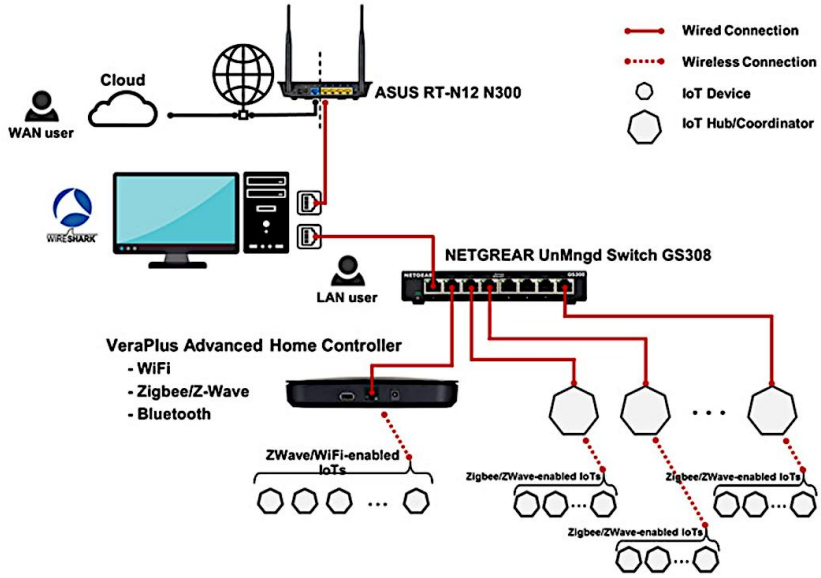


Fig.1. CIC IoT dataset 2022 network environment [22]

A. Scenarios for Collecting Traffic

To get the normal behavior and the vulnerabilities of IoT devices CIC Lab performs different network scenarios on various IoT devices. The data collected from the network traffic was captured under six types of experiments:

- Active: data captured throughout the day
- Idle: data captured during no human interaction periods, typically from late at night to early morning.
- Power: data captured when each device was powered on. For a duration of two minutes.
- Attacks: data captured under two types of attacks, Flood and RTSP-Brute Force.
- Interactions: All interactions of the device network activities were captured.
- Scenarios: data captured under six types of experiments to simulate Smart Home network activities.

B. Interactions Experiment

All the IoT lab devices have four types of interactions:

- Physical: utilizing manual commands of the IoT device
- LAN App: sharing the network of the IoT device and working with the companion app from the device
- WAN App: working with the companion app while using a different network from the IoT device.
- Voice: using the device's voice assistant to command devices.

3.3. Framework Architecture

The figure 2 provides an overview of the proposed model, addressing data privacy concerns by performing Federated Learning and TinyML in IoT architecture. We consider the realistic network scenarios of CIC Lab where the network traffic of the heterogeneous IoT devices has been captured by Wireshark tool. After analyzing the captured traffic, a comprehensive representation of the network traffic was provided in CIC IoT dataset 2022. For developing a data privacy model with federated learning and TinyML, the data needs to be prepared and analyzed in the edge to reduce the need for constant transmission of data to the cloud. Enabling efficiency, privacy, and scalability. On the edge, we carried out on each IoT device dataset multiple tasks to train the data for prediction and test the efficiency of our proposal. The first executed task is Features Engineering where we select some features, eliminate others, and add more relevant features for analysis. Then we conduct data preprocessing to guarantee that the input data for training are suitable, normalized, clean, and consistent. Further, we perform analysis using federated learning techniques and TinyML, to predict if the data of a specific traffic is private, and mentioning the privacy issue. Once designing the global model, the model is compressed with the respect to TinyML principles. For allowing the future use of our model in a constrained IoT device architecture. The last task is to send only non private data to the cloud for storage.

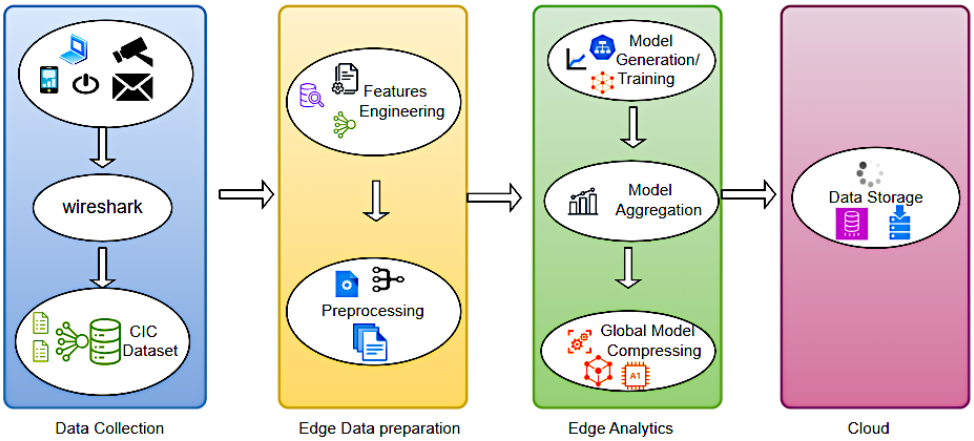


Fig.2. The proposed framework architecture overview

A. Features Engineering

In order to design data privacy models, features engineering is the most important step in manipulating the dataset to create and optimize the prediction model [23,24]. The features engineering design process steps of the model are as follows:

- Features Selection: decreasing the number of features provides numerous advantages regarding to model efficiency, lighter processing, easier data manipulation, and optimized computational resources. After analyzing the header information of the network traffic packets delivered in CIC dataset 2022, the subset of features selected for data privacy analysis is: 'L4_tcp', 'L4_udp', 'L7_http', 'L7_https', 'protocol', 'source_port', 'dest_port', 'most_freq_prot', 'most_freq_sport', 'most_freq_dport'.
- Adding two additional features for data privacy analysis, the first added feature is "is_it_private", it's a boolean feature that takes 0 if the traffic is safe, and takes the value 1 if the traffic is private. The second added feature is "privacy_issue", it is an integer feature giving the number of specific privacy issues of the traffic. Table 2 presents the list of the privacy issues in our model.

Table 2. List of the privacy issues tested

Privacy Issues	Number
Unsecured Web Services	01
Unsecured Transmission	02
Sensitive Information	03
Encrypted Data	04

B. Preprocessing

Multiple steps need to be performed to prepare the CSV dataset for training, with the purpose of guaranteeing that the data is structured, clean, and appropriate [25,26].

Exploring the datasets: Since each interaction experiment was repeated 3 times to generate sufficient network traffic, each IoT device has 3 different CSV datasets for every experiment. For example: Audio devices datasets provided for every device:

- Physical activities (Volume ON/ Volume OFF)
- LAN activities (Volume ON/ Volume OFF)
- WAN activities (Volume ON/ Volume OFF)
- VOICE activities (Volume ON/ Volume OFF)

In all, an Audio device will have 24 different datasets, a Camera device will have 18 different datasets, and a Home Automation will have 24 different datasets. We "concatenate" all the datasets of a single device into one dataset to perform federated learning on each device dataset as a large dataset provides the best model achievement.

Handling missing values: after selecting valuable features for analysis, we identify and manage the missing values by replacing them with the mean of the feature's existing values, in order to ensure model validity and prevent data leakage.

Features Scaling: in preprocessing, it is crucial to scale and normalize features to improve the performance of Gradient-based Algorithms. It helps gradient descent to find the optimal solution. In this work, we use the "Z-score" Normalization method.

Encode Categorical Features: converting categorical features to numerical forms is important, as machine learning algorithms take only numerical inputs. The "One-hot encoding" technique is the one we chose for our model

3.4. Implementation Details

In this section, we provide the implementation details of our proposal.

A. Loading the Datasets

Our model is performed on Audio device datasets. After the datasets' concatenation, each device has one global dataset containing all of its network traffic starting from Physical activities to WAN activities. The audio devices are:

- Amazon Alexa Echo Dot 1: 790 entries, memory usage: 314.9 KB
- Amazon Alexa Echo Dot 2: 1351 entries, memory usage: 538.4 KB
- Amazon Alexa Echo Spot: 4882 entries, memory usage: 1.9 MB
- Amazon Alexa Echo Studio: 1358 entries, memory usage: 541.2 KB
- Google Nest Mini: 12631 entries, memory usage: 4.9 MB
- Sonos One Speaker: 6569 entries, memory usage: 2.6 MB

B. Defining a Neural Network Model

Our neural network is set up with 3 layers: an input layer with 32 neurons, one hidden layer with 32 neurons each, and a classification output layer with 4 neurons, with softmax activation for multi-class classification. The input and hidden layers have ReLU as an activation function to support complex patterns learning in the network. In addition, 2 dropout layers are included with 50% dropout rate to prevent overfitting. The model is compiled with the Adam optimizer and Categorical Crossentropy loss function.

C. Setting Parameters and Evaluation Metrics

The parameters used while designing our federated learning model are presented in Table 3. For preventing the overfitting, two techniques are used: "Dropout", already mentioned in the previous part, and "Early Stopping" technique callback to monitor the validation loss inside the local device training process. This technique prevents overfitting, setting up with patience of 3 to stop the training if the validation loss does not increase after 3 epochs.

Table 3. List of parameters

Parameter	Value
Number of devices	6
Local epochs	5
Global epochs	5
Batch size	32
Hidden layers	1
Activation function	ReLU
Regularization	dropout
Classification function	softmax
Optimizer	adam
Loss function	Categorical crossentropy
Weights	Accuracy, loss
Datasets per device	1
Output classes	4
Training data size	80%

D. Training the Federated Learning Model

After setting the parameters and the evaluation metrics of our approach, we designed our federated learning model for an optimized decentralized data privacy training. Our training model is a Horizontal Federated Learning where the datasets of all devices have similar features and different samples [27,28]. As shown in Figure 3, an edge node is responsible for the initialization of a global Neural Network model. Then the model gets trained locally on a convenient IoT devices dataset. After the training, the returned updates of the locally trained model weights get aggregated into a new global model. The new model gets trained locally again for a new weight's aggregation journey. Our model training process is repeated for 5 rounds. Fewer rounds to reach a good accuracy level provides a more communication-efficient model.

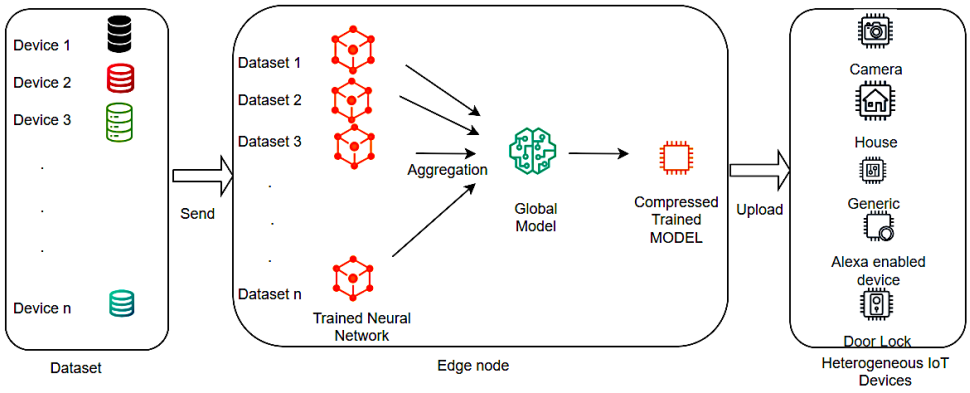


Fig.3. The tiny federated learning model training process

E. Tiny Model Configuration

To design a tiny compact model suitable for the resource constrained nature of IoT devices we used the Post-Training Optimization technique, where we converted our model to a lower precision format. Our tiny model architecture choice is Quantization, which allows for converting weights in INT8, and offers a significant increase in inference speed and reduction in the model size. The model size after reduction is: 0.01 MB.

4. Results and Discussion

In this section, we present the analysis of our model performance for addressing data privacy concerns in IoT architecture with federated learning and TinyML.

In Figures 4 and 5, an illustration of the training and validation performance of our model from device 1 and device 2 in the first round of federated learning before any aggregation. From this round we notice that the learning patterns are different between device 1 and device 2, presuming that their data distributions are different too. In fact, the learning in device 1 is faster than the learning in device 2. The similar patterns of training and validation metrics of devices reveal good generalization abilities despite the minimal learning gap, implying the need of enhancement. A potential of overfitting is shown in device 1 final epochs through the gap between the training and validation loss. Yet, from this round results analysis, we notice that succeeding rounds of federated averaging are crucial for increasing the global model performance.

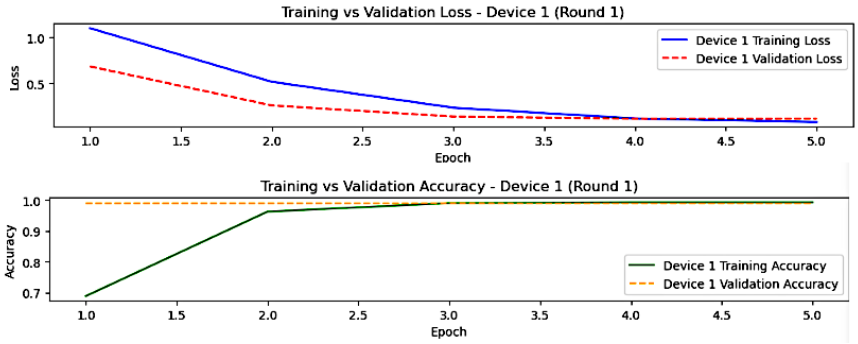


Fig.4. The model performance on device 1, round 1

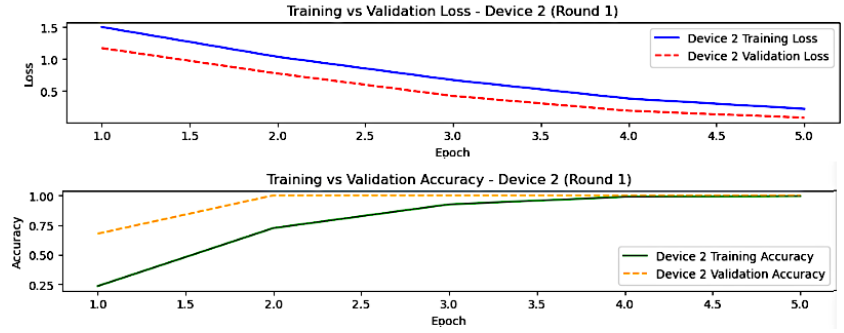


Fig.5. The model performance on device 2, round 1

We stick with the same devices, where round 2 of federated learning is presented in figures 6 and 7. In this round, we observe a lower loss compared to round 1, which means that the federated averaging technique upgraded our global model, leading to faster convergence. In addition, the overfitting is reduced and the generalization is better. Eventually, the device's performance appears more identical in this round due to federated averaging that helps to homogenize the models. However, the small gap remains from the training and validation, indicating the need for further enhancement.

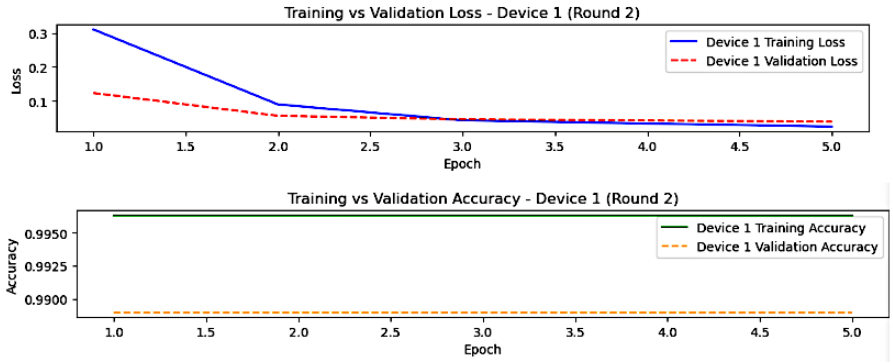


Fig.6. The model performance on device 1, round 2

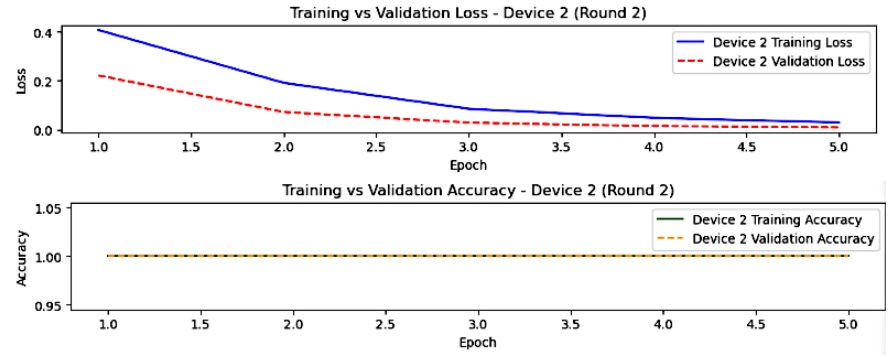


Fig.7. The model performance on device 2, round 2

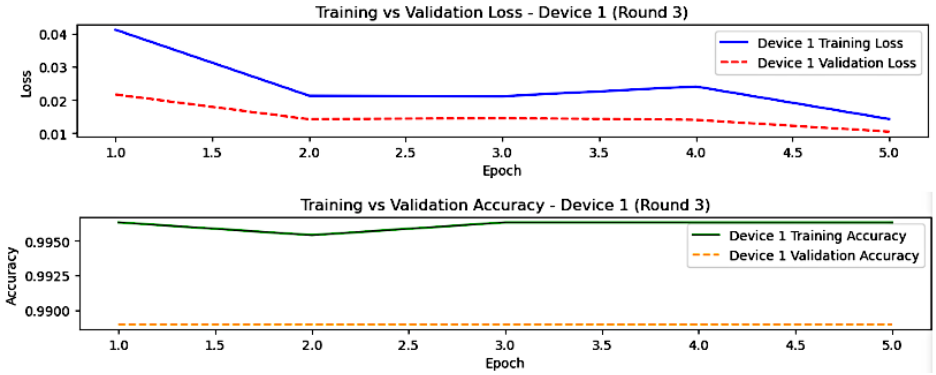


Fig.8. The model performance on device 1, round 3

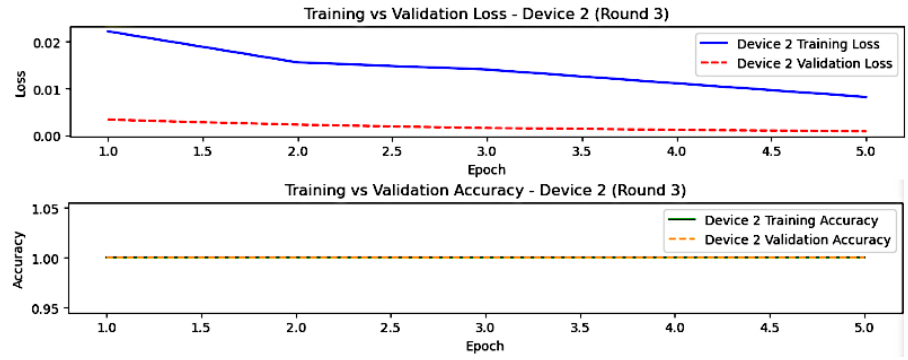


Fig.9. The model performance on device 2, round 3

Round 3 is shown in figures 8 and 9. In this round, the metrics values are very stable, indicating that device 1 and device 2 have greatly converged with a minimal overfitting for both devices. Although this is a good effect of federated averaging. The performance of devices is getting more similar, demonstrating the effectiveness of the federated learning process and the aggregation learning journey in achieving convergence, reducing overfitting, and improving generalization.

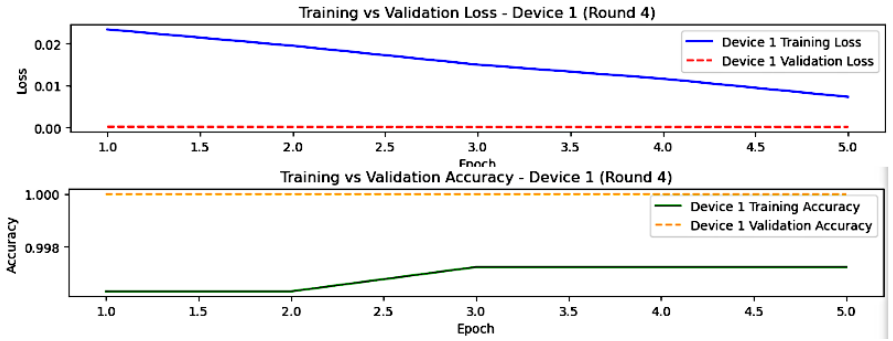


Fig.10. The model performance on device 1, round 4

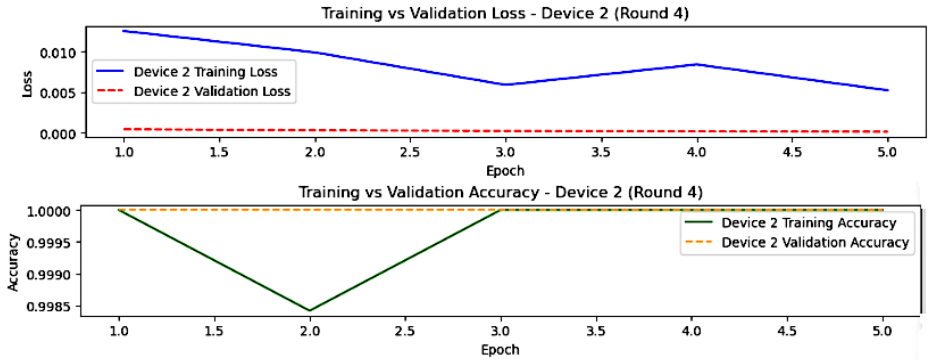


Fig.11. The model performance on device 2, round 4

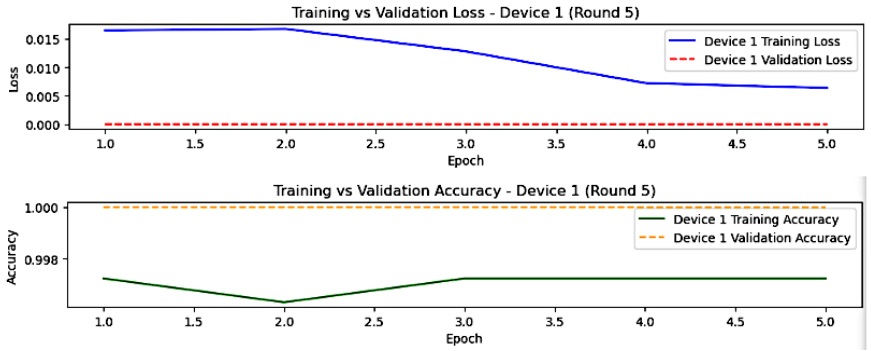


Fig.12. The model performance on device 1, round 5

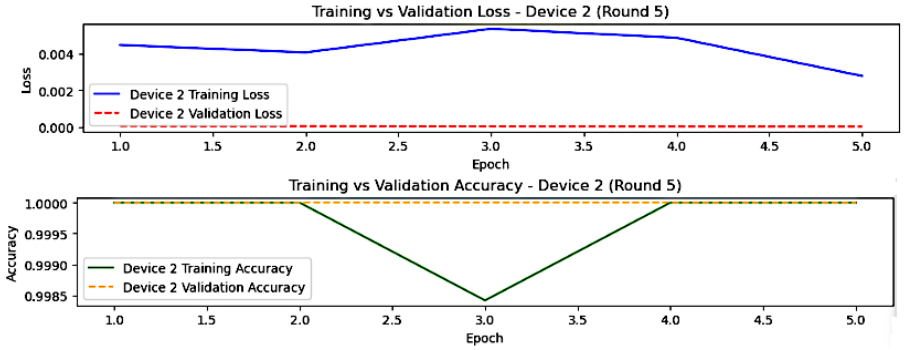


Fig.13. The model performance on device 2, round 5

By round 4, both devices' performance has gotten better stabilization, good generalization, and larger convergence as shown in figures 10 and 11. The overall performance is excellent, showing the triumph of federated learning technique. However, the slightest overfitting and the fluctuations in training accuracy of device 2 argue for exploring the particular data and the learning parameters of this device.

In the last round of federated learning of our model, the devices have fully converged with an almost non existent overfitting, and an excellent generalization as shown in figures 12 and 13. The heterogeneous nature of data or the learning process are likely responsible for device 2 training accuracy fluctuations.

After five rounds of the federated learning process. The results of the model's performance through all the six devices are similar. Table 4 shows the accuracy and loss values of all devices per round.

Table 4. The model metrics results of each device per round

Rounds	Device	Accuracy	Loss	Device	Accuracy	Loss	Device	Accuracy	Loss
Round1	Device1 Device4	0.9936 0.9818	0.0803 0.0527	Device2 Device5	0.9937 0.9947	0.2159 0.0147	Device3 Device6	0.9991 0.9987	0.0741 0.0104
Round 2	Device1 Device4	0.9963 0.9695	0.0235 0.0469	Device2 Device5	1.0000 0.9917	0.0282 0.0177	Device3 Device6	1.0000 0.9998	0.0087 0.0054
Round 3	Device1 Device4	0.9963 0.9862	0.0142 0.0335	Device2 Device5	1.0000 0.9988	0.0082 0.0114	Device3 Device6	1.0000 1.0000	0.0054 0.0024
Round 4	Device1 Device4	0.9972 0.9867	0.0072 0.0294	Device2 Device5	1.0000 0.9987	0.0052 0.0087	Device3 Device6	1.0000 0.9998	0.0026 0.0031
Round 5	Device1 Device4	0.9972 0.9867	0.0064 0.0271	Device2 Device5	1.0000 0.9988	0.0028 0.0071	Device3 Device6	1.0000 1.0000	0.0023 0.0016

As shown in the figure 14. The federated averaging technique has efficiently merged the insights for all the devices, demonstrated by the increase in accuracy and the decrease in loss at the first rounds. Upon rounds 3 and 4, the model has fully converged with an excellent generalization, despite the tiny gap between training and validation due to the heterogeneity of devices data. Moreover, the learning patterns of accuracy imply changes at specific rounds as certain aggregation updates or devices data are impactful. For example, we could stop the federated learning process after rounds 3 or 4.

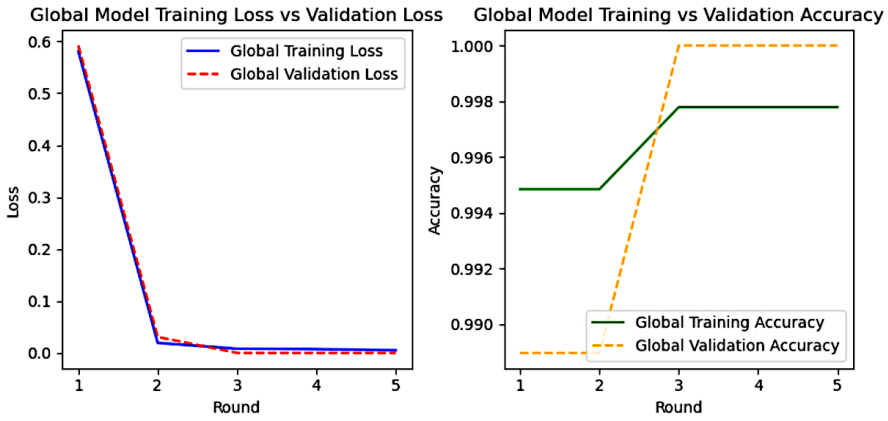


Fig.14. Global model training vs validation metrics results

In conclusion. The federated learning technique has successfully trained our model. The results demonstrate our model convergence; good generalization will obtaining excellent performance. Which proves that the federated averaging technique successfully combines learning from distributed data over various devices.

To strengthen credibility, we perform an analysis of variance (ANOVA), it's the most common statistical analysis method for performance variations. We apply the ANOVA test on the accuracy and the loss results of our framework devices to inspect a notable difference and the effect of a device on a metric result. With a significance level of 0.05, the accuracy results display the p-value: 0.000000208179, while the loss results display the p-value: 0.517027. From the p-values and the post-hoc analysis results, we can affirm that statistically, there is a significant difference in the mean accuracy across different devices. Meanwhile, Device 4 has considerably lower average accuracy compared to the other devices, as shown in Figure 15. Device 4 is the weakest performer with regard to accuracy. On the other hand, statistically, the loss performance of the devices is similar, indicating the impact of unbalanced data distribution (the unbalanced privacy issues distribution support) on accuracy performance.

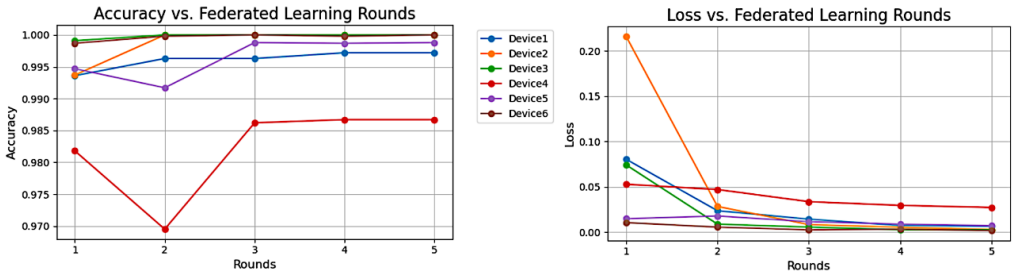


Fig.15. Accuracy and loss results vs federated learning rounds

Inference time: For examining the model performance, a simulation of inference on our tiny model shows that the time taken by the model to process the input and generate the output is 0.004001 seconds, which is lower than the inference time of existing scientific contribution models using Federated Learning in an IoT environment.

As the model size after TinyML compression is 0.01MB, and the inference time is 0.004001 seconds, we can confirm that the federated learning architecture combined with TinyML has successfully provided a model with:

- Communication efficiency: considering the final model size, which shows a perfect small model that can easily fit in the flash memory of any microcontroller. Exchanging the small model updates during the federated learning process will demand slightest data transfer.
- Computationally efficiency: the remarkably fast inference time shows that our tiny model can be executed swiftly on low-power IoT devices. Ensuring that the device can respond to events instantly with a lower energy consumption.
- Privacy-preserving data training: as the model data are trained locally, it never leaves the microcontroller of the IoT device due to TinyML. Basically, this will provide a lightweight privacy-preserving technique to train sensitive data.
- Reduced attack points: the fewer rounds of federated learning used, decreases the whole number of updates an attacker can intercept and explore through the learning process. Moreover, the compression technique provides less informative, smaller updates with a tiny amount of sensitive data that may be extracted by an attacker from the communicated updates. Furthermore, a tiny model is weak at memorizing specified data during training, which decreases the potential inference to specific data by an attacker.

5. Challenges and Future Research Directions

Handling data privacy concerns in IoT architecture is a challenging task that demands a federation of multiple techniques and measures. This work investigated the effectiveness of combining Federated Learning and TinyML for lightweight data privacy model deployment in IoT. Still, certain challenges need to be addressed in our future work. For instance, gradient leakage, model inversion, membership inference attacks, and adversarial robustness. These challenges can be faced by embracing some safeguard techniques, such as differential privacy for efficient communication, a lightweight secure aggregation protocol, and Homomorphic encryption for direct encryption of model updates. Besides, evaluations in a real-world IoT environment should be considered in our future contributions to validate the robustness, practicality, security, and scalability of the proposal.

6. Conclusions

In this work, we presented the development and experiments of our lightweight approach for addressing data privacy in IoT architecture. The proposed approach exploits federated learning for decentralized model training, combined with TinyML. This combination preserves privacy and reduces the communication overhead by training data locally. Moreover, it allows the optimization and deployment of machine learning models on hardware constraints devices. Further, our approach gets deeper for addressing data privacy by considering and analyzing the network traffic of heterogeneous IoT devices to extract relevant features for data privacy analysis. Four privacy issues have been analyzed: unsecured web services traffic, unsecured transmissions, sensitive information exchange, and encrypted data in transit. Despite the heterogeneity of data and devices in IoT architecture. The experimentation results of our model demonstrate the viability of combining federated learning and TinyML for a lightweight data privacy model deployment.

Author Contributions Statement

Hiba KANDIL – Conceptualization, data curation, formal analysis, Methodology, Model Training and Performance Evaluation, Writing review and editing.
Hafssa BENABOUD – Conceptualization, Supervision, validation.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement
The authors declare no conflict of interest.

Funding Declaration
The authors received no financial support for this research.

Data Availability Statement
The data that support the findings of this study are available in the Canadian Institute for Cybersecurity (CIC) Dataset Repository at https://cicresearch.ca/IOTDataset/CIC_IOT_Dataset2022/, reference number S. Dadkhah et al, "Towards the development of a realistic multidimensional IoT profiling dataset", Submitted to: The 19th Annual International Conference on Privacy, Security & Trust (PST2022) August 22-24, 2022, Fredericton, Canada.

Ethical Declarations
This research does not involve human subjects or animals.

Acknowledgments
We wish to express our sincere gratitude to the Canadian Institute for Cybersecurity (CIC) for providing CIC IoT dataset 2022, which was essential for our contribution, the dataset is available at: https://cicresearch.ca/IOTDataset/CIC_IOT_Dataset2022/.

Declaration of Generative AI in Scholarly Writing
The authors declare no Generative AI or AI-assisted technologies were used in the writing process of this paper.

Abbreviations
The following abbreviations are used in this manuscript:

IoT- Internet of Things
TinyML - Tiny Machine Learning
MCU - microcontroller unit
IIoT - Industrial Internet of Things
CSV - Comma-separated values
IP - Internet Protocol
NIC - Network Interface Cards.
WAN - Wide Area Network
LAN – Local Area Network
ReLU - Rectified Linear Unit
ANOVA - ANalysis of VAriance

References

[1] T. Bouraffa and K.-L. Hui, "Regulating Information and Network Security: Review and Challenges," ACM Comput. Surv., vol. 57, no. 5, p. 126:1-126:38, Jan. 2025, doi:10.1145/3711124.
[2] P. Sun, Y. Wan, Z. Wu, Z. Fang, and Q. Li, "A survey on privacy and security issues in IoT-based environments: Technologies, protection measures and future directions," Computers & Security, vol. 148, p. 104097, Jan. 2025, doi:10.1016/j.cose.2024.104097.
[3] P. Joshi and S. Joshi, "Security Challenges in Internet of Things," in Advanced Cyber Security Techniques for Data, Blockchain, IoT, and Network Protection, IGI Global Scientific Publishing, 2025, pp. 355–392. doi:10.4018/979-8 3693-9225-6.ch014.
[4] J. V, V. Balan, and B. V. S, "Security Issues in IoT: Perspective Review," Computer Networks and Communications, pp. 101–117, Mar. 2025, doi:10.37256/cnc.3120256140.
[5] H. Kandil and H. Benaboud, "Using Machine Learning to Deal with Privacy and Confidentiality in Internet of Things: An Overview," in Innovations in Smart Cities Applications Volume 8, M. Ben Ahmed, B. A. Abdelhakim, İ. R. Karaş, and K. Ben Ahmed, Eds., Cham: Springer Nature Switzerland, 2025, pp. 774–783. doi: 10.1007/978-3 031-88653-9 73.
[6] M. Ebrahimi, R. Sahay, S. Hosseinalipour, and B. Akram, "The Transition from Centralized Machine Learning to Federated Learning for Mental Health in Education: A Survey of Current Methods and Future Directions," Jan. 20, 2025, arXiv: arXiv:2501.11714. doi:10.48550/arXiv.2501.11714.
[7] N. Jia et al. "A Comprehensive Survey on Communication-Efficient Federated Learning in Mobile Edge Environments,"IEEE Communications Surveys & Tutorials, pp. 1–1, 2025, doi:10.1109/COMST.2025.3535957.

[8] S. Chitimoju, "Federated Learning in Cybersecurity: Privacy-Preserving AI for Threat Detection," International Journal of Digital Innovation, vol. 6, no. 1, Art. no. 1, Jan. 2025, Accessed: Mar. 23, 2025. [Online]. Available: https://researchworkx.com/index.php/ijdi/article/view/76.
[9] R. Dembani et al. "Agricultural data privacy and federated learning: A review of challenges and opportunities," Computers and Electronics in Agriculture, vol. 232, p. 110048, May 2025, doi:10.1016/j.compag.2025.110048.
[10] J. Zhao et al., "The Federation Strikes Back: A Survey of Federated Learning Privacy Attacks, Defenses, Applications, and Policy Landscape," ACM Comput. Surv., Mar. 2025, doi:10.1145/3724113.
[11] M. Arif and M. Rashid, "A Literature Review on Model Conversion, Inference, and Learning Strategies in EdgeML with TinyML Deployment," Computers, Materials & Continua, pp. 1–10, Jan. 2025, doi:10.32604/cmc.2025.062819.
[12] Dr. L. Dutta and S. Bharali, "TinyML Meets IoT: A Comprehensive Survey," Internet of Things, vol. 16, p. 100461, Dec. 2021, doi:10.1016/j.iot.2021.100461.
[13] H. Kandil, W. Bouimejane, M. Mouhcine, and H. Benaboud, "An Anomaly-based Intrusion Detection System for IoT environments using autoencoder neural networks and TinyML," IJCA, vol. 187, no. 59, pp. 9–15, Nov. 2025, doi: 10.5120/ijca2025926000.
[14] C. N. da Silva and C. V. S. Prazeres, "Tiny Federated Learning for Constrained Sensors: A Systematic Literature Review," IEEE Sensors Reviews, pp. 1–16, 2025, doi:10.1109/SR.2025.3548547.
[15] M. Ficco, A. Guerriero, E. Milite, F. Palmieri, R. Pietrantuono, and S. Russo, "Federated learning for IoT devices: Enhancing TinyML with on-board training," Information Fusion, vol. 104, p. 102189, Apr. 2024, doi:10.1016/j.inffus.2023.102189.
[16] K. Kopparapu, E. Lin, J. G. Breslin, and B. Sudharsan, "TinyFedTL: Federated Transfer Learning on Ubiquitous Tiny IoT Devices," In 2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), Mar. 2022, pp. 79–81, doi:10.1109/PerComWorkshops53856.2022.9767250.
[17] H. Ren, X. Li, D. Anicic, and T. A. Runkler, "TinyMetaFed: Efficient Federated Meta-Learning for TinyML," Sep. 28, 2023, arXiv: arXiv:2307.06822. doi:10.48550/arXiv.2307.06822.
[18] L. Yin, J. Feng, H. Xun, Z. Sun, and X. Cheng, "A Privacy-Preserving Federated Learning for Multi-party Data Sharing in Social IoTs," IEEE Trans. Netw. Sci. Eng., vol. 8, no. 3, pp. 2706–2718, Jul. 2021, doi:10.1109/TNSE.2021.3074185.
[19] J. O. Odeh, X. Yang, C. I. Nwakanma, and S. Dhelim, "Asynchronous Privacy-Preservation Federated Learning Method for Mobile Edge Network in Industrial Internet of Things Ecosystem," Electronics, vol. 13, no. 9, p. 1610, Apr. 2024, doi:10.3390/electronics13091610.
[20] Z. He, F. Zhang, Y. Li, Y. Cao, and Z. Cai, "Privacy-Enhanced Personalized Federated Learning With Layer-Wise Gradient Shielding on Heterogeneous IoT Data," IEEE Internet of Things Journal, vol. 11, no. 15, pp. 25771–25781, Aug. 2024, doi:10.1109/JIOT.2024.3391342.
[21] A. Imteaj and M. H. Amini, "FedPARL: Client Activity and Resource-Oriented Lightweight Federated Learning Model for Resource-Constrained Heterogeneous IoT Environment," Front. Commun. Netw., vol. 2, Apr. 2021, doi:10.3389/frcmn.2021.657653.
[22] S. Dadkhah, H. Mahdikhani, P. K. Danso, A. Zohourian, K. A. Truong, and A. A. Ghorbani, "Towards the Development of a Realistic Multidimensional IoT Profiling Dataset," in 2022 19th Annual International Conference on Privacy, Security & Trust (PST), Aug. 2022, pp. 1–11. doi: 10.1109/PST55820.2022.9851966.
[23] D. Shah, J. Wang, and Q. P. He, "Feature engineering in big data analytics for IoT-enabled smart manufacturing– Comparison between deep learning and statistical learning," Computers & Chemical Engineering, vol. 141, p. 106970, Oct. 2020, doi:10.1016/j.compchemeng.2020.106970.
[24] S. L et al., "A Multiclass Attack Classification Framework for IoT Using Hybrid Deep Learning Model. — EBSCO host." Accessed: Mar. 24, 2025. doi:10.54216/JCIM.150112.
[25] N. Sirisha et al. "IoT-based data quality and data preprocessing of multinational corporations," The Journal of High Technology Management Research, vol. 34, no. 2, p. 100477, Nov. 2023, doi:10.1016/j.hitech.2023.100477.
[26] M. Shahnawaz and M. Kumar, "A Comprehensive Survey on Big Data Analytics: Characteristics, Tools and Techniques," ACM Comput. Surv., vol. 57, no. 8, p. 196:1-196:33, Mar. 2025, doi:10.1145/3718364.
[27] J. L. Salmeron and I. Ar´ evalo, "Concurrent vertical and horizontal federated learning with fuzzy cognitive maps," Future Generation Computer Systems, vol. 162, p. 107482, Jan. 2025, doi:10.1016/j.future.2024.107482.
[28] Z. Sun, J. Xu, J. Li, and T. Zhang, "Privacy Protection Authentication Protocol for Consumer Internet of Things in Horizontal Federated Learning Environment," IEEE Transactions on Consumer Electronics, pp. 1–1, 2025, doi:10.1109/TCE.2025.3525497.

Authors' Profiles



PHD Student Hiba Kandil, Department of Computer Science, IPSS, Faculty of Sciences, Mohammed V University in Rabat, Avenue Ibn Battouta B.P. 1014 RP, Rabat, Morocco.
(ORCID ID https://orcid.org/0009-0008-1165-2596)

Major interests: Internet of Things, Security, Network Security, Machine Learning, and Networking.



Professor Hafssa Benaboud, Department of Computer Science, IPSS, Faculty of Sciences, Mohammed V University in Rabat, Avenue Ibn Battouta B.P. 1014 RP, Rabat, Morocco.
(ORCID ID https://orcid.org/0000-0002-5027-8217)

Major interests: Network Protocols, Network Security, Traffic Analysis, Internet of Things, Cloud Computing, and Quality of Service.

How to cite this paper: Hiba Kandil, Hafssa Benaboud, "Addressing Data Privacy Concerns in IoT Architecture with Federated Learning and TinyML", International Journal of Computer Network and Information Security(IJCNIS), Vol.18, No.2, pp.196-209, 2026. DOI:10.5815/ijcnis.2026.02.11