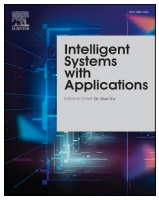
Intelligent Systems with Applications 25 (2025) 200462

Contents lists available at ScienceDirect

Intelligent Systems with Applications

journal homepage: www.journals.elsevier.com/intelligent-systems-with-applications



Computationally Efficient Deep Federated Learning with Optimized Feature Selection for IoT Botnet Attack Detection

Lambert Kofi Gyan Danquah [a,*], Stanley Yaw Appiah [a], Victoria Adzovi Mantey [a], Iddrisu Danlard [b], Emmanuel Kofi Akowuah [a]

[a] Department of Computer Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana
[b] Department of Electrical and Electronics Engineering, Sunyani Technical University, Sunyani, Ghana

ARTICLE INFO

Keywords:
Federated learning
Computational complexity
Differential privacy
Intrusion detection
Internet of Things

ABSTRACT

Internet of Things (IoT) is a technology that has revolutionized various fields, offering numerous benefits, such as remote patient monitoring, enhanced energy efficiency, and automation of routine tasks in homes. However, unsecured IoT devices are susceptible to botnet-based attacks such as distributed denial of Service (DDoS). Conventional machine learning models used for detecting these attacks compromise data privacy, prompting the adoption of federated learning (FL) to improve privacy. Yet, most FL-based cyberattack detection models proposed for IoT environments do not address computational complexity to suit their deployment on resource-constrained IoT edge devices. This paper introduces an FL model with low computational complexity, designed for detecting IoT botnet attacks. The study employs feature selection and dimensionality reduction to minimize computational complexity while maintaining high accuracy. First, an extreme gradient boosting model, trained with repeated stratified k-fold cross-validation, is used to select the optimal features of the botnet dataset based on feature importance. Principal component analysis is then used to reduce the dimensionality of these features. Finally, a differentially private multi-layer perceptron is trained locally by four FL clients and aggregated through federated averaging (FedAvg) to form a global Mirai botnet attack detection model. The model achieved an accuracy, precision, recall, and F1-score of 99.93 %, an area under the curve of 1.0, and 8,612 floating-point operations, contributing to 87.34 % reduction in computational complexity compared to the previous work. The proposed model is well-suited for detecting botnet attacks in smart homes, smart grids, and environments where resource-constrained IoT edge devices are deployed.

1. Introduction

The Internet of Things (IoT) is a framework where everyday objects are furnished with the ability to identify one another, sense specific environmental conditions, process data, and network with one another and other devices to accomplish a clearly defined objective without human intervention (Whitmore et al., 2015; Kumar et al., 2022). The adoption of 5G wireless communication technology, renowned for its faster data rates, expanded capacity, and reduced latency, has led to a substantial increase in the utilization of IoT devices (Sofana Reka et al., 2019). The numerous benefits IoT devices offer have also contributed to their exponential growth and adoption in various sectors. For instance, in smart transportation, IoT devices such as advanced driver assistance systems have been adopted to reduce accidents caused by drivers (Lombardi et al., 2021). Similarly, in smart healthcare, wearable devices facilitate remote patient health monitoring (Bhushan et al., 2023; Pratap Singh et al., 2020). Patients can now be observed via these devices that provide real-time information on key health indicators such as blood pressure and heart rate (Devi et al., 2023). In the energy sector, smart meters reduce costs, enhance scalability, and refine power management efficiency (Popoola et al., 2021). Disaster management uses early warning systems to detect and alert individuals about impending disasters (Adeel et al., 2018). The number of connected IoT devices worldwide is estimated to grow to over 30 billion by 2027 (Canavese et al., 2024). As the number of IoT devices increases, so does the frequency of cyberattacks targeting them. Attackers have improved their skills in exploiting vulnerabilities within these devices, resulting in a sharp increase in threats.

* Corresponding author.
E-mail addresses: lkgdanquah1@st.knust.edu.gh (L.K.G. Danquah), syappiah1@st.knust.edu.gh (S.Y. Appiah), vamanyey@st.knust.edu.gh (V.A. Mantey), iddrisu.danlard@stu.edu.gh (I. Danlard), ekakowuah.coe@knust.edu.gh (E.K. Akowuah).

https://doi.org/10.1016/j.iswa.2024.200462
Received 1 June 2024; Received in revised form 9 November 2024; Accepted 26 November 2024
Available online 30 November 2024
2667-3053/© 2024 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Botnet attacks are among the most severe cyberattacks targeting IoT devices (Hussain et al., 2021). A botnet is a coordinated network of malware-infected devices, often referred to as bots or zombies, controlled remotely by a botmaster using a command and control system (Popoola et al., 2021). The botmaster issues control commands to deploy the bots for attacks and to receive feedback from them, including updates on the status of ongoing attacks (Koroniotis et al., 2019). Research indicates that a botnet can consist of millions of infected IoT devices, which can be exploited for various cyberattacks, including distributed denial of service (DDoS) attacks, spying, keylogging in smart homes, and demand manipulation in smart grids (Vignau et al., 2021). Mirai is one of the prominent malwares that infects IoT devices and creates a botnet with them to launch large-scale DDoS attacks. In 2016, for instance, a significant DDoS attack was orchestrated by a Mirai botnet against the French cloud company OVH, reaching a bandwidth consumption exceeding 1 TB/s. It was also used to launch another DDoS attack targeting the Dyn Domain Name Service (DNS) servers (Kolias et al., 2017).

An intrusion detection system (IDS) is often employed to mitigate such attacks on IoT devices, examining network traffic to determine whether incoming packets are legitimate or malicious (Saharkhizan et al., 2020). However, the conventional IDS uses signatures of already known attacks to detect intrusions, so it does not perform well in detecting zero-day attacks (Prazeres et al., 2023). Consequently, several researchers have suggested machine learning (ML) techniques in an effort to increase the accuracy of IoT botnet attack detection (Nuaimi et al., 2023). Though these techniques improve the accuracy of botnet attack detection, extensive centralized data is often not readily available to train them. Sensitive data in sectors such as healthcare, power, and manufacturing must be transmitted to external servers to train these ML models, which compromises data privacy (Imteaj & Amini, 2022).

To overcome this privacy problem and comply with data-sharing regulations, federated learning (FL) was introduced (Han et al., 2024). FL overcomes this challenge by sending an initial ML model to the network edge devices to be trained on their local data. After the model is trained on each device's data, their weights are transmitted to a central server, where they are aggregated to form a single global model. The updated global model weights are finally transmitted to each device participating in the federated learning, allowing the model to accomplish its designated task without sharing local data (Agrawal et al., 2022). Several authors have recently suggested federated learning-based methods for IoT cyberattack detection.

For instance, Tian et al. (2021) introduced an asynchronous federated learning (FL) framework utilizing a denoising autoencoder (DAE) as the local model for detecting IoT botnet attacks. In asynchronous global model aggregation, delays occur in aggregating local model weights from some clients. To address this, the authors incorporated Taylor expansion during the model aggregation stage to compensate for the delay. This approach effectively reduced aggregation delays and resulted in an accuracy of 89.5 %.

Moulahi et al. (2022) introduced a blockchain-assisted federated learning model to detect intrusions in the Internet of Vehicles. The parameters of the trained FL local models were sent to a central server for aggregation through a blockchain system to improve model security. Four machine learning models were used for the experiment to decide the best local model to be used in the federated learning system. Among the four, the support vector machine (SVM) performed the best, achieving an accuracy of 82.45 % on the VeReMi dataset.

Sarhan et al. (2022) introduced a method for sharing cyber-threat intelligence based on federated learning. Their study evaluated the performance of a deep neural network (DNN) and a long short-term memory (LSTM) model. The FL system based on a DNN outperformed the LSTM for binary classification with an accuracy of 93.08 %. In contrast, the LSTM outperformed the DNN in multi-class classification with an accuracy of 94.61 %.

de Caldas Filho et al. (2023) integrated a host intrusion detection and prevention system (HIDPS) with a network-based intrusion detection system to combat zero-day DDoS attacks. The HIDPS relied on attack signatures, while the network-based system implemented a federated one-dimensional convolutional neural network (1D-CNN) to identify botnet-orchestrated DDoS attacks. Patterns detected by the federated learning model were then incorporated into the signature database for host-based intrusion detection and prevention, enabling the model to detect DDoS attacks with an accuracy of 89.753 %.

Abdel-Basset et al. (2023) introduced a lightweight federated learning model for IoT malware attack detection. The model was built on a hybrid architecture, combining a transformer neural network with a convolutional neural network. Additionally, it leveraged the federated averaging algorithm to aggregate local models for IoT malware detection. The model achieved an accuracy of 94 % when six clients were involved in the federated learning. The study also assessed the model's complexity based on the number of trainable parameters and inference time.

Houda et al. (2023) presented a secure federated deep neural network (DNN) tailored for intrusion detection within a software-defined network (SDN) setup. They deployed a smart contract on Ethereum's blockchain to bolster the FL model's security, with SDN nodes serving as collaborators in the smart contract and also functioning as clients in the FL process. Each client applied secure multi-party computation to encrypt the local model parameters before transmitting them to secure aggregators to form a global model with an accuracy of 89 %.

Zainudin et al. (2023) developed a federated learning model with low computational complexity to detect cyberattacks in industrial cyber-physical systems. They utilized the Pearson Correlation Coefficient (PCC) and chi-square to select the top nineteen features from the dataset. Following feature selection, they employed a hybrid deep learning model that comprises a convolutional neural network and a multi-layer perceptron for intrusion detection. Their proposed model achieved an accuracy of 95.41 %, with a computational complexity of 68,000 floating-point operations (FLOPs) and 7,140 trainable parameters.

Salim et al. (2024) introduced a federated learning approach for cyberattack detection in satellite communication systems, employing a deep autoencoder (DAE) within their model framework. The DAE's architecture was tailored to minimize computational complexity by using a moderate number of parameters. Additionally, the study integrated differential privacy into the federated learning process to safeguard against inference attacks during aggregation. Although the authors aimed to reduce computational complexity, they did not explicitly measure this complexity.

Most papers proposing federated learning models for intrusion detection did not consider or evaluate computational complexity (Moulahi et al., 2022; Houda et al., 2023; de Caldas Filho et al., 2023). The few that measured complexity did not incorporate any protection mechanism for model aggregation (Abdel-Basset et al., 2023; Zainudin et al., 2023). Moreover, some proposed approaches did not include feature selection techniques to enhance the model's ability to detect intrusions. The performance of many related works was also examined using only a few metrics (Regan et al., 2022; Rashid et al., 2023), which do not provide enough information regarding the model's performance. Furthermore, none of the presented models aimed to solve these challenges simultaneously. On this premise, this paper proposes a federated learning model tailored for IoT botnet attack detection with low computational complexity. The significant contributions of the paper are as follows:

i. Development of a federated learning model that employs a lightweight Multi-Layer Perceptron to detect IoT botnet attacks while preserving data privacy.
ii. Utilization of XGBoost to select optimal features for detecting IoT botnet attacks, and reduction of their dimensionality using

Principal Component Analysis to minimize computational complexity.

iii. Integration of differential privacy into the FL-based intrusion detection system (IDS) to enhance data privacy.

iv. Conducting experiments to assess the performance of the FL-based IDS model based on accuracy, precision, recall, F1 score, area under the curve, and floating-point operations using the Mirai-based N-BaIoT dataset.

The rest of this paper is structured as follows: Section 2 details the methodology for implementing our proposed method, including the model implementation pipeline and performance metrics. In Section 3, the performance of the proposed model is evaluated and compared with state-of-the-art models. Section 4 presents the paper's conclusion.

2. Materials and methods

This paper proposes a low-complexity federated learning model for detecting IoT botnet attacks. The materials and procedures used to implement the proposed FL-based IDS are detailed in this section.

Four federated learning clients were assigned portions of the N-BaIoT dataset. A lightweight deep learning model was then trained on each client's data. Finally, the parameters of these trained local models were aggregated using federated averaging to form a global model for IoT botnet attack detection. The critical materials employed in implementing the proposed model include Python version 3.11.2, TensorFlow and Keras version 2.14.0, and TensorFlow-Privacy version 0.9.0. The workflow for implementing the proposed model is shown in Fig. 1.

2.1. Dataset description

In this paper, we used the N-BaIoT dataset provided by Meidan et al. (2018) to evaluate our proposed model. This dataset contains botnet attack traffic from seven IoT devices infected with the Mirai malware. The devices include a doorbell, a thermostat, a baby monitor, and four security cameras. Attacks launched by the botmaster using the Mirai-infected IoT devices include automatic scanning for vulnerable devices (Mirai scan), ACK flooding (Mirai ACK), SYN flooding (Mirai SYN), user datagram protocol (UDP) flooding (Mirai UDP), and UDP flooding with fewer options optimized for higher packets per second (Mirai UDP Plain).

2.2. Data preprocessing

The dataset used for training a machine learning model must be processed into a suitable format to improve the model's ability to identify patterns for accurate predictions (Zainudin et al., 2023). To this end, various data wrangling techniques were employed to prepare the dataset for model training and prediction. These techniques include data cleaning, outlier removal, and data normalization, among others.

2.3. Data cleaning

The data wrangling process begins with cleaning the raw N-BaIoT dataset. Although this dataset contained no missing values, it included numerous duplicate rows, which were removed to reduce redundancy and improve the efficiency of model training (Rashid et al., 2023).

2.4. Outlier elimination

First, z-score standardization was applied to identify values in each feature that lie outside three standard deviations from the mean, which are regarded as outliers. Then, all rows in the dataset with such values, totaling 87,197 and accounting for approximately 2.38 % of the dataset, were eliminated. The z-score is computed as follows:

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

where x represents the value of a particular data point in the feature, μ denotes the mean of the feature, and σ indicates its standard deviation (Rashid et al., 2023).

2.5. Data normalization

After the outlier removal, min-max normalization was applied to the resulting data. Min-Max normalization utilizes the minimum value and the maximum value from the dataset's features to rescale all their respective data points, ensuring they fall within the range of 0 to 1 (Sarhan et al., 2022). The equation below denotes how the normalized data point is computed using min-max normalization:

$$\mathrm{MM}(x_n) = \frac{x_n - \min(x)}{\max(x) - \min(x)} \tag{2}$$

where x_n represents a data point for the feature x and $\mathrm{MM}(x_n)$ represents

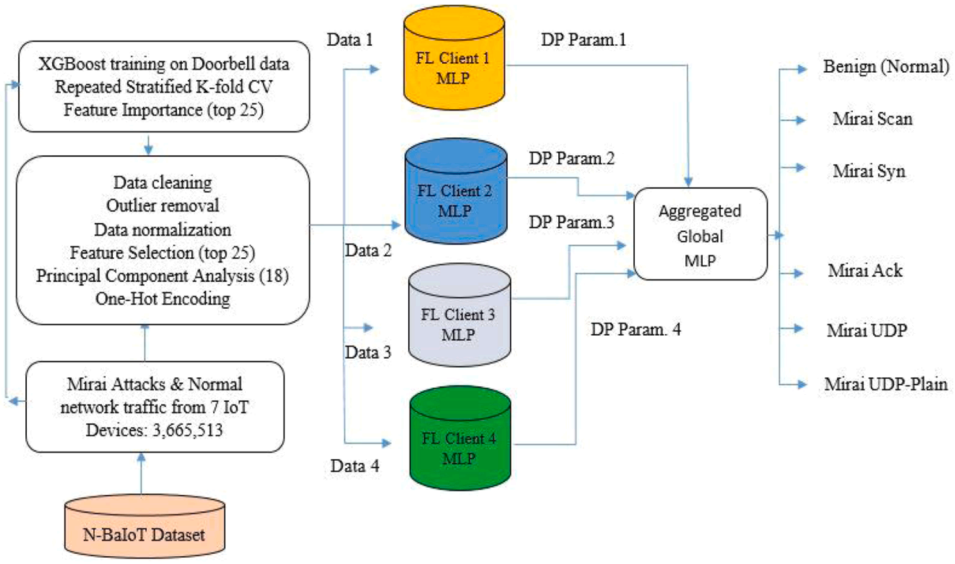


Fig. 1. Workflow of the Proposed FL-based Intrusion Detection Framework.

its corresponding normalized value.

2.6. Feature selection and dimensionality reduction

To ensure that the proposed federated learning model has a low computational complexity to be suitable for deployment on IoT edge devices while maintaining high accuracy, we performed feature selection using extreme gradient boosting (XGBoost) and dimensionality reduction using principal component analysis (Khaire & Dhanalakshmi, 2022). Initially, we used GridSearchCV with repeated stratified k-fold as the cross-validation technique to tune the hyperparameters of an XGBoost model, as detailed in Table 1. The GridSearch, which involved 216 iterations, identified the following optimal hyperparameters for the XGBoost model: 300 estimators (n_estimators), max depth of 6, learning rate of 0.1, and regularization parameters (reg_alpha and reg_lambda) both set to 0.1.

The XGBoost model was subsequently trained on the portion of the N-BaIoT dataset pertaining to the doorbell IoT device. Instead of using the conventional train-test split to train the model, repeated stratified k-fold cross-validation with the same setting used in the GridSearch was incorporated in the XGBoost training to deal with imbalanced classes in the dataset and enhance training efficiency (Jian et al., 2021). Then, the top twenty-five features of the dataset were selected based on feature importance to form a compact but informative data. Table 2 presents the twenty-five features selected from the 115 features in the N-BaIoT dataset.

These twenty-five features were transformed into eighteen principal components using principal component analysis (PCA). PCA is an unsupervised learning technique that transforms correlated features into uncorrelated principal components, making it effective for dimensionality reduction while maintaining essential information in the original data (Anowar et al., 2021). First, it standardizes the initial IoT botnet data and computes its covariance matrix. Next, it determines the eigenvectors and eigenvalues of the covariance matrix. The directions of maximum variance are represented by the top eigenvectors, which correspond to the largest eigenvalues. In this case, the original twenty-five features were projected onto the top eighteen eigenvectors, which served as the principal components. This reduced the dimensionality of the twenty-five Mirai botnet features while retaining most of the essential information in the data (Reddy et al., 2020).

Following the dimensionality reduction process, the six classes of Mirai botnet attacks within the dataset were subjected to one-hot encoding prior to training of the proposed model.

2.7. Proposed federated learning-based MLP training

The training of a machine learning model in a federated learning system follows several key steps. Each client in the federated learning system trains the model on its local data. Once local training is complete, clients transmit the updated parameters of their trained models to a central server. The server aggregates them to produce updated global model parameters, which are then redistributed to all clients for enhanced intrusion detection (Campos et al., 2022).

The training of the proposed federated learning model for IoT botnet attack detection adheres to the steps outlined above. Initially, the dataset was partitioned into a training set that constitutes 80 % of the data and a testing set that constitutes the remaining 20 %. The training set was then divided into four chunks and assigned to four clients. One-hot encoding was then used to encode the classes in the dataset, which comprised five Mirai attack classes and a benign traffic class (Meidan et al., 2018). The model used for botnet attack detection was a multi-layer perceptron (MLP). The MLP architecture employed in this paper consist of four hidden layers, each containing thirty-two neurons and using the rectified linear unit (ReLU) activation function for feature extraction. Batch normalization was applied to normalize the activations in these layers, and a dropout rate of 0.2 was used to prevent overfitting. Finally, an output layer with a softmax activation function was employed for multiclass classification. The structure of the MLP is shown in Fig. 2, and its training parameters are provided in Table 3.

Table 1
Parameters of the Grid Search + Repeated Stratified K-fold CV.

Hyperparameter	Values
max_depth	6, 10
n_estimators	300, 500, 800
learning_rate	0.1, 0.01, 0.05
reg_alpha	0.1, 0.5
reg_lambda	0.1, 0.5
estimator	XGBoost Classifier
scoring	Accuracy
cv = repeated stratified k-fold cv	3 splits and 3 repeats

The classes in the local datasets of the clients in a federated learning system are usually not balanced. Imbalanced data reduces the accuracy of machine learning models by biasing the training toward the majority class (Tyagi & Mittal, 2019). To address this issue, during the MLP training, higher weights were assigned to the minority classes and lower weights to majority classes, mitigating the negative effect of class imbalance during the model's training (Choi et al., 2022).

The following outlines the step-by-step process for integrating the MLP into the federated learning-based intrusion detection system. First, an initial MLP model was assigned to each of the four clients in the FL system. The model was then trained on the local botnet data of each client for twenty epochs using the categorical cross-entropy loss function and the adaptive moment (Adam) optimizer.

At each epoch of local training, particularly during backpropagation, the gradients of the loss function with respect to the parameters (weights) of the MLP model were computed over a mini-batch and clipped to a predefined norm to prevent any single gradient from excessively influencing local model updates. Gaussian noise was added to the clipped gradients through the differentially private Adam (DP-Adam) optimizer to ensure the privacy of the IoT botnet training data. The noise magnitude was scaled by the clipping value to regulate the sensitivity of the gradients. Through DP-Adam, each FL client updated the parameters of its respective MLP model by utilizing moving averages of its differentially private gradients, as well as moving averages of its squared differentially private gradients (Tang et al., 2024). The differentially private gradient $\tilde{g}_i$ is given as:

$$\tilde{g}_i = \bar{g}_i + \left(\frac{1}{B}\right) z_i, \quad z_i \sim N\left(0, \sigma^2 C^2 I_d\right) \tag{3}$$

where $\bar{g}_i$ is the mean of the clipped gradient across the mini-batch B, while z_i represents noise that follows a Gaussian distribution, 0 indicates that the distribution is centered around zero, σ is the noise multiplier, C denotes the highest value for L2 norm clipping, and I_d refers to an identity matrix of dimension d. In implementing the DP-Adam, B was set to 1, σ was set to 0.01, C was set to 1.6, and learning rate (η) was set to 0.001.

After each local training epoch or iteration, i, the weights of the MLP model for each client, k, were computed as follows:

$$w_{i+1}^k = w_i^k - \eta \tilde{g}_i^k \tag{4}$$

At the end of the twenty training epochs, each client transmitted its final updated MLP weights for the current federated learning round, t, to a central server, where they were aggregated to form new global MLP weights. The updated MLP weights for each communication round of federated learning were calculated as follows:

$$w_t^k = w_{i+1}^k \tag{5}$$

Federated Averaging (FedAvg) was the aggregation algorithm used to combine the MLP parameters from each client. It computes new parameters for the global MLP using a weighted average, where the

Table 2
Top twenty-five features selected using XGBoost + Repeated Stratified K-fold CV.

Rank	Feature Name	Rank	Feature Name	Rank	Feature Name
1	HH_L1_weight	10	H_L5_weight	19	MI_dir_L5_mean
2	HH_jit_L1_weight	11	H_L0.1_variance	20	HH_jit_L5_mean
3	MI_dir_L0.01_mean	12	MI_dir_L0.1_weight	21	HH_jit_L3_mean
4	H_L0.1_weight	13	MI_dir_L0.1_mean	22	H_L3_mean
5	H_L0.1_mean	14	MI_dir_L5_weight	23	HH_L3_magnitude
6	MI_dir_L1_mean	15	H_L0.01_mean	24	MI_dir_L3_weight
7	H_L1_mean	16	H_L5_mean	25	HpHp_L0.01_weight
8	H_L0.01_variance	17	MI_dir_L0.1_variance		
9	MI_dir_L0.01_variance	18	MI_dir_L3_mean		

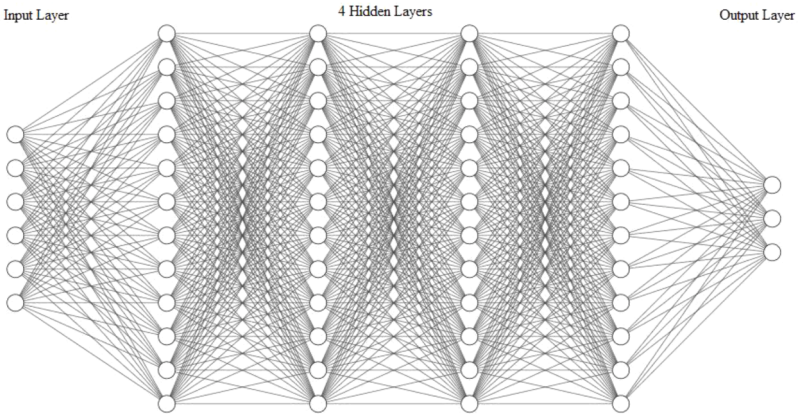


Fig. 2. The architecture of a Multi-Layer Perceptron.

Table 3
Training Parameters of the Proposed FL Model.

Parameter	Value
Local Model	MLP
Hidden Layers	4
Neurons per Hidden Layer	32
Local Training Epochs	20
Batch Size	64
Optimizer	DP-Adam
Hidden Layer Activation Function	ReLU
Output Layer Activation Function	Softmax
Learning Rate	0.001
Loss Function	Categorical Cross-Entropy
Number of FL Clients	4
FL Communication Rounds	5

influence of each client's updated MLP parameters is proportional to the size of its corresponding IoT botnet data on which the MLP was trained (Qi et al., 2024). FedAvg can be expressed mathematically as:

$$w_{t+1}^{r} = \sum_{k=1}^{K} \left(\frac{n^k}{n}\right) w_{t+1}^{k} \tag{6}$$

where k represents any one of the four clients in the FL process, n^k represents the size of the IoT botnet data on which client k trains the MLP, n represents the combined size of the botnet data across all the four clients, w_{t+1}^{k} represents the updated weights of the MLP transmitted to the aggregation server by client k at FL communication round $t + 1$. Thus, for each round of federated MLP training, the global MLP parameters w_{t+1}^{r} were computed by averaging the updated local model parameters of all clients, weighted by the amount of data each client trained the MLP on. This ensures that the MLP parameters of clients with more data have a greater influence on the global model than clients with fewer data samples, leading to a more accurate global model. After this first federated learning round, the updated global MLP parameters were transmitted back to all four clients to begin another round of federated learning.

We trained the MLP in this collaborative manner over five communication rounds of federated learning, after which the model converged.

2.8. Performance metrics

The following metrics were used to assess the performance of the proposed FL model: accuracy, precision, recall, f1 measure, and area under the curve. These metrics are based on the true positive (TP) value, false positive (FP), true negative (TN), and false negative (FN) values of the model when predicting the classes in the dataset (Regan et al., 2022).

$$\text{Accuracy} = \left(\frac{TP + TN}{TP + TN + FP + FN}\right)100 \tag{7}$$

$$\text{Precision} = \left(\frac{TP}{TP + FP}\right)100 \tag{8}$$

$$\text{Recall} = \left(\frac{TP}{TP + FN}\right)100 \tag{9}$$

$$\text{F1 Score} = \left(2 \times \frac{Precision \times Recall}{Precision \times Recall}\right)100 \tag{10}$$

3. Results and discussion

3.1. FL model performance analysis

Experiments were conducted with three deep neural network models to find a balance between differential privacy, low computational complexity, and high accuracy in federated learning-based IoT botnet attack detection. The first experiment was conducted with a convolutional neural network (CNN) and the second was with a hybrid convolutional neural network and gated recurrent unit (CNN-GRU). Finally,

the proposed multi-layer perceptron (MLP) was also tested. With the parameter values of DP-Adam kept constant for all experiments, the proposed MLP-based FL stood out as the best model for detecting the Mirai botnet attacks with an accuracy of 99.93 % as shown in Fig. 3. This accuracy was higher than that of the CNN and the CNN-GRU, which achieved 99.35 % and 98.57 %, respectively.

Additionally, the MLP-based FL achieved the lowest computational complexity, 8,612 FLOPs, in contrast to the 25,900 FLOPs and 32,100 FLOPs achieved by the CNN-GRU and the CNN, respectively, as shown in Fig. 4. The proposed MLP model had fewer trainable parameters, 4,166, than the CNN-GRU model but slightly higher than CNN.

Fig. 5 details the performance of the proposed FL model using the confusion matrix. Our proposed model achieved a true positive value of approximately 100 % for all six IoT network traffic classes, that is, normal (benign) class, Mirai Ack flooding class, Mirai Scan class, Mirai Syn flooding class, Mirai UDP flooding class and Mirai UDP plain flooding class. It also had a false positive value of approximately 0 % for most classes in the dataset. The high true positive values show that the proposed model is highly reliable for intrusion detection (Sarhan et al., 2022), while the low false positive values indicate that the model rarely produces false intrusion alarms. Additionally, the low false negative values indicate that almost no attack went undetected (Kidmose et al., 2020). Overall, the results demonstrate a strong multiclass classification performance of the proposed model, which is further validated by its approximately 100 % precision, recall, and F1-score across all six classes, as shown in Table 4.

Fig. 6 illustrates the proposed model's exceptional performance in terms of the Receiver Operating Characteristic Area Under the Curve (ROC AUC), which is a critical metric for assessing the efficacy of classification models. The AUC value shows the model's ability to distinguish between positive and negative classifications, where a perfect classification is denoted by an AUC value of 1.00. In this ideal scenario, the model achieves a low false positive rate alongside a high true positive rate, indicating its proficiency in correctly distinguishing classes in the data (Zuech et al., 2021). The proposed FL model achieved an impressive AUC value of 1.00 for all classes in the dataset, resulting in a weighted average of 1.00. This remarkable performance underscores the model's capability to effectively differentiate between Mirai botnet attack traffic and normal IoT network traffic.

Table 5 compares the model's overall performance with existing privacy-compromising centralized machine learning models in terms of accuracy, F1 score, and area under the curve. In addition to preserving data privacy, the proposed federated learning model outperforms most of the existing centralized botnet detection models. For example, it surpasses the Artificial Neural Network (ANN) model proposed by Palla and Tayeb (2021) for Mirai botnet detection with 7.13 % higher accuracy. Similarly, it performs better than the LSTM-XGBoost model

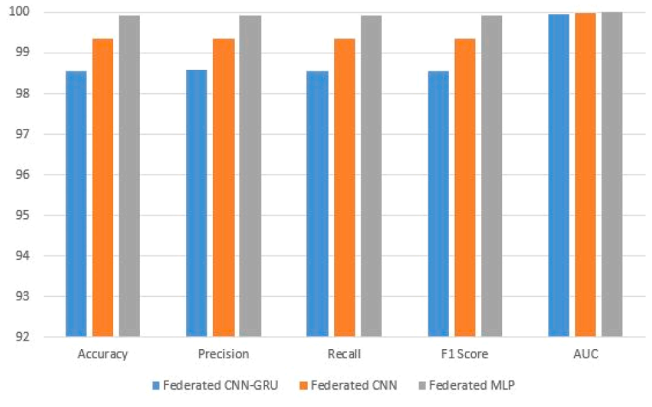


Fig. 3. Performance of the proposed model compared to other models during experiment.

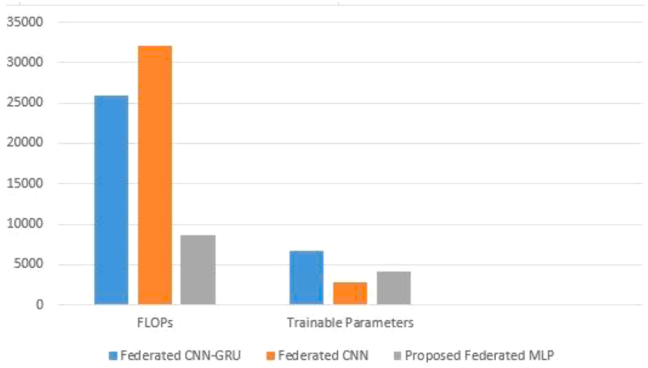


Fig. 4. Computational complexity of the proposed model and other experimental models.

suggested by Vajrobol et al. (2024) with 2.23 % higher accuracy. It also outperforms the model proposed by Fan et al. (2024) with a 21.93 % higher F1 score for Mirai botnet detection. Additionally, our model's performance is validated by several metrics, unlike some centralized models, which were evaluated only on accuracy (De La Torre Parra et al., 2020; Sharma et al., 2023; Duan et al., 2024)

3.2. Computational complexity analysis

The number of floating point operations performed by the deep federated learning model is used to evaluate the model's computational complexity (Akter et al., 2021; Zainudin et al., 2023; Liang et al., 2023).

Table 6 shows the effectiveness of the XGBoost-based feature selection and PCA-based dimensionality reduction techniques in minimizing computational complexity. When all the 115 features in the dataset was used for training without feature selection and dimensionality reduction, the model achieved an accuracy of 99.75 % with a computational complexity of 14,820 FLOPs. When XGBoost was used to select the top eighteen important features, the computational complexity reduced to 8,612 but the accuracy also reduced to 99.48 %. However, when the top twenty-five features were selected based on XGBoost's feature importance and PCA was used to transform them to eighteen principal components, not only did the computational complexity reduce to 8,612 but the accuracy also increased to 99.93 %. This is because the feature selection eliminates the noisy features from the training data while the PCA transforms those twenty-five selected features to eighteen eigenvectors (principal components) to create a lower-dimensional representation, which reduces complexity and still retains most of the essential information (Reddy et al., 2020). The computational complexity of the MLP is partly determined by its number of input neurons which is equivalent to the number of features of the input data. Hence, a reduction from twenty-five features to eighteen principal components with most of the important information maintained led to a reduction in the MLP's computational complexity without compromising intrusion detection accuracy.

The limited computational capacity and memory of IoT edge devices make it challenging to implement high-complexity deep learning models. However, like other low-complexity models—such as the lightweight MobileNetV3 proposed by Sun et al. (2022), which has 0.26 MB of parameters, requires 5.26 million FLOPs, and has been tested on a Raspberry Pi 4 Model B—our proposed model, with only 0.01702 MB of parameters and 8,612 FLOPs, is well-suited for IoT edge deployment, achieving a significantly lower parameter count and computational cost compared to the lightweight MobileNetV3.

3.3. Comparison with state-of-the-art FL-Based intrusion detection models

As shown in Table 7, most existing FL-based intrusion detection

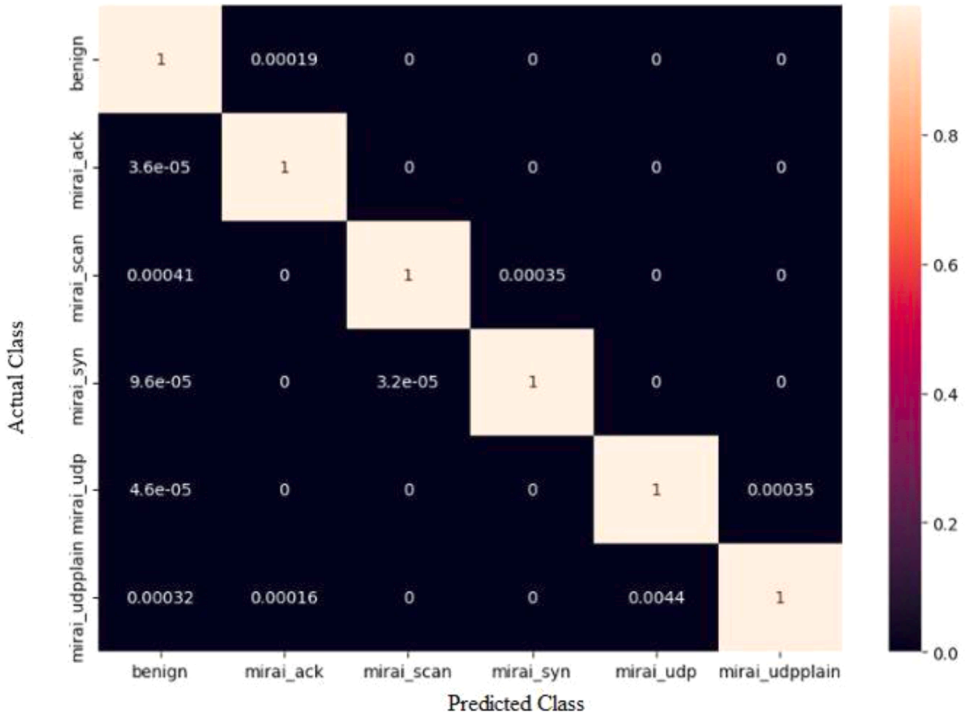


Fig. 5. Confusion Matrix of the Proposed Model.

Table 4
Multiclass Classification Performance of the Proposed MLP-based FL Model.

Class	Precision	Recall	F1 Score
Benign	100 %	100 %	100 %
Mirai Ack	100 %	100 %	100 %
Mirai Scan	100 %	100 %	100 %
Mirai Syn	100 %	100 %	100 %
Mirai UDP	100 %	100 %	100 %
Mirai UDP Plain	100 %	100 %	100 %

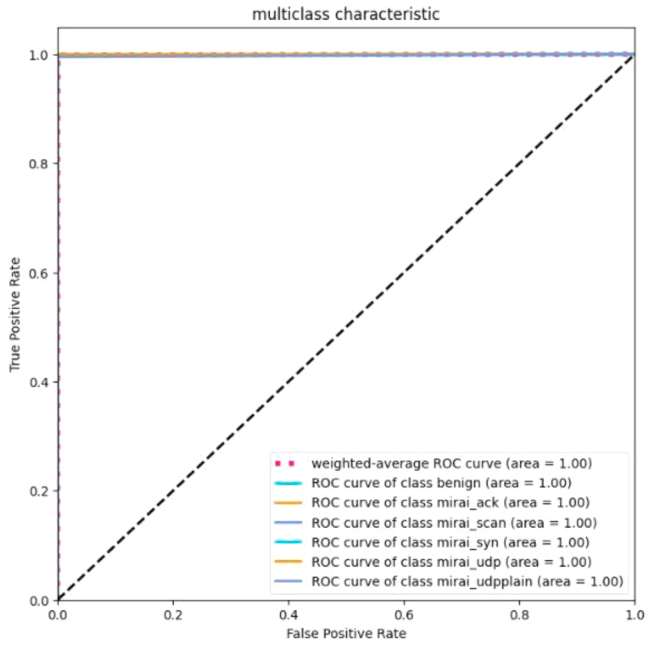


Fig. 6. ROC Area Under the Curve (AUC) for the Multiclass IoT Botnet Attack Detection.

Table 5
Comparison of the proposed FL-based IDS with existing Centralized ML models.

Model	Dataset	Accuracy	F1 Score	AUC
LSTM (De La Torre Parra et al., 2020)	N-BaIoT	94.8 %	✘	✘
ANN (Palla & Tayeb, 2021)	N-BaIoT	92.8 %	99 %	0.92
DNN (Kunang et al., 2021)	NSL-KDD	83.33 %	82.04 %	✘
1D-CNN (Kundu et al., 2024)	Kitsune	97.9 %	97.72 %	0.999
GAN-DNN (Sharma et al., 2023)	UNSW-NB15	91 %	✘	✘
Bi-GRU, RNN (Bojarajulu et al., 2023)	IoT-Botnet	97 %	82.4 %	✘
Transformer (Wang et al., 2023)	ToN-IoT	97.06 %	96.94 %	✘
XGBoost (Fan et al., 2024)	IoTID20	75 %	78 %	✘
LSTM-XGBoost (Vajrobol et al., 2024)	CICIoT2023	97.7 %	97.4 %	✘
MRENN (Duan et al., 2024)	N-BaIoT	95 %	✘	✘
Proposed MLP-based FL	**N-BaIoT**	**99.93 %**	**99.93 %**	**1.0**

Table 6
Proposed FL Model with and without XGBoost Feature Selection + PCA.

Model	FLOPs	Accuracy
Proposed FL without FS and PCA	14,820	99.75 %
Proposed FL with XGBoost FS only	8,612	99.48 %
Proposed FL with XGBoost FS + PCA	**8,612**	**99.93 %**

models proposed for deployment on IoT edge devices have not had their computational complexity examined, raising questions about their feasibility in IoT applications. By contrast, our model not only underwent such evaluation but also proved to be more computationally efficient than the few models whose complexity was assessed. Particularly, the 4,166 trainable parameters of our model is more efficient than the 87,432 trainable parameters which characterized the model suggested by Abdel-Basset et al. (2023) and the 7,410 trainable parameters of the model proposed by Zainudin et al. (2023). Our model also achieved an

Table 7
Comparison of the proposed FL model with state-of-the-art FL-based IDS models.

Model	Dataset	Accuracy	Precision	Recall	F1 Score	AUC	FLOPs
Denoising AE (Tian et al., 2021)	IoT-23	89.5 %	96.82 %	84.45 %	90.15 %	✘	✘
SVM (Moulahi et al., 2022)	VeReMi	82.45 %	91.02 %	< 50 %	< 50 %	✘	✘
Autoencoder (Regan et al., 2022)	N-BaIoT	98 %	✘	✘	✘	✘	✘
CNN (Abdel-Basset et al., 2023)	Virus MNIST	94 %	✘	✘	✘	✘	N/A
DNN (Sarhan et al., 2022)	BoT-IoT	93.08 %	✘	91.92 %	93.01 %	0.9595	✘
1D-CNN (de Caldas Filho et al., 2023)	Private Dataset	89.753 %	✘	✘	✘	✘	✘
CNN, RNN (Rashid et al., 2023)	Edge-IIoT	92.49 %	✘	✘	✘	✘	✘
FNN (Cholakoska et al., 2023)	IoTID-20	98.3 %	✘	✘	✘	✘	✘
CNN-MLP (Zainudin et al., 2023)	Edge-IIoT	95.41 %	✘	✘	✘	0.9723	68,000
CNN (Jia et al., 2024)	NSL-KDD	97.8 %	✘	✘	✘	✘	✘
MLP (Han et al., 2024)	UNSW-NB15	84.67 %	89.03 %	83.69 %	78.96 %	0.8305	✘
Deep AE (Salim et al., 2024)	BoT-IoT	92 %	92 %	89 %	90 %	✘	✘
Proposed (MLP)	**N-BaIoT**	**99.93 %**	**99.93 %**	**99.93 %**	**99.93 %**	**1.0**	**8,612**

extremely lower number of floating-point operations, 8,612, compared to the 68,000 FLOPs of the model suggested by Zainudin et al. (2023). Our model, therefore, reduced computational complexity in terms of FLOPs by a margin of approximately 87.34 %.

Table 7 further compares the intrusion detection performance of the proposed FL model with state-of-the-art FL models. Our model performs better than the related works by achieving 99.93 %. The performance of some related works (Jia et al., 2024; Cholakoska et al., 2023; Regan et al., 2022) was examined solely based on accuracy. However, our model was examined on other metrics as well to give holistic information regarding its performance. For the related works evaluated on precision, recall, and f1 score, our model still performs better than them, achieving 99.93 % for all three metrics. Our proposed model also performs better than the related works by achieving a perfect area under the curve value of 1.00. Unlike some related works that performs highly on one metric but poorly on the others (Tian et al., 2021; Moulahi et al., 2022; Salim et al., 2024), our model performs highly on all the metrics.

4. Conclusion

Federated learning is a privacy-preserving framework that enables the collaborative training of a machine learning model across devices in different networks by aggregating their parameters or weights into a global model without sharing local data. This paper introduced a federated learning system based on a Multi-Layer Perceptron with low computational complexity to detect IoT botnet attacks. The proposed approach employed an XGBoost model trained with repeated stratified k-fold cross-validation to select optimal features for IoT botnet attack detection, followed by Principal Component Analysis for dimensionality reduction, effectively reducing computational complexity. Differential privacy was incorporated into the MLP training to enhance data privacy during the model aggregation phase of the federated learning system. The proposed approach achieved the lowest computational complexity of 8,612 FLOPs and the highest accuracy of 99.93 % compared to related works, demonstrating its capability to be deployed on IoT edge devices with low computational capacity. These results also validate the potential of the proposed federated learning model in improving security while maintaining data privacy in smart homes, smart healthcare, smart grids, and other settings where resource-constrained IoT edge devices are deployed.

In future research, we aim to further reduce the model's complexity by employing quantization and pruning techniques, improve its energy efficiency, enhance its robustness against Byzantine attacks from malicious clients, and improve its communication efficiency during model aggregation.

CRediT authorship contribution statement

Lambert Kofi Gyan Danquah: Conceptualization, Methodology, Software, Visualization, Writing – original draft. **Stanley Yaw Appiah:** Writing – review & editing. **Victoria Adzovi Mantey:** Validation. **Iddrisu Danlard:** Writing – review & editing. **Emmanuel Kofi Akowuah:** Supervision.

Declaration of competing interest

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

Data availability

Data will be made available on request.

References

Abdel-Basset, M., et al. (2023). Efficient and lightweight convolutional networks for IoT malware detection: A federated learning approach. *IEEE Internet of Things Journal, 10*, 7164–7173. April, Volume.

Adeel, A., et al. (2018). A survey on the role of wireless sensor networks and IoT in disaster management. *Geological disaster monitoring based on sensor networks* (pp. 57–66). s.l.:Springer Singapore.

Agrawal, S., et al. (2022). Federated Learning for intrusion detection system: Concepts, challenges and future directions. *Computer Communications, 195*, 346–361. November, Volume.

Akter, R., Doan, V.-S., Huynh-The, T., & Kim, D.-S. (2021). RFDOA-Net: An efficient ConvNet for RF-based DOA estimation in UAV surveillance systems. *IEEE Transactions on Vehicular Technology, 70*, 12209–12214. November, Volume.

Anowar, F., Sadaoui, S., & Selim, B. (2021). Conceptual and empirical comparison of dimensionality reduction algorithms (PCA, KPCA, LDA, MDS, SVD, LLE, ISOMAP, LE, ICA, t-SNE). *Computer Science Review, 40*, Article 100378. May, Volume.

Bhushan, B., et al. (2023). Towards a secure and sustainable Internet of Medical Things (IoMT): Requirements, design challenges, security techniques, and future trends. *Sustainability, 15*, 6177. April, Volume.

Bojarajulu, B., Tanwar, S., & Singh, T. P. (2023). Intelligent IoT-BOTNET attack detection model with optimized hybrid classification model. *Computers & Security, 126*, Article 103064. March, Volume.

Campos, E. M., et al. (2022). Evaluating Federated Learning for intrusion detection in Internet of Things: Review and challenges. *Computer Networks, 203*, Article 108661. February, Volume.

Canavese, D., Mannella, L., Regano, L., & Basile, C. (2024). Security at the edge for resource-limited IoT devices. *Sensors, 24*, 590. January, Volume.

Choi, H.-S., Jung, D., Kim, S., & Yoon, S. (2022). Imbalanced data classification via cooperative interaction between classifier and generator. *IEEE Transactions on Neural Networks and Learning Systems, 33*, 3343–3356. August, Volume.

Cholakoska, A., et al. (2023). Federated learning for network intrusion detection in ambient assisted living environments. *IEEE Internet Computing, 27*, 15–22. July, Volume.

de Caldas Filho, F. L., et al. (2023). Botnet Detection and Mitigation Model for IoT Networks Using Federated Learning. *Sensors, 23*, 6305. July, Volume.

De La Torre Parra, G., Rad, P., Choo, K.-K. R., & Beebe, N. (2020). Detecting Internet of Things attacks using distributed deep learning. *Journal of Network and Computer Applications, 163*, Article 102662. August, Volume.

Devi, D. H., et al. (2023). 5G technology in healthcare and wearable devices: A review. *Sensors, 23*, 2519. February, Volume.

Duan, Y., Wu, Q., Zhao, X., & Li, X. (2024). Mobile edge computing based cognitive network security analysis using multi agent machine learning techniques in B5G. *Computers and Electrical Engineering, 117*, Article 109181. July, Volume.

Fan, Z., Sohail, S., Sabrina, F., & Gu, X. (2024). Sampling-based machine learning models for intrusion detection in imbalanced dataset. *Electronics, 13*, 1878. May, Volume.

Han, W., et al. (2024). Heterogeneous data-aware federated learning for intrusion detection systems via meta-sampling in artificial intelligence of things. *IEEE Internet of Things Journal, 11*, 13340–13354. April, Volume.
Houda, Z. A. E., Hafid, A. S., & Khoukhi, L. (2023). MiTFed: A privacy preserving collaborative network attack mitigation framework based on federated learning using SDN and blockchain. *IEEE Transactions on Network Science and Engineering, 10*, 1985–2001. July, Volume.
Hussain, F., et al. (2021). A two-fold machine learning approach to prevent and detect IoT botnet attacks. *IEEE Access : Practical Innovations, Open Solutions, 9*, 163412–163430. Volume.
Imteaj, A., & Amini, M. H. (2022). Leveraging asynchronous federated learning to predict customers financial distress. *Intelligent Systems with Applications, 14*, Article 200064. May, Volume.
Jian, Y., Pasquier, M., Sagahyroon, A., & Aloul, F. (2021). A machine learning approach to predicting diabetes complications. *Healthcare, 9*, 1712. December, Volume.
Jia, Y., Lin, F., & Sun, Y. (2024). A novel federated learning aggregation algorithm for AIoT intrusion detection. *IET Communications, 18*, 429–436. March, Volume.
Khaire, U. M., & Dhanalakshmi, R. (2022). Stability of feature selection algorithm: A review. *Journal of King Saud University - Computer and Information Sciences, 34*(4), 1060–1073.
Kidmose, E., Stevanovic, M., Brandbyge, S., & Pedersen, J. M. (2020). Featureless discovery of correlated and false intrusion alerts. *IEEE Access : Practical Innovations, Open Solutions, 8*, 108748–108765. Volume.
Kolias, C., Kambourakis, G., Stavrou, A., & Voas, J. (2017). DDoS in the IoT: Mirai and other botnets. *Computer, 50*, 80–84. Volume.
Koroniotis, N., Moustafa, N., Sitnikova, E., & Turnbull, B. (2019). Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Generation Computer Systems, 100*, 779–796. November, Volume.
Kumar, A., Shridhar, M., Swaminathan, S., & Lim, T. J. (2022). Machine learning-based early detection of IoT botnets using network-edge traffic. *Computers & Security, 117*, Article 102693. June, Volume.
Kunang, Y. N., Nurmaini, S., Stiawan, D., & Suprapto, B. Y. (2021). Attack classification of an intrusion detection system using deep learning and hyperparameter optimization. *Journal of Information Security and Applications, 58*, Article 102804. May, Volume.
Kundu, P. P., et al. (2024). Detection and classification of botnet traffic using deep learning with model explanation. *IEEE Transactions on Dependable and Secure Computing*, 1–15.
Liang, Y., Li, M., Jiang, C., & Liu, G. (2023). CEModule: A computation efficient module for lightweight convolutional neural networks. *IEEE Transactions on Neural Networks and Learning Systems, 34*, 6069–6080. September, Volume.
Lombardi, M., Pascale, F., & Santaniello, D. (2021). Internet of Things: A general overview between architectures, protocols and applications. *Information, 12*, 87. February, Volume.
Meidan, Y., et al. (2018). N-BaIoT—Network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Computing, 17*, 12–22. July, Volume.
Moulahi, T., et al. (2022). Privacy-preserving federated learning cyber-threat detection for intelligent transport systems with blockchain-based security. *Expert Systems, 40*. July.Volume.
Nuaimi, M., Fourati, L. C., & Hamed, B. B. (2023). Intelligent approaches toward intrusion detection systems for Industrial Internet of Things: A systematic comprehensive review. *Journal of Network and Computer Applications, 215*, Article 103637. June, Volume.
Palla, T. G., & Tayeb, S. (2021). Intelligent Mirai malware detection for IoT nodes. *Electronics, 10*, 1241. May, Volume.
Popoola, S. I., et al. (2021). Stacked recurrent neural network for botnet detection in smart homes. *Computers & Electrical Engineering, 92*, Article 107039. June, Volume.
Pratap Singh, R., et al. (2020). Internet of Medical Things (IoMT) for orthopaedic in COVID-19 pandemic: Roles, challenges, and applications. *Journal of Clinical Orthopaedics and Trauma, 11*, 713–717. July, Volume.
Prazeres, N., Costa, R. L.d. C., Santos, L., & Rabadão, C. (2023). Engineering the application of machine learning in an IDS based on IoT traffic flow. *Intelligent Systems with Applications, 17*, Article 200189. February, Volume.
Qi, P., et al. (2024). Model aggregation techniques in federated learning: A comprehensive survey. *Future Generation Computer Systems, 150*, 272–293. January, Volume.
Rashid, M. M., et al. (2023). A federated learning-based approach for improving intrusion detection in industrial Internet of Things networks. *Network (Bristol, England), 3*, 158–179. January, Volume.
Reddy, G. T., et al. (2020). Analysis of dimensionality reduction techniques on big data. *IEEE Access : Practical Innovations, Open Solutions, 8*, 54776–54788. Volume.
Regan, C., et al. (2022). Federated IoT attack detection using decentralized edge data. *Machine Learning with Applications, 8*, Article 100263. June, Volume.
Saharkhizan, M., et al. (2020). An ensemble of deep recurrent neural networks for detecting IoT cyber attacks using network traffic. *IEEE Internet of Things Journal, 7*, 8852–8859. September, Volume.
Salim, S., et al. (2024). Deep-federated-learning-based threat detection model for extreme satellite communications. *IEEE Internet of Things Journal, 11*, 3853–3867. February, Volume.
Sarhan, M., Layeghy, S., Moustafa, N., & Portmann, M. (2022). Cyber threat intelligence sharing scheme based on federated learning for network intrusion detection. *Journal of Network and Systems Management, 31*. October.Volume.
Sharma, B., Sharma, L., Lal, C., & Roy, S. (2023). Anomaly based network intrusion detection for IoT attacks using deep learning technique. *Computers and Electrical Engineering, 107*, Article 108626. April, Volume.
Sofana Reka, S., Dragičević, T., Siano, P., & Prabaharan, S. R. S. (2019). Future generation 5G wireless networks for smart grid: A comprehensive review. *Energies, 12*, 2140. June, Volume.
Sun, C., et al. (2022). Lightweight traffic classification model based on deep learning. *Wireless Communications and Mobile Computing, 2022*, 1–16. October, Volume.
Tang, Q., Shpilevskiy, F., & Lécuyer, M. (2024). DP-AdamBC: Your DP-Adam is actually DP-SGD (unless you apply bias correction). *Proceedings of the AAAI Conference on Artificial Intelligence, 38*, 15276–15283. March, Volume.
Tian, P., Chen, Z., Yu, W., & Liao, W. (2021). Towards asynchronous federated learning based threat detection: A DC-Adam approach. *Computers & Security, 108*, Article 102344. September, Volume.
Tyagi, S., & Mittal, S. (2019). Sampling approaches for imbalanced data classification problem in machine learning. In *Proceedings of ICRIC 2019* (pp. 209–221). s.l.: Springer International Publishing.
Vajrobol, V., Gupta, B. B., Gaurav, A., & Chuang, H.-M. (2024). Adversarial learning for Mirai botnet detection based on long short-term memory and XGBoost. *International Journal of Cognitive Computing in Engineering, 5*, 153–160. Volume.
Vignau, B., Khoury, R., Hallé, S., & Hamou-Lhadj, A. (2021). The evolution of IoT malwares, from 2008 to 2019: Survey, taxonomy, process simulator and perspectives. *Journal of Systems Architecture, 116*, Article 102143. June, Volume.
Wang, M., Yang, N., & Weng, N. (2023). Securing a smart home with a transformer-based IoT intrusion detection system. *Electronics, 12*, 2100. May, Volume.
Whitmore, A., Agarwal, A., & Da Xu, L. (2015). The Internet of Things—A survey of topics and trends. *Information Systems Frontiers, 17*, 261–274. April, Volume.
Zainudin, A., Akter, R., Kim, D.-S., & Lee, J.-M. (2023). Federated learning inspired low-complexity intrusion detection and classification technique for SDN-based industrial CPS. *IEEE Transactions on Network and Service Management, 20*, 2442–2459. September, Volume.
Zuech, R., Hancock, J., & Khoshgoftaar, T. M. (2021). Detecting web attacks using random undersampling and ensemble learners. *Journal of Big Data, 8*. May.Volume.