Journal of Artificial Intelligence and Technology, (Ahead of Print)
https://doi.org/10.37965/jait.2025.0798

ISTP
RESEARCH ARTICLE

Federated Deep Learning-Based Security Model Against DDoS Attack for Cloud Environment

Jyoti Tolanur[1] and Shilpa Chaudhari[2]
[1]Department of Computer Science and Engineering, REVA University, Bangalore, India
[2]Department of Computer Science and Engineering, Ramaiah Institute of Technology (Affiliated to VTU), Bangalore, India

(Received 26 April 2025; Revised 18 July 2025; Accepted 20 August 2025; Published online 13 September 2025)

Abstract: Technological advancements on the internet have joined the realms of space, air, land, and sea warfare with a substantial impact on both personal and national security. Distributed denial of service (DDoS) attacks cause a denial of service for legitimate users by flooding a targeted network or service with large volumes of traffic from multiple hacked systems. This traffic dataset falls under the Bigdata category as DDoS attacks can generate a large amount of traffic in a brief period. Cloud-based networks and services are under threat from these attacks. Traditional machine learning-based DDoS attack detection lacks scalability, latency due to huge, centralized data requirements, and corresponding security concerns. Federated learning (FL)-based security model addresses these issues without sharing raw traffic data required for the training step. This paper proposes an FL-based security model for enhancing FL security in a cloud environment while preserving data privacy by default. The model focuses on trust assumption, FL protection mechanism, and trust boundaries using zero-trust principles. FL-based security model ensures CIA (confidentiality, integrity, and authentication) during the model training using CRYSTALS-Kyber post-quantum cryptographic algorithm, which is a lattice-based algorithm for key exchange. Even though it supports encryption, we use Advanced Encryption Standard (AES) for confidentiality. Simulation scenarios study the behavior of the system under different conditions to understand the effects of a DDoS attack on a cloud computing environment. The model achieves high DDoS detection accuracy with reduced communication overhead.

Keywords: cloud computing; deep learning; distributed denial of service attacks; federated learning; security model

I. INTRODUCTION

Internet-based cloud computing advancements provide easy access to computing power, storage, and other capabilities from anywhere. Security is a primary concern in cloud computing. One of the threats affecting the availability of resources is denial of service. Intruder consumes resources or processing power of legitimate users in a cloud computing pay-per-use paradigm, resulting in significant charges or denying them access altogether [1]. An attack known as a distributed denial of service (DDoS) intensifies its impact on the target computer by originating from multiple vulnerable devices.

The danger posed by DDoS attacks is dynamic and constantly evolving [2]. Some of the wide range of systems and services that are susceptible to DDoS attacks includes websites making it inaccessible to user by flooding the server with traffic, network architecture where routers or switches disrupt communication and connectivity, DNS servers preventing users from accessing websites through, servers for online gaming preventing or disturbing players, cloud services disrupting their the availability, Internet of Things (IoT) gadgets causing them to malfunction or become unavailable, application servers preventing user access, email server disturbing the availability, Voice over Internet Protocol (VoIP) connectivity disrupting phone communication, and edge networks overwhelming a variety of resources. Table I illustrates multiple categories of DDoS attacks that utilize diverse techniques to accomplish the objective [3].

DDoS attacks can have several impacts on cloud computing [4]. The attacker will attempt to perform any one or a combination of the following techniques to launch an attack. (1) *Overloading resources:* sending a large volume of traffic to overload the resources of the target, making it unable to handle legitimate requests. (2) *Disrupting service* to revenue loss and customer satisfaction: success of overwhelming the resources of a cloud computing service makes them unavailable or degrades the performance for users. (3) *Spreading to other services*: spread DDoS attacks to other services within a cloud computing infrastructure, such as load balancers or database servers, to disrupt the overall functioning of the cloud computing environment. (4) *Exposing security vulnerabilities*: Usage of DDoS attacks covers other types of attacks, such as malware infections or data breaches. If a cloud computing provider is not adequately prepared to defend against DDoS attacks, it may be vulnerable to these types of secondary attacks. (5) Increasing costs: due to scaling up the infrastructure to handle the additional traffic and resources required to defend against a DDoS attack.

It is possible to initiate a DDoS attack without a botnet by using cloud-based DDoS attack services [12]. The detection of DDoS attacks in cloud computing settings has been approached from a variety of angles by different authors in the literature. The following are the techniques listed from the literature [5]. (1) Traffic analysis: analyzing the traffic patterns in a network helps to identify anomalies or deviations from standard traffic patterns that may indicate a DDoS attack. (2) Machine learning

Corresponding authors: Jyoti Tolanur (e-mail: jyoti.tolanur@gmail.com); Shilpa Chaudhari (e-mail: shilpasc29@msrit.edu).

Table I. Major types of DDoS attacks with details

Attack type	Description	Characteristics	Methods
Volume-based or network-centric	Overload a server or network's bandwidth.	A large number of packets are sent to the target.	UDP, ICMP floods.
Protocol	Leverage flaws in network protocols to create traffic jams and waste bandwidth.	Exploitation of protocol weaknesses.	SYN floods, DNS amplification attacks.
Application layer	Focus on specific services or applications.	Attack on the application layer of the OSI model.	HTTP floods, SQL injection attacks.
Fragmentation	Exceed a target's capacity to reassemble the streams by sending a deluge of TCP or UDP packets.	Transport and network layer attack	TCP and IP Fragmentation Attacks
Distributed Reflection Denial of Service (DRDoS)	Use a network of compromised devices (a botnet) to amplify the traffic sent to the targeted server.	Amplification of traffic using a botnet.	NTP amplification, DNS amplification.
Advanced Persistent DDoS (APT DDoS)	Persistent and long-lasting, they often include a combination of different attack types.	Sophisticated, persistent, and long-lasting.	A combination of different DDoS attack types.

(ML): Usage of ML algorithms for analyzing traffic patterns spots anomalous activity that can point to a DDoS attack. (3) Hash-based methods: generated hashes of network traffic are compared with the known good hashes to detect suspicious activity. (4) Signature-based methods: Signatures of known DDoS attack patterns are used to identify and block similar traffic. (5) Anomaly-based methods: identifying deviations from standard traffic patterns and behavior as potential indicators of a DDoS attack. (6) Reputation-based methods: maintain a list of known malicious IP addresses and block traffic from those sources.

There is no one technique that works everywhere. Hence, a combination of methods is needed to effectively detect and mitigate DDoS attacks in a cloud computing environment. Additionally, ML overcomes the drawbacks of conventional security techniques [25] and boosts network security by offering the instruments required to improve cloud-based DDoS attack detection, classification, and mitigation in a scalable, accurate, and efficient manner [6]. Commonly used datasets for analysis of DDoS attack patterns, and to develop DDoS attack detection and mitigation techniques using ML, include CICDDoS2019, DDoS-2019, and CAIDA DDoS Attack 2017 Dataset.

Traditional ML-based DDoS attack detection lacks scalability, is prone to latency due to huge, centralized data requirements, and raises corresponding security concerns. Traditional ML approach, such as Random Forest (RF), is a strong baseline for classification tasks, which may show competitive performance in centralized learning scenarios. Federated learning (FL)-based security model addresses these issues without sharing raw traffic data required for the training step [7]. A dual advantage of FL includes mitigating data privacy issues and reducing the overhead of data transmission [26]. Nodes are trained locally on their data in FL. The central server collects the model hyperparameters and aggregates them for the global model, which will be shared back with clients for further DDoS analysis [13]. Although FL provides data privacy, it operates in a decentralized and untrusted environment, motivating the researcher to develop a security model that assumes trust boundaries and protective mechanisms. Traditional ML and FL approaches [10] are shown in Fig. 1. The work in this paper uses an FL approach to address data privacy, scalability, and learning in cloud-based distributed environments. As FL enables collaborative model training without sharing sensitive data, it is used for decentralized data where directly sharing raw data is not feasible due to privacy regulations. FL-based DDoS detection for IoT is given in [27,28], and FL-based DDoS for software-defined networks is given in [29,30].

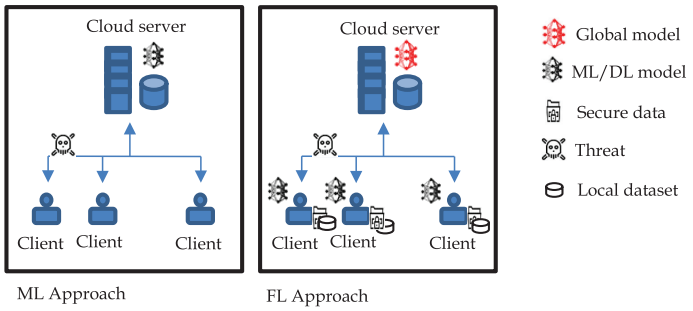

Fig. 1. Comparison of ML and FL approaches for DDoS detection.

This paper proposes an FL-based security model for enhancing FL security in a cloud environment while preserving data privacy by default. The model focuses on trust assumption, FL protection mechanism, and trust boundaries using zero-trust principles. FL-based security model ensures CIA (confidentiality, integrity, and authentication) during the model training using CRYSTALS-Kyber post-quantum cryptographic algorithm, which is a lattice-based algorithm for key exchange. Even though it supports encryption, we use the AES symmetric key algorithm for confidentiality to reduce the computation cost of data communication.

The significant contributions of the paper are as follows. (1) FL-based approach for cloud environment against DDoS attack. (2) Evaluation of the proposed FL method against contemporary solutions using the CICIDS2019 dataset using a Python environment. (3) Security model for the proposed FL method using CRYSTALS-Kyber post-quantum cryptographic algorithm, which is a lattice-based algorithm for key exchange, and AES for confidentiality. (4) Security analysis and performance analysis of the security model.

The paper is organized as follows. Section II presents the existing related work on FL-based DDoS analysis. Section III briefly describes the proposed FL-based security model. Sections IV and V present the security analysis and performance analysis of the model. Finally, Section VI concludes the work with a summary of the research findings.

II. BACKGROUND AND LITERATURE REVIEW

Federated Learning Approach to DDoS attack Detection (FLAD) [8] was a dynamic client selection and resource allocation approach for federated DDoS detection. Unlike conventional FedAvg, FLAD focused on clients with more challenging attack profiles, utilizing arithmetic mean aggregation to improve handling of non-IID data. It used an Multilayer Perceptron (MLP)-based adaptive FL. A superior F1 score (0.9667) over FedAvg and FLDDoS was achieved with significantly reduced convergence time (617 s vs. 12,205 s), while maintaining accuracy in the presence of new attack types. FLAD optimized FL for DDoS scenarios by addressing data heterogeneity, reducing training time, and preserving privacy.

The FL-based approach for IoT edge device training used a shared model locally without sharing raw data [9]. Convolutional neural network (CNN) and Long Short-Term Memory (LSTM) architectures were employed to extract spatial and temporal patterns from network traffic data. The framework's performance was assessed against traditional centralized models to evaluate efficiency and accuracy. The proposed federated model maintained detection accuracy similar to centralized approaches while enhancing data privacy. It effectively reduced the risk of data exposure in IoT networks. FL provided an efficient and privacy-preserving solution for decentralized DDoS attack detection, ensuring robust security in IoT environments.

Instead of deploying mitigation intelligence around the victim, FLEAM [11] placed it on the attacking path in a distributed, attacker-centric manner. Policies and packet symbols were used for DDoS detection instead of relying on patterns in the arriving packets. A collaborative and active mitigation strategy provided more accurate and faster results. The mitigation response time was reduced by 72%, while the accuracy was increased by 47% compared to centralized training.

The architecture combined Proof of Authentication (PoAh) from [15] consensus with FL to enhance security, privacy, and data authentication in IoT environments. The architecture was structured into four layers: device, federated, authentication, and cloud. FL ensured privacy-preserving local model training, while PoAh offered lightweight, decentralized authentication suitable for resource-constrained IoT devices. Experimental results showed that the proposed system achieved high accuracy ($\approx$ 98.6%), precision, recall, and efficiency for detecting DDoS attacks, outperforming existing models.

The asynchronous FL model (AsyncFL-bLAM) combined bidirectional LSTM and attention mechanisms to detect low-rate DDoS (LDDoS) attacks in IoT environments [16]. It featured a leader node election algorithm and a weight correction mechanism to ensure robust, privacy-preserving model aggregation. The model effectively processed time-series data using a sliding window technique and handled missing or noisy data. Experimental results showed high accuracy (98.8%), outperforming traditional ML and existing FL methods.

A secure and reliable DDoS detection framework integrated Federated Machine Learning (FML) with Blockchain technology, featuring a reputation-based miner selection and incentive mechanism that ensured only trustworthy nodes participated in training, thereby protecting the model from poisoning attacks [17]. The trained global model was securely stored on the blockchain, enhancing model integrity and system reliability. Experimental results showed that the RF model achieved 99.1% accuracy, outperforming other classifiers and existing methods.

FedDB integrated personalized FL (PFL) with DBSCAN clustering to improve DDoS attack detection while preserving data privacy [18]. It used LSTM neural networks and mutual learning to enhance model performance across heterogeneous clients. DBSCAN clustered similar models to reduce noise and effectively address the non-IID data problem. Experimental results on the CICDDoS2019 dataset showed that FedDB achieved high and consistent accuracy (up to 97%) under both balanced and imbalanced data conditions, outperforming FedAvg and FedMe.

The collaborative FL (CFL) framework, optimized for 6G networks, dynamically balanced model accuracy and response time across device-level, edge, and cloud learning [19]. A deep reinforcement learning (DRL) controller selected the optimal collaboration strategy based on network and device conditions. When applied to DDoS attack detection, the CFL approach achieved superior performance in terms of accuracy, latency, and adaptability compared to traditional FL models. The system used GRU-based neural networks for efficient time-series traffic analysis.

EFLDDoS was an evidence-based FL framework for classifying DDoS attacks in Industrial IoT environments [20]. It integrated Dempster-Shafer (DS) theory into FL to handle uncertainty and enable set-valued classification of mixed attack types. A CNN model was trained locally on IoT devices, and DS theory was used to combine uncertain outputs, improving classification accuracy without compromising data privacy. Experimental results on the CICDDoS2019 dataset showed improved performance over traditional FL and CNN models, particularly for complex, hybrid DDoS attacks.

A federated software defined network (SDN)-based solution to detect and mitigate distributed denial of service (DDoS) attacks in a collaborative, distributed manner introduced the Network Detection and Prevention Agent (NDPA) algorithm, which dynamically adjusted traffic throughput by reconfiguring switches and routers to limit excessive data flow [21]. Detection was initiated by either the victim server or the edge SDN controller based on system metrics like CPU, memory, and network usage. Experimental results demonstrated that the system successfully detected attacks, prevented server crashes, and restored normal throughput to ensure service continuity and improve quality of service (QoS).

FL-DAD used an FL-based approach with CNNs for decentralized DDoS attack detection in IoT networks [22]. It enabled local IoT nodes to train models on-device, preserving privacy while reducing communication overhead. Evaluated on the CICIDS2017 dataset, the model achieved over 98% accuracy across various DDoS attack types. The approach was highly scalable, adaptive to network size, and effective in minimizing false positives and negatives. It also reduced communication overhead over training rounds, making it suitable for large-scale IoT deployments.

A FL framework using a hybrid ResVGG-SwinNet model (combining ResNet, VGGNet, and Swin-Transformer) was used to detect DDoS attacks in IoT networks, emphasizing privacy by avoiding centralized data aggregation and enhancing performance with novel preprocessing techniques and feature optimization [23]. The model achieved 99% accuracy with low false alarm rates across multiple benchmark datasets. Its design supported scalability, efficient training, and resilience in resource-constrained, heterogeneous IoT environments.

A privacy-preserving and explainable DDoS detection framework, utilizing Federated Deep Neural Networks (FDNN) and Explainable AI (XAI) techniques, leveraged SHAP with XGBoost

Table II. Comparison of existing works

Study	Focus	Technique-client	Technique-server	Dataset	Finding	Relevance
8	Client selection and resource allocation	MLP-based adaptive FL	FedAvg	CIC-DDoS2019	Identifying attack types	Optimal FL global model
9	IoT DDoS detection	CNN and LSTM	Weighted averaging	CICIDS2017	Aggregation protocol	Detection basis and federated insight
11	Edge mitigation	Gated recurrent unit (GRU)	An iterative model averaging	UNSW NB15 dataset	Edge success	Federated insight
15	IoT DDoS	(GRU neural network model	Aggregation model	From Kaggle – Application Layer DoS Attack Dataset	Ensures data authentication and validation, high security	Detection strategy
16	Low-rate DDos	Bidirectional LSTM (bi-LSTM) and attention mechanism	Weighted averaging	ISCX-2016-SlowDos	Reduces the overall communication rounds	Detection strategy
17	Protect the blockchain attacks	Random forest, multilayer perceptron, and logistic regression	Safe multi-party computation or federated averaging methods	IDS 2018 Intrusion CSVs (CSE-CIC-IDS2018) and (CIC-DDoS2019)	Miner selection and incentive calculation method	Detection strategy
18	Preserves data privacy	LSTM with DBSCAN clustering	Aggregation model	CICDDOS2019	Address the issue of non-IID data distribution imbalance	Detection strategy
19	6G-based cloud services	Deep reinforcement learning	Average of model – partial aggregation in edge servers	CICDDoS 2019	Optimal recognition accuracy and response time of recognition	Detection strategy
20	Industrial IoT DDoS attack	Dempster-Shafer (DS) theory	Mass functions based on Dempster's rule within the DS layer	DDoS2019	Attack types	Detection strategy
22	DDoS patterns	CNN	Weighted averaging	CICIDS2017	Attack types	DDoS attacks identification
23	IoT DDoS	ResVGG-SwinNet	Aggregation model	CIC-DDoS2019, UNSW-NB15, and IoT23	Attack patterns	Multi-label DDoS attack detection
24	Heterogeneous IoT DDoS	DNN and XGBoost with SHapley Additive explanations (SHAP)	Aggregation model	CIC IOT 2023	Attack types	DDoS detection

for feature selection and trained DNN models across distributed IoT clients without sharing raw data [24]. Evaluated on the CIC-IoT-2023 dataset, the model achieved high performance with 99.78% accuracy, 99.80% precision, and 99.76% F1-score. The framework proved robust, scalable, and interpretable, making it suitable for real-world IoT security challenges.

Table II presents a comparative analysis of existing works, focusing on primary focus, technique at client, technique at server, dataset, findings, and relevance.

III. WORKING PRINCIPLES OF FL-BASED SECURITY MODEL

The proposed security model includes DDoS threat mitigation with trust handling in the system. The security model operates on the participants, who have registered for cloud services using the authentic registration process. FL-based security model ensures CIA (confidentiality, integrity, and authentication) during the model training. Although FL provides data privacy, it operates in a decentralized and untrusted environment, motivating researchers to develop security models that assume trust boundaries and protective mechanisms. A malicious participant is one of the issues in FL, which needs identification of trust boundaries using zero-trust principles, where post-quantum cryptographic concepts are used to enforce security instead of implicit trust. FL client participation is based on incentive mechanisms such as reputation scoring, access control, and computational rewards.

The proposed security model uses the CRYSTALS-Kyber post-quantum cryptographic algorithm that is lattice-based [14]. It supports key exchange as well as encryption. Participating nodes and the cloud server securely establish a shared secret over an insecure channel. It is selected as it is built on the concept of the complex lattice problem called module learning with errors. The mathematical operations involved are based on polynomials over a finite ring. It is believed to be secure against quantum computers, as it has been selected as a finalist and eventual standard in the NIST Post-Quantum Cryptography Standardization Process [14]. The proposed method assumes that the participating node has to enroll with the cloud server using a secure registration process, wherein the node will be assigned credentials/certificates by the cloud server. The proposed FL-based security model starts with the enrollment phase, followed by key exchange and FL model training with secure data transmission, as shown in Fig. 2. The FL model has two main components, called the local model and the global model.

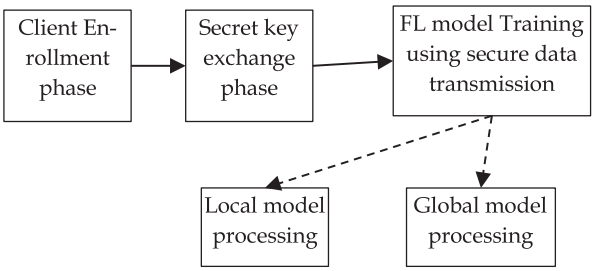

Fig. 2. Components of FL-based security model.

The client enrollment phase requires the client to go through a secure registration process for further authorized communication with the client during the FL process. An existing credential with the cloud server during service registration is used for the authentication of the client. The client has information, including the private key dPrBS and a device certificate containing dPuBS and dID, which are used for secure secret sharing in our model. The cloud server maintains the device certificate containing dPuBS. The enrollment process uses this data.

The shared secret sharing phase requires key pair generation at the cloud server and a shared secret key at the client. The cloud server key pair consists of the public key and private key, denoted as cPuK and cPrK, respectively. The client generates a shared secret key denoted as ss, which is used for securing further data communication using symmetric cryptographic algorithms. Double encryption is used for initial client authentication as ICA and ss sharing as shown in equation (1). This is used only once, so the double encryption cost is ignored:

$$ICA = E_{cPuK}\{E_{dPrBS}\{dID, ss\}\} \tag{1}$$

The server can decrypt the message with its private key, then again decrypts with the client's public key to get the dID and ss as shown in equation (2):

$$IdID, ss = D_{dPuBS}\{D_{cPrK}\{dID, ss\}\} \tag{2}$$

This shared secret key is used to share FL model parameters.

As shown in Fig. 3, FL trains multiple devices locally, and only model parameters are transmitted to the cloud server rather than data from each device using a shared secret. Consider the number of nodes participating in FL training is N. Each node, n_i, has its local dataset denoted as DB_i with S_i samples. The process starts with sharing global model parameters denoted as G, with all nodes N. The participating nodes train the ML/DL model using the received G and its DB_i. Each node computes δG to indicate the requirement to update the global model as shown in equation (3):

$$\delta G_i = Train(G, DB_i) \tag{3}$$

Cloud server receives δG_i from the participating nodes. If it does not receive it, the previous δG_i is considered. Aggregation of all δG_i indicates the new G shown in equation (4):

$$G_{new} = G + \alpha \sum_{i}^{N} \frac{Si}{S} \delta G_i \tag{4}$$

where α is the learning rate and $S = \sum_1^N S_i$ is the total number of samples available from all nodes. The FL process, shown in Fig. 3, uses the CRYSTALS-Kyber post-quantum cryptographic algorithm with the generated shared secret. The process is repeated

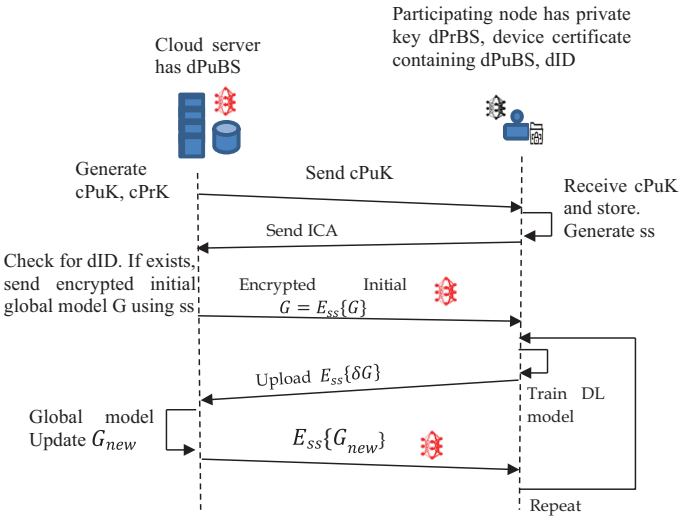

Fig. 3. Working of the FL process for DDoS detection.

until the model converges or the predefined criteria/condition is satisfied.

A. DATASET DESCRIPTION

Exploratory data analysis (EDA) is a process of analyzing and summarizing a dataset to understand its properties and characteristics. The goal of EDA in this work is to gain insights and a deeper understanding about the DDoS attack data from the Canadian Institute for Cybersecurity (CIC) and to identify potential outliers that may indicate unusual or suspicious activity, trends, and patterns that may be useful in detecting and preventing DDoS attacks. EDA can involve a variety of techniques such as visualizing the data using graphs and plots, summarizing the data using statistical measures, and identifying patterns and correlations in the data using ML algorithms. For example, EDA is used to determine the most common types of DDoS attacks that are observed in the CIC data, as well as the geographic regions and IP addresses that are most associated with DDoS attacks.

The DDoS Evaluation Dataset (CIC-DDoS2019) is a large-scale dataset that contains several types of DDoS attacks and regular traffic. It is a challenging and complex dataset that requires specialized knowledge and tools to work with effectively. However, it is also a valuable resource for researchers and practitioners working in the field of network security and cloud security, as it provides a rich source of real-world data for ML tasks, including data analysis. It comprises a total of 80 network traffic files, with a total size of approximately 125 GB. The dataset contains more than 16 million network flows, including over 8 million benign flows and over 8 million attack flows.

Because of the reasons mentioned above, we have selected this dataset to test and evaluate deep learning algorithms in FL for their effectiveness in detecting and classifying DDoS attacks. The data preprocessing steps are applied to rectify inconsistent values or missing values. The numerical attributes in the dataset are normalized using min-max scalar techniques. The categorical attributes in the dataset are encoded using the one-hot encoding technique to convert them into a binary matrix representation. The preprocessed dataset is mimicked for a federated structure so that multiple nodes can represent it, as shown in Fig. 4. The client dataset is reshaped to 8×8 to treat it like an image (for CNN).

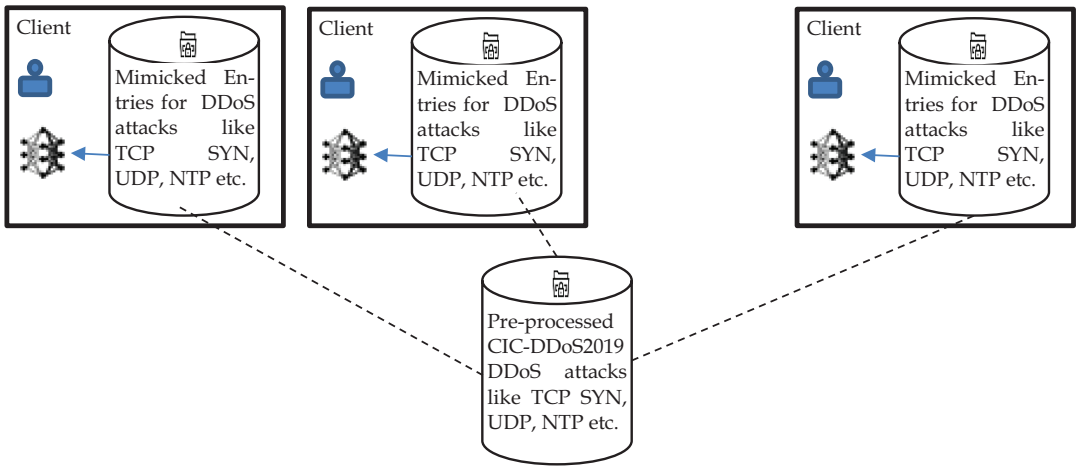

Fig. 4. CIC-DDoS2019 dataset for FL approaches.

Table III. CNN and Bi-LSTM architecture used in the proposed FL

Layer type	Bi-LSTM model	CNN model
Input	Input shape: 1×8×8, where 8×8 data is treated as a time series (sequence of 8 steps, each with 8 features).	Input shape: 1×8×8 where 8×8 is treated like an image
Input	[batch, 8, 8] (8 timesteps, 8 features each)	[batch, 1, 8, 8] (grayscale image-like format)
Layer 1	Bi-LSTM (hidden_size = 512)	Conv2D (1 → 2 filters, kernel = 3×3, padding = 1), activation: ReLU, output shape: 2×8×8
Pooling layer	None	MaxPooling2D (kernel = 2×2), output shape: 2×4×4
Flatten	None	output shape: 32 (2×4×4)
Fully connected 1	Linear (512×2 → num_classes)	Linear (2×4×4 → 8), ReLU + Dropout ($P = 0.8$), output shape: 8
Fully connected 2	N/A	Linear (8 → num_classes)
Dropout	Optional	Dropout ($P = 0.8$)
Output activation	Softmax (via loss function)	Softmax (via loss function)

B. DEEP LEARNING MODEL

Bi-LSTM is used in FL for comparing the performance with CNN-based FL [9]. The architecture of Bi-LSTM is given in Table III, while the model training dynamics are elucidated by employing hyperparameters given in Table IV. Their accuracy, efficiency, robustness, and other metrics are compared to determine the most effective algorithm(s) for classifying normal and attack instances. Although CICDDoS2019 is a tabular dataset, we are reshaping the tabular data to a 2D format (8×8) so that CNNs can be applied, mimicking image input.

IV. SECURITY ANALYSIS

This section discusses security analysis of the proposed FL-based security model for node authentication, message authentication, confidentiality, and integrity of messages under adversarial conditions, including impersonation, eavesdropping, and message forgery.

The first message communicated between the client and the cloud server contains cPuK, which anyone can easily access. The subsequent exchange of confidential information, such as shared secret (ss) and device identity (dID), is protected using Kyber's post-quantum key encapsulation mechanism. The usage of double encryption provides security to ss and dID. Suppose an attacker wants to impersonate the authorized client. Knowing dID, he can create his own ss. When he tries to encrypt the (dID, ss) message, he doesn't have a dPrBS of dID, so he cannot generate ICA as shown in Fig. 5.

Even if he tries to use his private key dPrBS "and generate ICA" as he has cPuK. When the cloud server receives the ICA and attempts to decrypt, it cannot do so as the dPuBS will not be able to decrypt the message. If the attacker adds his dID and his ss, the cloud server will not be able to decrypt the message, as he doesn't have the corresponding public key. Hence, the client is authorized.

If the attacker tries to decrypt the message, he cannot do so as he doesn't have the private key of the cloud server. Only the client and the cloud server have access to the SS. This ensures the client and server have a secure channel.

The messages between the client and cloud server are encrypted using ss, which is known to the client and cloud server only. This ensures message authentication and confidentiality. No attacker can have access to the message, ensuring message integrity also.

These cryptographic properties for node authentication, message authentication, confidentiality, and integrity of messages

Table IV. Hyperparameters used for the Bi-LSTM-based FL model

Hyper parameter	CNN value	Bi-LSTM
Hidden size	N/A	512
Learning rate	0.1 – controls step size in gradient descent	0.001 – controls how much weights are updated during backpropagation
Batch size -	32 → number of samples per weight update step, which affects convergence speed and stability	128 – number of samples per weight update step
Epochs -	15 indicates the number of times the model sees the entire dataset. It stops earlier if validation accuracy plateaus	15 – the number of times the model considers the whole dataset.
Optimizer	Stochastic Gradient Descent – chooses how the model updates weights	RMSProp ($\alpha = 0.9$) – adapts the learning rate per parameter in addition to non-stationary objectives handling
Loss function	CrossEntropyLoss – suitable for multi-class classification like DDoS attack types	CrossEntropyLoss
Dropout rate	0.8	N/A
Number of filters	2 in Conv2D layer – for a low-capacity model to avoid overfitting. Only two filters keep computations light.	512 hidden units in LSTM (256 forward + 256 backward) – allows for capturing both past and future contexts in the DDoS traffic patterns. Acts like filters in CNN, but over sequences
Kernel size	3×3 in Conv2D → learns local spatial relationships in the reshaped 8×8 matrix.	N/A – but processes each feature vector sequentially
Pooling size	2×2 Max Pooling → reduces feature map size (8×8 → 4×4) and adds translation invariance	N/A

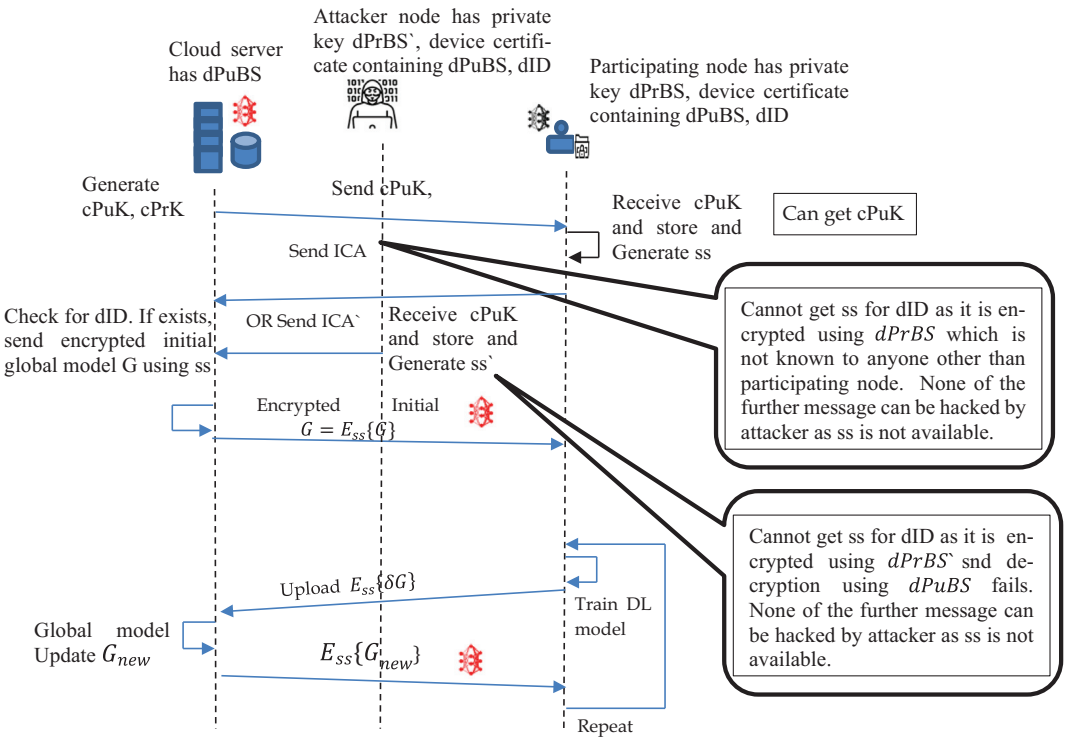

Fig. 5. Summary of security analysis at each step.

under adversarial conditions ensure that the probability of a successful attack is negligible, but at the cost of double encryption.

The paramount security is due to the key exchange and encryption processes based on the CRYSTALS-Kyber algorithm, which has been proven to be indistinguishable under an adaptive chosen ciphertext attack. It is the strongest, widely accepted security notion for public key encryption schemes. It is based on the module learning with errors problem—a lattice-based assumption believed to be hard even for quantum adversaries.

V. PERFORMANCE ANALYSIS

By testing the deep algorithms on this dataset, we can compare their accuracy, efficiency, robustness, and other metrics to determine the

Table V. Performance parameters of the deep learning model in FL

Model	Accuracy	Precision	Recall	F1-score	AUC	FPR	FNR	Training time (s)	CPU (%)	Memory (%)
CNN	0.8146	0.7318	0.814	0.7318	0.6115	0.00	99.97	2023.08	3	22
Bi-LSTM	0.9998	0.9998	0.9998	0.9998	0.9984	0.0026	0.0001	1617.93	12.27	24.89

most effective algorithm(s) for classifying normal and attack instances.

The proposed FL is experimented using the TensorFlow Federated framework, which is the state-of-the-art library for FL. The experimentation outcome is discussed in terms of various performance parameters, such as accuracy, as shown in Table V.

Our experimental results demonstrate the observed absolute gain for Bi-LSTM–FL in terms of accuracy, precision, recall, F1-score, Area Under the Curve (AUC), and training time compared to CNN-FL. The experimental results show insight into effective DDoS detection using a Bi-LSTM model. The proposed model boasts high accuracy, precision, recall, F1-score, and AUC. Accuracy indicates overall correctness. If the DDoS classes are imbalanced, then it may mislead. Precision gives the number of attacks that are attacks. Recall shows the number of actual attacks correctly identified. F1-score balances precision and recall. False

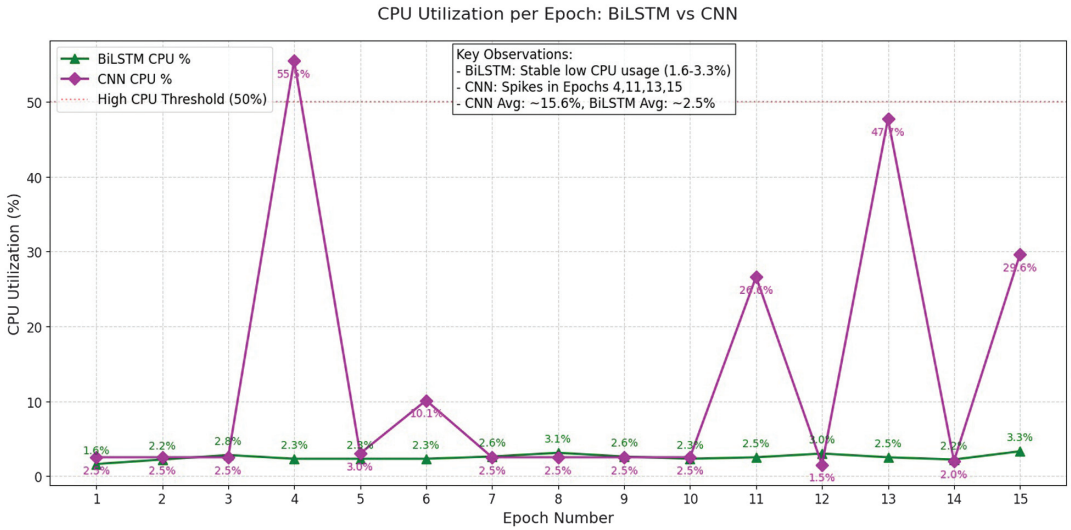

Fig. 6. Comparison of CPU utilization per epoch.

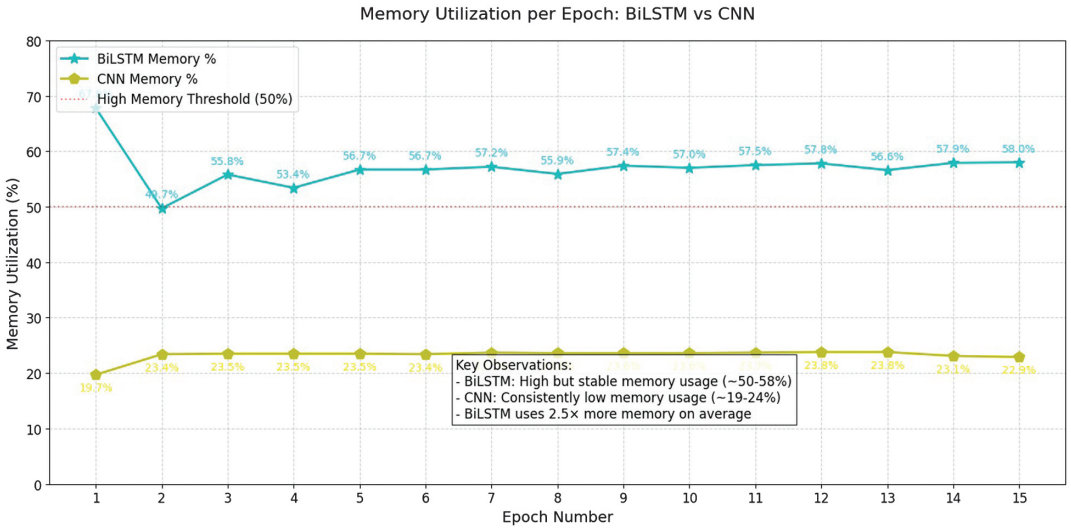

Fig. 7. Comparison of memory usage per epoch.

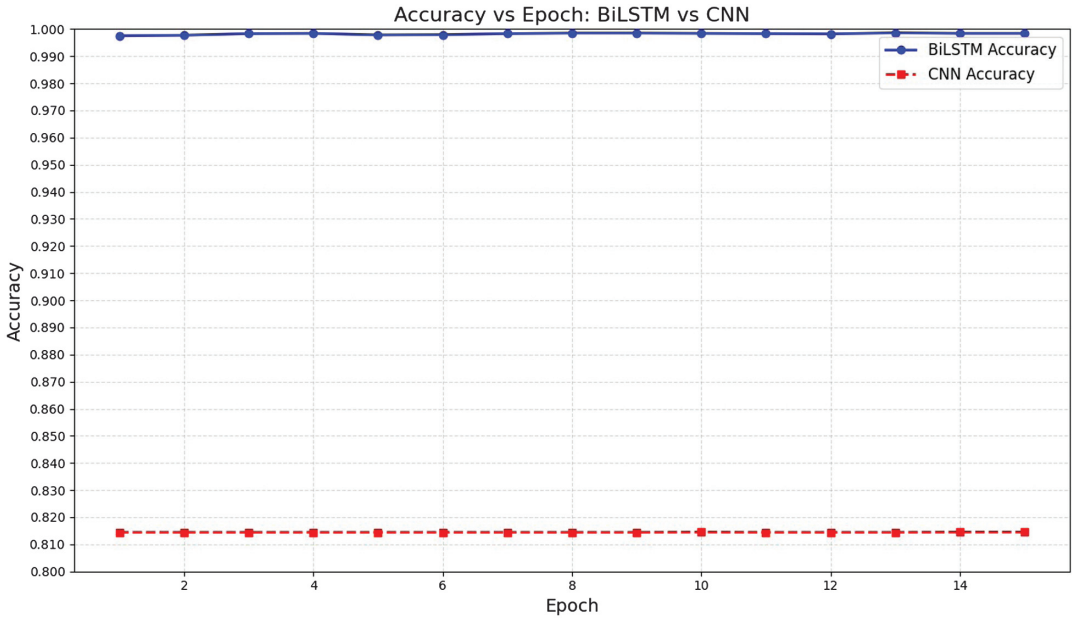

Fig. 8. Comparison of model accuracy.

Table VI. Scalability check with increasing number of nodes

	Training time (s)		Bandwidth (MB)		Accuracy		Data exchanged (MB)	
Nodes	**CNN**	**Bi-LSTM**	**CNN**	**Bi-LSTM**	**CNN**	**Bi-LSTM**	**CNN**	**Bi-LSTM**
5	2198.72	1612.00	53.95	36.64	0.8145	0.9998	0.09	2501.95
10	2143.44	1591.25	43.60	40.82	0.7794	0.9994	0.17	5003.91
15	2110.18	1591.42	4.17	41.18	0.7462	0.9988	0.26	7505.86
20	2115.53	1581.71	5.38	41.50	0.7131	0.9766	0.35	10007.81
25	2100.25	1601.66	6.10	41.61	0.6786	0.9732	0.43	12509.77

positive rate (FPR) and false negative rate (FNR) should be minimal, as they indicate benign attacks and vice versa. FPR uses resources unnecessarily for inspecting benign traffic, while FNR undetects potential threats. The CNN-FL model had a perfect FPR of 0 and a high FNR of 99.97%, indicating a failure to detect almost all actual attacks. The Bi-LSTM-FL model achieves lower FPR (0.0026) and FNR (0.0001) compared to the CNN-FL model. The CNN-FL model may not be suitable for DDoS detection scenarios. CPU and memory usage are moderately higher than those of CNN. As we are using the security model using post-quantum cryptography, the improved detection performance justifies the trade-off for critical DDoS detection tasks in federated settings.

The CPU and memory usage during training of the federated BiLSTM and federated CNN for DDoS detection are shown in Figs. 6 and 7, respectively. The CPU utilization remains low and steady, ranging from 2% to 4%, for federated BiLSTM. In contrast, the performance varies for federated CNNs in detecting DDoS attacks. Memory usage for federated BiLSTM begins higher, approximately around 68% but maintains a steady position close to 55–58% through the training epochs. For a federated CNN, the initial value is 20 and increases by 4 to 5% throughout the training epochs. Although memory utilization doubles in federated BiLSTM, it is accepted due to the benefits of data privacy, scalability, and learning in cloud-based distributed environments, which are not applicable to federated CNN. Accuracy per epoch is almost constant in both models, but it is higher in federated BiLSTM compared to federated CNN, as shown in Fig. 8.

The performance is observed by increasing the number of nodes/clients to ensure robustness and scalability of the proposed FL model, as shown in Table VI. It indicates that the bi-LSTM model maintains high accuracy, low training time, and almost constant bandwidth, while linearly increasing the data exchanged between the client and cloud server, which is a communication overhead compared to the CNN model. Overhead is computed as the amount of data transmitted during each epoch. Overhead is reduced as an increase in epoch number indicates the convergence of the model.

VI. CONCLUSIONS

The proposed FL-based security model ensured node authentication, message authentication, and message confidentiality between clients and the cloud server. Post-quantum cryptography has achieved node authentication and message authentication, whereas the AES symmetric algorithm provides confidentiality. The use of BiLSTM for analyzing and detecting DDoS attacks was found to be

an efficient method compared to CNN across multiple performance parameters like training time, bandwidth, accuracy, and data exchanged. The performance was evaluated under varying node sizes using the CIC-IDS 2019 dataset. The evaluation of these models using metrics such as accuracy, precision, recall, and F1 measure provided valuable insights into their performance. The Bi-LSTM consistently achieved accuracy above 97%, whereas CNN accuracy linearly decreases from 81% to 67% with an increase in the number of clients. Although BiLSTM trained faster with stable bandwidth compared to CNN, the data exchange overhead aligned with FL dynamics. This study highlighted the potential of these methods for accurately identifying DDoS attacks in real-world scenarios.

CONFLICT OF INTEREST STATEMENT

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

REFERENCES

[1] F. J. Abdullayeva, "Distributed denial of service attack detection in E-government cloud via data clustering," *Array*, vol. 15, p. 100229, 2022.

[2] M. Ouhssini, K. Afdel, M. Akouhar, E. Agherrabi, and A. Abarda, "Advancements in detecting, preventing, and mitigating DDoS attacks in cloud environments: a comprehensive systematic review of state-of-the-art approaches," *Egypt. Inform. J.*, vol. 27, p. 100517, 2024.

[3] M. Alauthman, A. Almomani, M. Alarqan, B. Belaton, M. Al-Betar, and V. Arya, "Information theory-based DDoS attack detection in cloud computing: a systematic survey of approaches, challenges, and future directions," *Int. J. Cloud Appl. Comput.*, vol. 15, no. 1, pp. 1–38, 2025.

[4] S. Kumar, M. Dwivedi, M. Kumar, and S. S. Gill, "A comprehensive review of vulnerabilities and AI-enabled defense against DDoS attacks for securing cloud services," *Comput. Sci. Rev.*, vol. 53, p. 100661, 2024.

[5] G. Kirubavathi, I. R. Sumathi, J. Mahalakshmi, and D. Srivastava, "Detection and mitigation of TCP-based DDoS attacks in cloud environments using a self-attention and intersample attention transformer model," *J. Supercomput.*, vol. 81, no. 3, p. 474, 2025.

[6] D. M. A. A. Afraji, J. Lloret, and L. Peñalver, "Deep learning-driven defense strategies for mitigating DDoS attacks in cloud computing environments," *Cyber Secur. Appl.*, vol. 3, p. 100085, 2025.

[7] B. Ghimire and D. B. Rawat, "Recent advances in federated learning for cybersecurity and cybersecurity for the internet of things," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8229–8249, 2022.

[8] R. Doriguzzi-Corin and D. Siracusa, "FLAD: adaptive federated learning for DDoS attack detection," *arXiv preprint arXiv:2205.06661*, 2022.

[9] Y. Alhasawi and S. Alghamdi, "Federated learning for decentralized DDoS attack detection in IoT networks," *IEEE Access*, vol. 12, pp. 42357–42368, 2024.

[10] M. Asad, S. Shaukat, D. Hu, Z. Wang, E. Javanmardi, J. Nakazato, and M. Tsukada, "Limitations and future aspects of communication costs in federated learning: a survey," *Sensors*, vol. 23, no. 17, p. 7358, 2023.

[11] J. Li, L. Lyu, X. Liu, X. Zhang, and X. Lyu, "FLEAM: a federated learning empowered architecture to mitigate DDoS in industrial IoT," *IEEE Trans. Ind. Inf.*, vol. 18, no. 6, pp. 4059–4068, 2021.

[12] Q. Li et al., "A comprehensive survey on DDoS defense systems: new trends and challenges," *Comput. Netw.*, vol. 233, p. 109895, 2023.

[13] M. A. Ferrag, O. Friha, L. Maglaras, H. Janicke, and L. Shu, "Federated deep learning for cyber security in the Internet of things: concepts, applications, and experimental analysis," *IEEE Access*, vol. 9, pp. 138509–138542, 2021.

[14] R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, and D. Stehlé, "CRYSTALS-Kyber algorithm specifications and supporting documentation," *NIST PQC Round*, vol. 2, no. 4, pp. 1–43, 2019.

[15] J. H. Park, S. Yotxay, S. K. Singh, and J. H. Park, "PoAh-enabled federated learning architecture for DDoS attack detection in IoT networks," *Hum.-Centric Comput. Inf. Sci.*, vol. 14, no. 3, pp. 1–25, 2024.

[16] Z. Liu, C. Guo, D. Liu, and X. Yin, "An asynchronous federated learning arbitration model for low-rate DDoS attack detection," *IEEE Access*, vol. 11, pp. 18448–18460, 2023.

[17] D. Saveetha, G. Maragatham, V. Ponnusamy, and N. Zdravković, "An integrated federated machine learning and blockchain framework with optimal miner selection for reliable DDoS attack detection," *IEEE Access*, vol. 12, pp. 127903–127915, 2024.

[18] Y. C. Lee, W. C. Chien, and Y. C. Chang, "FedDB: a federated learning approach using DBSCAN for DDoS attack detection," *Appl. Sci.*, vol. 14, no. 22, p. 10236, 2024.

[19] S. Kianpisheh and T. Taleb, "Collaborative federated learning for 6G with a deep reinforcement learning based controlling mechanism: a DDoS attack detection scenario," *IEEE Trans. Netw. Serv. Manag.*, vol. 21, no. 4, pp. 4731–4749, 2024.

[20] J. Cheng and Z. Jin, "Evidence-based federated learning for set-valued classification of industrial IoT DDoS attack traffic," *J. Internet Things*, vol. 4, no. 3, pp. 183–195, 2022.

[21] O. El Aeraj and C. Leghris, "Intrusion detection system based on an intelligent multilayer model using machine learning," *J. Artif. Intell. Technol.*, vol. 4, no. 4, pp. 332–341, 2024.

[22] M. Dilshad, M. H. Syed, and S. Rehman, "Efficient distributed denial of service attack detection in internet of vehicles using Gini index feature selection and federated learning," *Future Internet*, vol. 17, no. 1, p. 9, 2025.

[23] A. A. Alshdadi, A. A. Almazroi, N. Ayub, M. D. Lytras, E. Alsolami, F. S. Alsubaei, and R. Alharbey, "Federated deep learning for scalable and privacy-preserving distributed denial-of-service attack detection in internet of things networks," *Future Internet*, vol. 17, no. 2, p. 88, 2025.

[24] A. Almadhor, A. Altalbe, I. Bouazzi, A. A. Hejaili, and N. Kryvinska, "Strengthening network DDoS attack detection in heterogeneous IoT environment with federated XAI learning approach," *Sci. Rep.*, vol. 14, no. 1, p. 24322, 2024.

[25] S. Mian, "Foundations of artificial intelligence and applications," *J. Artif. Intell. Technol.*, vol. 2, no. 1, pp. 1–2, 2022.

[26] B. Liu, E. B. Blancaflor, T. Fang, and L. Cao, "Privacy protection based on federated learning," *J. Artif. Intell. Technol.*, vol. 5, pp. 10–19, 2025.

[27] M. Nkoom, S. G. Hounsinou, and G. V. Crosby, "Securing the internet of robotic things (IoRT) against DDoS attacks: a federated learning with differential privacy clustering approach," *Comput. Secur.*, vol. 155, p. 104493, 2025.

[28] M. Abdullah, H. A. Mengash, M. Maray, F. A. Alrslani, H. Alkhudhayr, N. A. Alghanmi, and J. Majdoubi, "Federated learning with blockchain on denial-of-service attacks detection and classification of edge IIoT networks using deep transfer learning model," *Comput. Electr. Eng.*, vol. 124, p. 110319, 2025.

[29] R. Abdelhadi, M. H. Alsafasfeh, and B. I. Alqudah, "Encountering distributed denial of service attack utilizing federated software defined network," *Int. J. Electr. Comput. Eng.*, vol. 14, no. 1, pp. 574–588, 2024.

[30] Y. S. N. Fotse, V. K. Tchendji, and M. Velempini, "Federated learning based DDoS attacks detection in large scale software-defined network," *IEEE Trans. Comput.*, vol. 74, no. 1, pp. 101–115, 2025.