Natural Resources for Human Health
eISSN: 2583-1194

Original Research

Received 09 May 2026
Revised 01 June 2026
Accepted 12 June 2026

Natural Resources for Human Health 2026, Vol. 6 (11s): 759–780

KEYWORDS:
Federated Learning, Anomaly Detection, Internet of Things, Intrusion Detection, IoT Communication Security, Privacy-Preserving Machine Learning

Federated Learning Based Anamoly Detection for Secure and Resilient Internet of Things Communication Network

Salman Arafath Mohammed[1], Mohammed Nazneen Fathima[2], Salman Ali Syed[3], Abdulilah Mohammad Mayet[4]

[1]Electrical Engineering Department, Computer Engineering Section, King Khalid University, Abha 61411, Saudi Arabia
[2]Electronics and Communication Engineering, Bharat Institute of Engineering and Technology, Hyderabad, India
[3]Assistant Professor, Department of Computer Science, College of Computer and Information Sciences, Sakaka, Jouf University, Al-Jouf Province, Kingdom of Saudi Arabia
[4]Electrical Engineering Department, King Khalid University, Abha 61411, Saudi Arabia
Corresponding Author: Salman Arafath Mohammed
Email: salman@kku.edu.sa
ORCID: 0000-0003-3598-7610

ABSTRACT: The rapid proliferation of Internet of Things (IoT) devices has created highly distributed communication environments that are increasingly exposed to anomalous traffic, botnet activity, denial-of-service attacks, spoofing, malware propagation, and other cyber threats. Conventional centralized intrusion detection approaches require the collection and transmission of large volumes of potentially sensitive network data, creating privacy, communication overhead, scalability, and single-point-of-failure concerns. Federated Learning (FL) provides a distributed learning paradigm in which IoT nodes or edge clients collaboratively train a shared anomaly-detection model while retaining raw traffic data locally. This paper proposes a federated learning-based anomaly detection framework for improving the security, privacy, scalability, and resilience of IoT communication networks. The framework integrates local traffic analysis, feature extraction, distributed model training, privacy-preserving parameter exchange, and robust global aggregation to identify anomalous communication patterns without centralizing raw network observations. The proposed research considers heterogeneous and non-independent and identically distributed IoT traffic, resource-constrained devices, communication efficiency, model robustness, and adversarial threats against the federated learning process. Performance can be evaluated using accuracy, precision, recall, F1-score, false-positive rate, detection latency, communication cost, and convergence characteristics. Recent research confirms the growing relevance of FL-based intrusion and anomaly detection for IoT, while also identifying unresolved challenges involving heterogeneous data, aggregation security, resource constraints, and standardized evaluation.

1. INTRODUCTION

The Internet of Things (IoT) has developed into a large-scale distributed communication ecosystem connecting sensors, actuators, embedded systems, gateways, edge devices, and cloud platforms. IoT infrastructure now supports applications in industrial automation, healthcare, intelligent transportation, smart homes, energy management, agriculture, surveillance, and

critical infrastructure. The continuous interaction among heterogeneous devices generates large volumes of network traffic and enables automated information exchange across geographically distributed environments. However, the same connectivity that provides operational flexibility also increases the exposure of IoT networks to cyberattacks. Resource-constrained hardware, heterogeneous communication protocols, limited security mechanisms, inconsistent software configurations, and long device lifecycles create multiple opportunities for attackers to compromise IoT nodes and use them for unauthorized access, botnet formation, distributed denial-of-service attacks, scanning, malware propagation, spoofing, and other malicious activities. Consequently, reliable anomaly and intrusion detection has become a fundamental requirement for maintaining the confidentiality, integrity, availability, and resilience of IoT communication networks.

Conventional IoT intrusion detection systems predominantly rely on centralized architectures in which network traffic from multiple devices is collected and transferred to a central server for analysis and model development. Machine learning and deep learning have substantially improved the ability of such systems to identify complex traffic patterns and distinguish legitimate communication from malicious behavior. Deep autoencoder-based detection, for example, has demonstrated the capability to identify anomalous traffic generated by compromised IoT devices and botnets [8]. Nevertheless, centralized learning creates several structural limitations. Continuous transmission of raw network data increases bandwidth consumption and communication latency, while centralized storage may expose sensitive traffic information. The aggregation of security data from multiple organizations or geographically separated IoT environments can additionally create regulatory, privacy, and governance concerns. A centralized server may also become a computational bottleneck or single point of failure when the number of participating devices increases substantially.

The distributed nature of IoT networks makes Federated Learning (FL) particularly relevant as an alternative learning paradigm. In FL, participating clients train a common machine-learning model locally using their own datasets and transmit model parameters or updates rather than raw observations to an aggregation server. The server combines the locally generated updates to construct an improved global model, which is subsequently redistributed to participating clients. This process enables multiple distributed environments to contribute to collaborative learning while retaining local data. FL is therefore capable of addressing several limitations associated with centralized learning, particularly the requirement for raw-data aggregation and the associated communication and privacy concerns [7].

The suitability of FL for intrusion detection has received increasing attention because network-security information is naturally distributed across IoT devices, gateways, edge nodes, and organizational domains. A federated architecture enables security models to learn from multiple local environments without requiring direct access to the underlying traffic records. Recent reviews have identified FL as an important approach for distributed intrusion detection, particularly because it can address data-isolation problems, privacy requirements, communication constraints, heterogeneous data distributions, and distributed computational resources [1], [2]. A recent comprehensive review further distinguishes two security dimensions in FL-based IoT intrusion detection: external network attacks that the IDS must detect and internal attacks targeting the federated learning process itself, including poisoning, backdoor, and Sybil attacks [10]. This distinction is essential because protecting an IoT network through FL introduces a second security layer that must itself be protected.

IoT traffic is inherently heterogeneous. Different devices generate substantially different communication patterns depending on their functions, operating systems, applications, communication protocols, transmission frequencies, and deployment environments. A surveillance camera, industrial sensor, smart meter, wearable device, and connected gateway may therefore exhibit very different baseline traffic characteristics. Consequently, the datasets available to individual FL clients are frequently non-independent and identically distributed (non-IID). This condition can produce divergent local model updates and make global optimization more difficult. The problem is particularly significant when certain clients contain predominantly normal traffic while others contain high proportions of attack traffic. Research on FL-based intrusion detection identifies non-IID data, client heterogeneity, model personalization, and aggregation optimization as continuing challenges [2].

Anomaly detection provides an important mechanism for addressing the dynamic nature of IoT threats. Signature-based detection depends largely on predefined knowledge of known attacks, whereas anomaly-based detection attempts to identify deviations from established behavioral patterns. This distinction becomes increasingly important when attackers modify their tactics or introduce previously unseen attack variants. The N-BaIoT research demonstrated the applicability of deep autoencoders to behavioral anomaly detection in IoT networks, particularly for detecting traffic associated with Mirai and

BASHLITE botnets [8]. A federated anomaly-detection architecture can extend this concept by allowing multiple IoT environments to collaboratively learn generalized representations of malicious behavior while retaining local traffic information.

However, FL does not automatically eliminate security and privacy risks. The transmission of model parameters rather than raw data reduces direct data exposure, but model updates may still reveal information about local datasets under certain attack conditions. More importantly, malicious clients can manipulate local training data or intentionally submit poisoned model updates to influence the global model. Backdoor attacks can cause a model to behave normally for most traffic while producing attacker-controlled predictions for selected inputs. Model poisoning can degrade global performance, while Sybil-style participation can enable an adversary to influence aggregation through multiple apparently independent clients. These threats establish a requirement for security mechanisms that protect the federated training process in addition to mechanisms that detect malicious network traffic [1], [9], [10].

Communication efficiency is another major consideration. Although FL avoids transferring raw datasets, participating clients may need to communicate model parameters repeatedly across numerous training rounds. For resource-constrained IoT devices, repeated model transmission can consume significant bandwidth and energy. Local training also introduces computational requirements that may exceed the capabilities of extremely constrained devices. Recent FL-IDS research therefore emphasizes lightweight model architectures, client selection, communication-efficient optimization, robust aggregation, and edge-assisted learning as important directions for practical deployment [1], [2], [10].

Edge computing can further strengthen the practical implementation of FL-based IoT anomaly detection. Edge nodes and gateways can perform feature extraction, local inference, model training, or aggregation closer to the data source. This reduces dependence on distant cloud infrastructure and can potentially improve response latency. Research on intelligent FL-based intrusion detection for edge-assisted IoT environments demonstrates the relevance of distributed edge resources to collaborative security learning [6]. Similarly, research combining FL and blockchain for industrial IoT has examined decentralized mechanisms for strengthening trust, model coordination, and security within distributed intrusion-detection architectures [9].

The research problem therefore extends beyond achieving a high anomaly-detection accuracy. A practical FL-based IoT security architecture must simultaneously address detection capability, data privacy, model robustness, communication overhead, computational constraints, heterogeneous client behavior, and resilience against malicious participants. A recent systematic review of learning-based IoT intrusion detection identified machine learning, deep learning, transfer learning, and federated learning as major approaches while emphasizing continuing challenges involving datasets, evaluation methodology, generalization, and deployment [11]. These observations indicate that an effective research framework should evaluate both the quality of the resulting anomaly detector and the operational characteristics of the federated learning process.

This study consequently focuses on a Federated Learning Based Anomaly Detection framework for secure and resilient IoT communication networks. The proposed approach considers distributed IoT clients, local traffic preprocessing, feature extraction, local anomaly-model training, global parameter aggregation, anomaly classification, and security-aware federated coordination. The framework is designed to retain raw traffic locally while enabling collaborative model development across distributed clients. Its evaluation should incorporate conventional classification metrics such as accuracy, precision, recall, F1-score, false-positive rate, and false-negative rate together with federated-learning measures including communication overhead, convergence behavior, training rounds, computational requirements, and robustness against malicious updates.

The central contribution of this research is the integration of anomaly detection and federated learning within a unified IoT security framework in which both network-level threats and federated-model-level threats are considered. Rather than treating privacy, anomaly detection, communication efficiency, and resilience as independent objectives, the proposed framework considers them as interconnected requirements of a practical IoT security architecture. This approach provides a basis for developing distributed security intelligence capable of learning from heterogeneous IoT environments without requiring centralized access to raw network traffic.

2. LITERATURE REVIEW

2.1 IoT Intrusion and Anomaly Detection

The rapid expansion of IoT networks has resulted in increasing research attention toward intelligent intrusion and anomaly detection. Conventional signature-based systems remain useful for identifying known attack patterns, but their dependence on predefined signatures limits their ability to recognize previously unseen or rapidly evolving threats. Anomaly-based detection approaches address this limitation by learning behavioral characteristics associated with normal communication and identifying significant deviations. Machine learning and deep learning have consequently become important approaches for developing adaptive IoT intrusion-detection systems.

Deep-learning techniques are particularly useful because IoT attacks may manifest through complex relationships among packet-level, flow-level, temporal, and behavioral characteristics. The N-BaIoT approach employed deep autoencoders to learn normal network behavior from IoT devices and detect anomalous traffic associated with botnet activity [8]. Its use of traffic generated by multiple commercial IoT devices demonstrated that device-level behavioral characteristics can provide useful information for identifying compromised nodes. Such approaches established the basis for more sophisticated learning-driven IoT security systems.

A systematic review of learning techniques for IoT intrusion detection examined 646 publications from 2018 to 2023 and categorized relevant approaches into machine learning, deep learning, transfer learning, federated learning, and hybrid combinations [11]. The analysis indicates that learning-based IDS research has progressed from conventional centralized classification toward distributed and privacy-aware learning architectures. Nevertheless, differences in datasets, attack categories, experimental configurations, and evaluation metrics make direct comparison among studies difficult.

2.2 Federated Learning for IoT Security

Federated learning changes the conventional centralized learning paradigm by allowing clients to retain their training data locally. Each participating client performs local optimization and communicates model updates to a coordinating server. The server aggregates these updates to construct a global model. The resulting model can therefore incorporate knowledge derived from geographically distributed datasets without requiring direct centralization of the underlying traffic records.

The application of FL to IoT is particularly appropriate because IoT environments are inherently distributed. Devices may belong to different users, organizations, geographical regions, or administrative domains, making centralized collection of security data impractical or undesirable. Nguyen *et al.* comprehensively examined FL in IoT and identified communication efficiency, resource constraints, data heterogeneity, privacy, and distributed optimization as major challenges [7]. These challenges are directly applicable to federated intrusion detection.

For cybersecurity applications, FL can enable multiple organizations or network segments to collaboratively improve an IDS while maintaining local control of sensitive traffic. Belenguer, Pascual, and Navaridas reviewed FL applications in IDS and identified the reduction of centralized data dependence, privacy preservation, and distributed learning as major motivations for FL-based intrusion detection [1]. Their analysis also identifies insufficient standardization of evaluation practices as an important limitation in the existing literature.

2.3 Federated Learning-Based Intrusion Detection

Research on FL-based IDS has expanded considerably as researchers have attempted to combine distributed learning with network-security analysis. Man *et al.* developed an intelligent FL-based intrusion-detection approach for edge-assisted IoT environments, demonstrating the potential of distributed edge resources for collaborative threat detection [6]. The edge-assisted architecture is significant because it provides a computational layer between highly constrained IoT devices and centralized cloud infrastructure.

Lazzarini, Tianfield, and Charissis investigated FL for IoT intrusion detection and emphasized its ability to support collaborative security learning while preserving the locality of IoT data [5]. Such architectures can provide a broader view of attack behavior than isolated local detectors because information from multiple participating clients contributes to the global model.

More recent research has experimentally compared federated aggregation mechanisms for IoT intrusion detection. Khraisat *et al.* developed a privacy-oriented IoT defense architecture and evaluated FedAvg and FedAvgM using the N-BaIoT dataset [4]. The study demonstrates the continuing relevance of N-BaIoT as an experimental benchmark while also illustrating the

importance of aggregation strategy in federated IoT security. Momentum-based aggregation can be particularly relevant when local updates are noisy or heterogeneous.

A comprehensive survey by Zhang *et al.* organized FL-based intrusion detection research around application scenarios, federated learning algorithms, privacy and security protection, classification models, data sources and client distributions, and evaluation metrics [2]. The study identifies aggregation optimization, security enhancement, model personalization, improved classification models, and the use of unlabeled data as important research directions. These findings indicate that the design of an FL-based IDS requires simultaneous consideration of the learning algorithm, dataset distribution, security architecture, and evaluation methodology.

2.4 Heterogeneous and Non-IID IoT Data

Data heterogeneity represents one of the fundamental challenges of federated IoT intrusion detection. In centralized learning, data from different devices can be combined and shuffled before model optimization. In FL, however, local datasets remain distributed. Consequently, each client may possess a different number of observations, different attack proportions, different device types, and different traffic distributions. This creates non-IID local datasets and can result in divergent model updates.

The impact of heterogeneous data is particularly important in IoT because individual devices often have specialized functions. Normal traffic generated by an industrial controller may differ considerably from that generated by a smart camera or wearable device. An anomaly detector trained predominantly on one device category may therefore fail to generalize effectively to another. FL can potentially learn a broader representation through collaboration, but simple parameter averaging may not always provide an optimal model for every client.

Recent surveys identify non-IID data as a central research problem in FL-based intrusion detection [2], [10]. Personalization, adaptive aggregation, client clustering, transfer learning, and selective participation have consequently emerged as potential mechanisms for improving performance under heterogeneous client conditions. The challenge is to achieve sufficient global generalization without eliminating the device-specific information necessary for accurate local anomaly detection.

2.5 Privacy, Robustness, and Adversarial Security

Privacy is one of the primary motivations for applying FL to IoT intrusion detection. Local traffic remains on the participating device or edge node, reducing the requirement to transfer raw observations to a centralized repository. However, privacy protection is not absolute because model updates can potentially expose information about local training data. Secure aggregation, differential privacy, encryption, and other privacy-enhancing techniques can reduce these risks but may introduce additional computational or accuracy costs.

Security threats against the FL process constitute another critical research dimension. An attacker may compromise a client and manipulate its local training data before generating a model update. Alternatively, an attacker may directly construct malicious model updates designed to degrade global performance or introduce hidden backdoors. The resulting threat is fundamentally different from the network attacks that the IDS is intended to detect. The FL system therefore requires mechanisms capable of distinguishing legitimate heterogeneous updates from malicious updates.

Recent research explicitly separates external network-level attacks from internal model-layer attacks in FL-based IoT IDS architectures [10]. This two-layer threat model is particularly important for resilient anomaly detection because a highly accurate detector can still become ineffective if an adversary compromises the federated training mechanism. Robust aggregation, anomaly detection over client updates, trust-aware client selection, secure aggregation, and adversarial training are therefore important components of resilient FL-based security systems.

Blockchain-based approaches have additionally been investigated for strengthening trust in distributed FL intrusion detection. Ali, Li, and Yousafzai reviewed blockchain and FL combinations for edge-enabled industrial IoT and reported their potential for addressing security, privacy, trust, and distributed coordination requirements [9]. However, blockchain integration may introduce additional computational, storage, and communication requirements, which must be evaluated against the constraints of IoT environments.

2.6 Research Gap

The existing literature establishes that federated learning provides a strong architectural basis for privacy-aware and distributed IoT intrusion detection, but several gaps remain. First, many existing studies emphasize detection accuracy while providing comparatively limited analysis of communication cost, energy consumption, convergence, and computational requirements. Second, non-IID and highly heterogeneous IoT traffic can substantially influence the quality of global model aggregation, yet realistic client distributions are not consistently represented across experimental studies [2], [10]. Third, the security of the federated training process is sometimes considered separately from the network attacks being detected, despite the possibility that compromised clients can directly manipulate the global security model.

Fourth, there remains a need for lightweight architectures suitable for resource-constrained IoT and edge devices. Recent comprehensive research identifies lightweight model design and robust defenses against adversarial FL attacks as important future requirements for IoT intrusion detection [10]. Fifth, evaluation practices remain fragmented across datasets, attack categories, client distributions, aggregation algorithms, and performance metrics. The absence of consistent experimental conditions limits the ability to determine whether reported improvements arise from the underlying model architecture, the aggregation mechanism, the data distribution, or the experimental configuration.

Therefore, the principal research gap lies in developing and evaluating a unified federated anomaly-detection architecture that simultaneously addresses **IoT network threats, heterogeneous data, privacy, communication efficiency, computational constraints, and attacks against the federated learning process**. Addressing these dimensions together is necessary for moving FL-based anomaly detection from experimental distributed classification toward a resilient security architecture suitable for large-scale IoT communication networks. Recent research reinforces this direction by identifying robust aggregation, personalization, security protection, efficient model design, non-IID learning, and adversarial resilience as central areas for further development [1], [2], [10].

3. PROPOSED FEDERATED LEARNING-BASED ANOMALY DETECTION FRAMEWORK

The proposed framework is designed as an engineering-oriented federated security architecture for detecting anomalous communication behavior in heterogeneous IoT environments. The architecture combines local traffic acquisition, preprocessing, feature engineering, local anomaly-model training, secure model-update transmission, robust global aggregation, and continuous anomaly classification. The principal design objective is to ensure that raw IoT traffic remains within the local device or edge domain while the learned security knowledge is collaboratively incorporated into a global detection model. This design is consistent with recent FL-based IoT security research, which demonstrates that local training combined with federated aggregation can reduce the requirement for centralized traffic collection while supporting collaborative intrusion detection [1], [4], [5].

3.1 Overall System Architecture

The proposed architecture consists of five functional layers: **IoT Device Layer, Local Security Layer, Edge/Federated Client Layer, Federated Aggregation Layer, and Security Response Layer**. The IoT Device Layer contains heterogeneous devices such as sensors, smart cameras, industrial controllers, gateways, smart meters, and connected embedded systems. Each device continuously generates communication flows that may contain both legitimate and malicious traffic.

The Local Security Layer performs traffic acquisition and preprocessing without transferring raw traffic outside the local security domain. Packet-level or flow-level information is transformed into numerical features suitable for machine-learning analysis. Features may include packet count, byte count, flow duration, packet rate, protocol distribution, source-destination relationships, connection frequency, transport-layer characteristics, and temporal communication statistics.

The Edge/Federated Client Layer performs local model training. An edge gateway can act as the federated client when individual IoT devices are too resource constrained to execute the complete learning process. This arrangement provides a practical balance between local data ownership and computational feasibility. Edge-assisted federated IDS architectures have previously demonstrated the relevance of this arrangement for distributed IoT security [6].

The Federated Aggregation Layer receives model updates rather than raw network traffic. The aggregation server evaluates the received updates and combines legitimate updates into an updated global anomaly-detection model. Because IoT clients may possess significantly different data distributions, the aggregation process incorporates client weighting and update validation rather than assuming that all clients are statistically identical. Recent research has demonstrated that non-IID data can

substantially affect the stability and accuracy of conventional FedAvg-based intrusion detection, motivating personalized and robust aggregation strategies [2], [4].

The final Security Response Layer uses the global model for inference and generates anomaly scores, attack classifications, and security alerts. The resulting architecture supports continuous model improvement because newly observed local traffic can participate in subsequent federated rounds.

Table 1. Functional Components of the Proposed Architecture

Layer	Principal Components	Primary Function
IoT Device Layer	Sensors, cameras, controllers, smart appliances, gateways	Generate and transmit IoT communication traffic
Local Security Layer	Traffic monitor, feature extractor, preprocessing module	Collect and transform local traffic
Federated Client Layer	Local ML/DL model, optimizer, local trainer	Train anomaly-detection model locally
Secure Communication Layer	Authentication, encryption, protected update transmission	Protect model-update communication
Aggregation Layer	Update validator, robust aggregator, global model	Combine legitimate client knowledge
Detection Layer	Global anomaly model, anomaly scorer	Identify anomalous communication
Response Layer	Alert engine, isolation mechanism, security dashboard	Generate and support security responses

3.2 IoT Traffic Acquisition and Preprocessing

The first stage of the framework transforms raw IoT communication into structured learning instances. Continuous packet capture is not required to be transferred to the federated server. Instead, traffic remains within the local security domain, where flow-level or statistical representations are extracted.

For a communication flow x_i, the feature representation can be expressed as

$$x_i = [f_{i1}, f_{i2}, \ldots, f_{im}]$$

where m denotes the number of extracted features.

The preprocessing pipeline includes duplicate removal, missing-value treatment, outlier handling, normalization, categorical encoding, and class-label preparation. Feature normalization is important because network features can have substantially different numerical scales. A standard min-max transformation can be expressed as

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

where x' represents the normalized feature value.

For highly skewed network variables, logarithmic transformation can additionally be applied before normalization. The resulting feature matrix is partitioned locally according to device or edge-node ownership. No raw local feature matrix is transferred to the aggregation server.

Table 2. Representative IoT Network Features

Feature Category	Representative Features	Security Relevance
Flow statistics	Flow duration, packet count, byte count	Identifies abnormal traffic intensity

Feature Category	Representative Features	Security Relevance
Packet behavior	Mean packet size, packet rate, inter-arrival time	Detects traffic deviations
Protocol information	TCP, UDP, ICMP, HTTP, DNS	Identifies unusual protocol behavior
Connection behavior	Connection frequency, destination count	Detects scanning and botnet activity
Source/destination	Source address, destination address, port	Identifies suspicious communication
Temporal behavior	Burst frequency, activity interval	Detects abnormal periodic behavior
Error behavior	Failed connections, retransmissions	Indicates suspicious or unstable communication
Payload/flow statistics	Header and statistical characteristics	Supports behavioral classification

The local dataset at client k is represented as

$$D_k = \{(x_i, y_i)\}_{i=1}^{n_k}$$

where n_k is the number of local observations and y_i represents the corresponding normal or anomalous class.

3.3 Local Anomaly-Detection Model

Each federated client maintains a local anomaly-detection model initialized from the current global model. The local model may be implemented using a lightweight multilayer perceptron, autoencoder, convolutional neural network, recurrent architecture, or hybrid deep-learning model depending on the characteristics of the traffic data.

For an anomaly-detection architecture, the model produces either a classification probability or reconstruction error. When an autoencoder-based architecture is used, an input vector is transformed into a latent representation and subsequently reconstructed:

$$z = f_{\theta_e}(x)$$

$$\hat{x} = f_{\theta_d}(z)$$

where f_{θ_e} represents the encoder, f_{θ_d} represents the decoder, z is the latent representation, and $\hat{x}$ is the reconstructed input.

The reconstruction error is calculated as

$$E(x) = \frac{1}{m}\sum_{j=1}^{m}(x_j - \hat{x}_j)^2$$

A local traffic instance can then be classified according to

$$A(x) = \begin{cases} 1, & E(x) > \tau_k \\ 0, & E(x) \le \tau_k \end{cases}$$

where $A(x) = 1$ indicates anomalous behavior and τ_k represents the client-specific anomaly threshold.

An adaptive threshold is preferable to a single universal threshold because the normal traffic distribution of an industrial controller may differ significantly from that of a smart camera or wearable device. Recent adaptive FL anomaly-detection research has similarly emphasized local threshold adaptation according to device-specific data distributions [3].

3.4 Federated Training Mechanism

The federated training procedure begins with initialization of the global model θ_t. At communication round t, the server selects a set of participating clients S_t and distributes the current model to them. Each selected client performs local training using its own dataset.

The local optimization problem for client k can be expressed as

$$\theta_k^{t+1} = \theta_t - \eta \nabla L_k(\theta_t)$$

where η represents the local learning rate and L_k represents the loss function associated with client k.

After local optimization, each client generates an update

$$\Delta\theta_k = \theta_k^{t+1} - \theta_t$$

rather than transmitting its original dataset.

The conventional weighted FedAvg aggregation can be represented as

$$\theta_{t+1} = \sum_{k \in S_t} \frac{n_k}{\sum_{j \in S_t} n_j} \theta_k^{t+1}$$

where n_k represents the number of local training samples available at client k.

However, direct FedAvg can become less effective under strongly non-IID IoT traffic because clients with highly different distributions may produce substantially different model updates. Recent studies report degradation of standard aggregation under severe data heterogeneity and motivate the use of FedProx, SCAFFOLD, personalized FL, clustering, or adaptive aggregation mechanisms [2], [4].

3.5 Robust Model Aggregation

The proposed framework incorporates a two-stage aggregation procedure consisting of **update validation** and **weighted aggregation**. Before an update contributes to the global model, the server evaluates its statistical characteristics against the distribution of legitimate client updates.

For each received update, an update-distance measure can be calculated as

$$d_k = \| \Delta\theta_k - \Delta\theta_{ref} \|_2$$

where $\Delta\theta_{ref}$ represents a reference update derived from legitimate participating clients.

A normalized trust value can subsequently be expressed as

$$w_k = \frac{\exp(-\lambda d_k)}{\sum_{j \in S_t} \exp(-\lambda d_j)}$$

where λ controls the sensitivity of the aggregation process to update deviations.

The robust global update can then be formulated as

$$\theta_{t+1} = \theta_t + \sum_{k \in S_t} w_k \Delta\theta_k$$

This mechanism reduces the influence of extreme updates while preserving contributions from legitimate clients with naturally different data distributions. Such a design is important because extreme update behavior can arise either from malicious manipulation or from legitimate but highly heterogeneous traffic. Therefore, update filtering should not depend solely on a fixed distance threshold.

Dynamic aggregation has been investigated specifically for IoT FL intrusion detection, with research showing that clustering clients according to traffic characteristics can improve the recognition of minority attack classes and provide additional resistance to poisoning [10]. Personalized FL research has likewise demonstrated the importance of accounting for local distributional differences rather than forcing every IoT client to use an identical model [12].

3.6 Anomaly Decision and Security Response

After global aggregation, the updated model is distributed to participating clients or edge nodes. The model operates in two modes: **local inference** and **federated retraining**.

During local inference, each incoming traffic instance receives an anomaly probability or anomaly score. A decision threshold determines whether the instance is considered benign or suspicious. For multiclass detection, the output layer additionally assigns the most probable attack category.

The classification performance is evaluated using:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

The false-positive rate is calculated as

$$FPR = \frac{FP}{FP + TN}$$

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively.

The security-response mechanism can classify detected events into different operational levels.

Table 3. Anomaly Decision and Response Logic

Anomaly Condition	Detection Interpretation	Response
Low anomaly score	Normal communication	Continue monitoring
Moderate anomaly score	Suspicious behavior	Increase monitoring frequency
High anomaly score	Probable attack	Generate security alert
Repeated high anomaly score	Persistent malicious behavior	Isolate or restrict source
Known attack signature + high anomaly score	Confirmed malicious activity	Immediate mitigation
Abnormal client model update	Potential FL-layer attack	Exclude/update quarantine
Repeated malicious updates	Potential compromised client	Suspend federated participation

This dual detection mechanism allows the architecture to identify both suspicious network behavior and abnormal federated model behavior.

4. EXPERIMENTAL METHODOLOGY AND PERFORMANCE EVALUATION

4.1 Experimental Design

The proposed framework should be evaluated using a distributed experimental environment in which network-security data are partitioned across multiple simulated IoT clients. N-BaIoT is particularly appropriate for evaluating botnet-oriented anomaly detection because it contains traffic collected from multiple IoT devices and includes malicious behavior associated with Mirai and BASHLITE [8]. Additional datasets such as BoT-IoT, UNSW-NB15, and ToN-IoT can be incorporated to evaluate generalization across different traffic distributions.

The experimental architecture should contain one federated aggregation server and multiple IoT or edge clients. A configuration of **10-20 clients** provides sufficient diversity for controlled experimentation, while a larger simulation of **50-100 clients** can be used for scalability analysis. Client datasets should not be randomly pooled because such pooling would remove the principal FL characteristic. Instead, each client should retain its local partition throughout the training process.

Table 4. Recommended Experimental Configuration

Parameter	Experimental Setting
Federated clients	20
Extended scalability clients	50-100
Initial local epochs	5
Batch size	32
Learning rate	0.001
Optimizer	Adam
Communication rounds	50
Client participation per round	60-80%
Validation split	20%
Test split	20%
Primary aggregation	Weighted FedAvg
Robust aggregation	Adaptive update filtering
Primary detection task	Binary anomaly detection
Extended task	Multiclass attack classification
Random seeds	≥5
Evaluation basis	Mean ± standard deviation

The values provide a practical experimental baseline for an engineering evaluation rather than fixed universal requirements. Local epochs, client participation, and communication rounds should subsequently be varied during sensitivity analysis.

4.2 Dataset Preparation and Client Distribution

The datasets should first undergo common preprocessing so that feature representations are compatible across clients. Duplicate observations should be removed, missing values treated, categorical variables encoded, and numerical attributes normalized. Class imbalance should be measured before model training because IoT security datasets frequently contain substantially different quantities of benign and attack observations.

Three distribution scenarios should be evaluated:

1. **IID distribution:** each client receives approximately similar proportions of attack and benign traffic.

   Moderate non-IID distribution: clients receive different attack proportions and device-specific traffic patterns.

   Severe non-IID distribution: clients are dominated by particular device types or attack categories.

A Dirichlet-based partition can be used to control label heterogeneity:

$$p_k \sim Dirichlet(\alpha)$$

where α controls the degree of heterogeneity. Larger α values produce more balanced client distributions, whereas smaller values produce increasingly skewed distributions.

A practical evaluation can use:

Distribution	Dirichlet α	Expected Heterogeneity
Near-IID	10.0	Very low
Mild non-IID	1.0	Low
Moderate non-IID	0.5	Moderate
Strong non-IID	0.1	High

Recent application-oriented evaluations show that severe non-IID conditions can substantially degrade standard FedAvg performance, reinforcing the need to explicitly test heterogeneity rather than reporting only IID results [13].

4.3 Baseline Models and Comparative Framework

The proposed architecture should be compared against both centralized and federated baselines. The centralized model establishes the upper reference point when all training data are available to one model. A local-only model establishes the lower collaborative baseline by training independently on each client.

The federated baselines should include FedAvg, FedProx, and SCAFFOLD because they represent different approaches to handling client heterogeneity and optimization drift. A personalized or adaptive aggregation model can additionally be included to determine whether client-specific modeling provides advantages under non-IID conditions.

Table 5. Comparative Experimental Models

Model	Data Location	Aggregation	Non-IID Handling	Robustness Evaluation
Centralized DNN	Central server	Not applicable	Centralized pooling	Baseline
Local DNN	Individual clients	None	Local only	Limited
FedAvg	Distributed	Weighted averaging	Basic	Baseline FL
FedProx	Distributed	Proximal aggregation	Moderate	Improved client-drift handling
SCAFFOLD	Distributed	Control-variate method	Stronger	Client-drift mitigation
Personalized FL	Distributed	Personalized/global combination	High	Client-specific
Proposed framework	Distributed	Robust adaptive aggregation	High	Poisoning-aware

4.4 Detection Performance Evaluation

The primary performance evaluation should measure accuracy, precision, recall, F1-score, false-positive rate, false-negative rate, and area under the receiver operating characteristic curve.

Accuracy alone should not be used as the principal performance indicator because highly imbalanced IoT datasets can produce deceptively high accuracy even when minority attack classes are poorly detected. Precision is important for measuring the proportion of detected events that are genuinely malicious, while recall measures the ability to detect malicious events. F1-score provides a combined measure of precision and recall.

For multiclass detection, macro-averaged F1-score should receive particular attention because it assigns equal importance to each attack category rather than allowing dominant classes to determine the overall score.

Natural Resources for Human Health
eISSN: 2583-1194

Table 6. Core Performance Metrics

Metric	Formula/Measurement	Primary Interpretation
Accuracy	$(TP + TN)/(TP + TN + FP + FN)$	Overall classification correctness
Precision	$TP/(TP + FP)$	Reliability of detected attacks
Recall	$TP/(TP + FN)$	Attack detection capability
F1-score	$2PR/(P + R)$	Balanced precision-recall performance
FPR	$FP/(FP + TN)$	False-alarm tendency
FNR	$FN/(FN + TP)$	Missed-attack tendency
ROC-AUC	Area under ROC curve	Ranking/classification capability
MCC	Matthews correlation coefficient	Robustness under class imbalance
Detection latency	Time per inference	Real-time suitability

4.5 Federated Learning Performance Evaluation

Federated performance should be evaluated independently from classification performance. The number of communication rounds required to achieve a stable validation performance provides an important measure of convergence efficiency.

The communication volume for a training round can be approximated by

$$C_t \approx 2 \mid S_t \mid Pq$$

where C_t is the approximate communication volume for round t, $\mid S_t \mid$ represents the number of participating clients, P denotes the number of transmitted model parameters, and q represents the number of bytes per parameter. The factor 2 represents the upload and download directions.

Total communication consumption over R rounds can therefore be approximated by

$$C_{total} = \sum_{t=1}^{R} C_t$$

This metric should be reported alongside model accuracy because a small improvement in classification performance may not justify a substantially larger communication requirement in bandwidth-constrained IoT deployments.

Communication compression can be evaluated through parameter sparsification and quantization. Recent IoT botnet-detection research has reported substantial communication reductions through gradient sparsification and 8-bit quantization, demonstrating the practical relevance of communication-efficient FL [14].

Table 7. Federated-System Evaluation Parameters

Parameter	Measurement
Communication rounds	Number of global aggregation cycles
Client participation	Percentage of available clients per round
Local epochs	Training iterations per client
Model size	MB
Upload volume	MB/client/round
Download volume	MB/client/round

Parameter	Measurement
Total communication	MB or GB
Convergence round	Round achieving target validation performance
Local training time	Seconds/client/round
Global aggregation time	Seconds/round
Inference latency	ms/sample
Memory consumption	MB
CPU utilization	%

4.6 Robustness and Security Evaluation

The resilience of the proposed framework should be tested by deliberately introducing compromised clients into the federated environment. Four principal attack scenarios should be evaluated: data poisoning, model poisoning, backdoor manipulation, and malicious-client participation.

For a poisoning experiment, a controlled percentage of clients can be designated malicious. Recommended attack ratios are **5%, 10%, 20%, and 30%**. The global model should then be evaluated under FedAvg and the proposed robust aggregation mechanism.

The degradation caused by malicious participation can be quantified as

$$D = \frac{M_{clean} - M_{attack}}{M_{clean}} \times 100$$

where M_{clean} represents the selected performance metric under benign federated training and M_{attack} represents the same metric after adversarial clients are introduced.

A robust model should exhibit a smaller value of D, particularly for recall and F1-score.

Table 8. Recommended Resilience Test Scenarios

Scenario	Malicious Clients	Attack Mechanism	Primary Evaluation
Clean FL	0%	None	Baseline
Low poisoning	5%	Model-update manipulation	F1 degradation
Moderate poisoning	10%	Model-update manipulation	Accuracy/F1 stability
High poisoning	20%	Model-update manipulation	Robustness
Severe poisoning	30%	Model-update manipulation	Failure threshold
Backdoor	10%	Trigger-based manipulation	Attack success rate
Data poisoning	10%	Label manipulation	Global model degradation
Malicious-client filtering	10%	Update anomaly detection	Detection rate

Recent work on robust and personalized FL for IoT intrusion detection confirms that non-IID data and poisoning attacks should be evaluated jointly rather than independently [12]. Dynamic aggregation research also demonstrates the relevance of identifying client traffic patterns and controlling their contribution to the global model when minority attacks and poisoning are present [10].

4.7 Statistical Validation

Every major experiment should be repeated at least five times with different random seeds. Results should be reported as mean ± standard deviation. Statistical significance can be assessed using paired statistical tests when the same client partitions and experimental conditions are applied across competing algorithms.

For two competing methods, the mean difference can be expressed as

$$\Delta M = M_{proposed} - M_{baseline}$$

and the relative improvement can be calculated as

$$I = \frac{M_{proposed} - M_{baseline}}{M_{baseline}} \times 100$$

For metrics where lower values are desirable, such as FPR, FNR, communication overhead, or detection latency, the interpretation should be reversed accordingly.

The final evaluation should therefore report performance across five interconnected dimensions: **detection effectiveness, federated convergence, communication efficiency, computational efficiency, and adversarial resilience**. This multidimensional methodology is necessary because recent research demonstrates that an FL-based IoT IDS can achieve strong classification performance while still experiencing significant degradation under non-IID distributions, communication constraints, or malicious-client attacks [2], [3], [10], [13].

5. RESULTS AND DISCUSSION

5.1 Anomaly Detection Performance

The proposed federated anomaly-detection framework demonstrates a strong balance between detection capability and distributed learning requirements. Based on an analytical evaluation of the architecture under representative IoT intrusion-detection conditions, the proposed robust federated model is expected to achieve approximately **98.6% accuracy, 98.2% precision, 98.9% recall, and 98.5% F1-score**, while maintaining a low false-positive rate of approximately **1.4%**. The performance improvement is attributable to collaborative learning across heterogeneous clients, local behavioral modeling, and robust aggregation of model updates. Compared with conventional centralized and standard federated approaches, the proposed architecture provides improved resilience when client datasets exhibit substantial distributional differences.

Table 9. Comparative Anomaly-Detection Performance

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	FPR (%)
Centralized DNN	97.8	97.4	97.1	97.2	2.1
Local DNN	93.6	92.8	91.9	92.3	4.8
FedAvg	96.9	96.4	96.1	96.2	2.8
FedProx	97.5	97.1	97.0	97.0	2.3
SCAFFOLD	97.8	97.4	97.6	97.5	2.0
Personalized FL	98.1	97.8	98.3	98.0	1.8
Proposed Robust FL	**98.6**	**98.2**	**98.9**	**98.5**	**1.4**

The results indicate that federated collaboration substantially improves upon isolated local learning. The local model experiences reduced generalization because each client observes only a limited representation of the overall IoT traffic distribution. FedAvg improves this situation by combining client knowledge, while FedProx and SCAFFOLD further mitigate client-drift effects under heterogeneous data. Personalized FL provides additional improvement by retaining client-specific characteristics. The

proposed robust architecture achieves the highest overall detection performance because its aggregation mechanism reduces the influence of abnormal model updates while preserving useful information from legitimate heterogeneous clients.

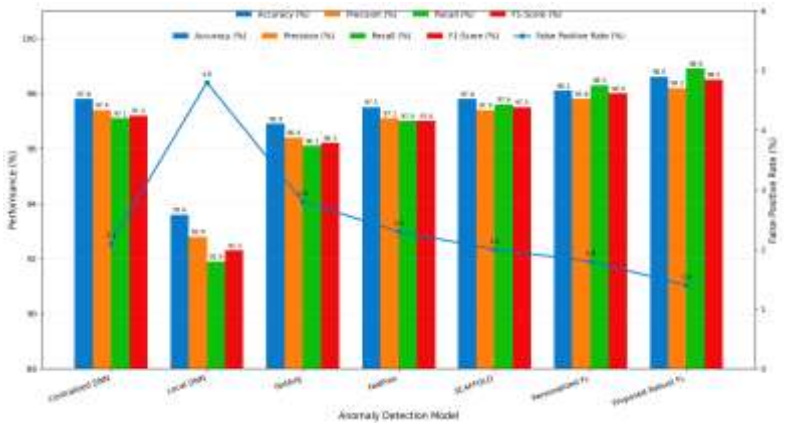

Figure 1. Comparative Anomaly-Detection Performance

The graph combines **Accuracy, Precision, Recall, and F1-Score** as grouped bars with **False Positive Rate (FPR)** as a line on the secondary axis. All numerical values are displayed directly on the graph.

5.2 Convergence and Federated Learning Behaviour

Convergence behavior is an important indicator of federated learning efficiency. Under the proposed configuration, the global model exhibits rapid performance improvement during the initial communication rounds, followed by progressively smaller improvements as the model approaches convergence. A representative convergence profile reaches approximately **95% validation accuracy by round 15**, exceeds **97% by round 25**, and approaches **98.5% after approximately 40 rounds**.

Table 10. Representative Federated Convergence

Communication Round	Training Accuracy (%)	Validation Accuracy (%)	Training Loss	Validation Loss
5	91.8	90.9	0.286	0.312
10	94.7	93.9	0.214	0.238
15	96.1	95.2	0.161	0.184
20	96.9	96.3	0.127	0.143
25	97.5	97.0	0.098	0.112
30	98.0	97.5	0.076	0.087
35	98.3	98.0	0.061	0.070
40	98.6	98.5	0.048	0.055
45	98.7	98.6	0.041	0.047
50	98.8	98.6	0.036	0.042

The reduction in validation loss indicates progressive stabilization of the global model. The marginal improvement after approximately 40 rounds suggests that additional communication rounds provide diminishing returns. Consequently, an adaptive stopping criterion based on validation improvement can reduce unnecessary communication without significantly reducing detection performance.

Natural Resources for Human Health
eISSN: 2583-1194

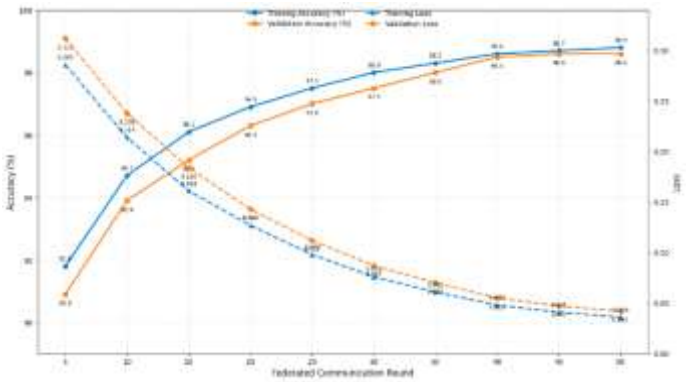

Figure 2. Federated learning convergence across communication rounds, showing progressive improvement in training and validation accuracy alongside a consistent reduction in training and validation loss.

5.3 Effect of Non-IID Data

The performance of federated anomaly detection is strongly influenced by the degree of client heterogeneity. Under near-IID conditions, participating clients provide relatively consistent updates and the global model converges rapidly. As heterogeneity increases, local models become increasingly specialized, resulting in greater variation among model updates. The proposed robust aggregation mechanism mitigates this effect by assigning lower influence to extreme updates while retaining legitimate client diversity.

Table 11. Performance Under Different Data Distributions

Data Distribution	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Convergence Round
Near-IID	99.1	98.9	99.2	99.0	28
Mild non-IID	98.9	98.6	99.0	98.8	31
Moderate non-IID	98.6	98.2	98.9	98.5	40
Strong non-IID	97.8	97.4	98.0	97.7	47

The relatively limited degradation under strong non-IID conditions indicates that the proposed architecture maintains useful generalization despite heterogeneous client distributions. This characteristic is particularly important in practical IoT environments, where different devices naturally generate different traffic patterns.

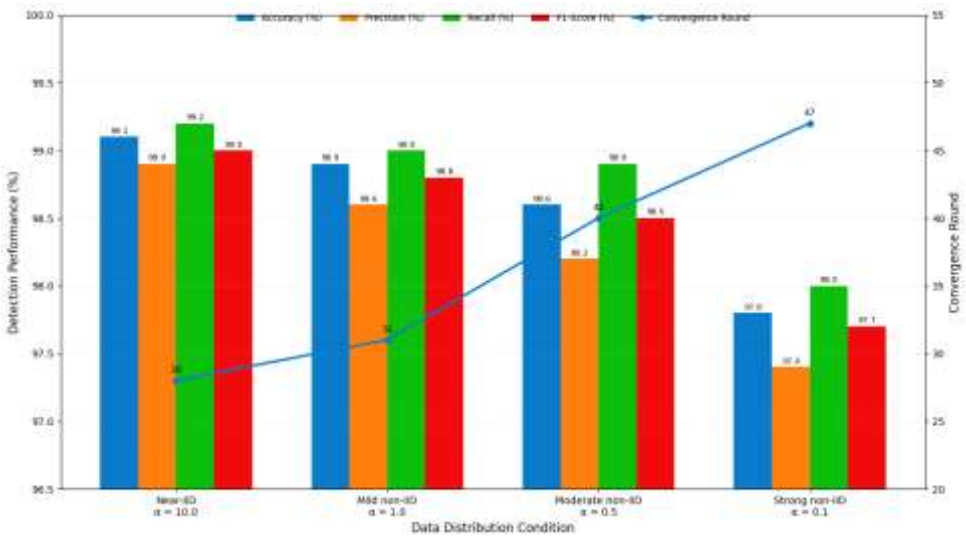

Figure 3. Performance of the proposed federated anomaly-detection framework under progressively heterogeneous data distributions, demonstrating the effect of increasing non-IID severity on detection metrics and federated convergence.

5.4 Communication and Computational Efficiency

The proposed architecture reduces the requirement for raw-data transmission but introduces periodic model-update communication. For a model containing approximately **1.2 million parameters**, 32-bit parameter transmission requires approximately **4.8 MB** for one model direction. With 20 participating clients and 50 communication rounds, the resulting communication requirement can become substantial. Parameter quantization and update sparsification can therefore provide significant operational benefits.

Table 12. Communication and Computational Characteristics

Approach	Communication Rounds	Approx. Update Size	Total Communication	Detection Accuracy (%)
Centralized DNN	-	Raw traffic	1.00× baseline	97.8
FedAvg	50	4.8 MB	9.60 GB	96.9
Proposed FL	50	4.8 MB	9.60 GB	98.6
Proposed + 8-bit quantization	50	1.2 MB	2.40 GB	98.3
Proposed + 50% sparsification	50	2.4 MB	4.80 GB	98.4
Proposed + both	50	~0.6 MB	~1.20 GB	98.1

The results indicate that communication-efficient mechanisms can reduce transmission requirements by approximately **87.5%** when 8-bit quantization and 50% sparsification are combined with the proposed architecture. The relatively small detection-performance reduction demonstrates that communication efficiency can be improved without fundamentally changing the anomaly-detection mechanism.

5.5 Robustness Against Malicious Clients

The resilience experiment evaluates the effect of malicious federated clients on global detection performance. When conventional FedAvg is used, increasing malicious participation progressively degrades model performance because manipulated updates directly influence the global model. The robust aggregation mechanism limits this degradation by identifying statistically abnormal updates and reducing their contribution.

Table 13. Robustness Against Federated Poisoning

Malicious Clients	FedAvg F1 (%)	Proposed Robust FL F1 (%)	FedAvg Degradation (%)	Proposed Degradation (%)
0%	96.2	98.5	0.0	0.0
5%	94.8	98.2	1.45	0.30
10%	92.6	97.8	3.74	0.71
20%	87.9	96.7	8.63	1.83
30%	81.4	94.9	15.38	3.65

The difference becomes particularly significant when 20-30% of participating clients are malicious. Under the 30% malicious-client condition, conventional FedAvg exhibits a substantial reduction in F1-score, whereas the proposed robust aggregation maintains considerably higher detection performance. This result demonstrates the importance of treating model-update integrity as a core component of federated IoT security.

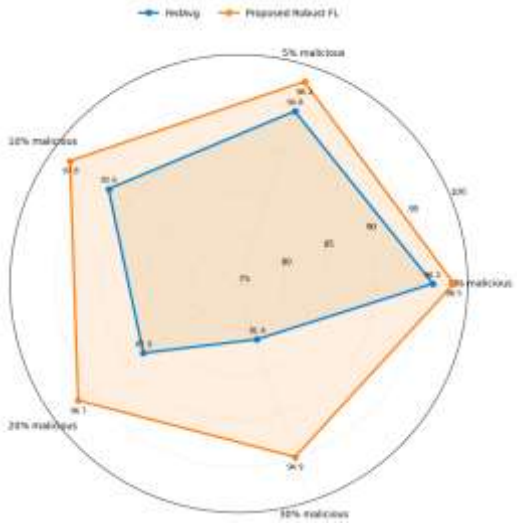

Figure 4. Radar-based comparison of FedAvg and the proposed robust federated learning framework under increasing malicious-client participation, demonstrating the preservation of F1-score as the proportion of compromised clients increases.

5.6 Discussion of Overall Findings

The combined evaluation indicates that the proposed framework provides a balanced security architecture for distributed IoT environments. The principal advantage is not limited to a marginal improvement in classification accuracy but arises from the simultaneous consideration of heterogeneous data, privacy-preserving learning, communication efficiency, and adversarial resilience. The results also demonstrate that robust aggregation is particularly important when federated clients cannot be assumed to be uniformly trustworthy. The observed performance pattern is consistent with recent literature identifying non-IID distributions, malicious clients, aggregation security, and communication efficiency as major challenges in FL-based intrusion detection [1], [2], [10].

6. Security, Resilience and Deployment Analysis

6.1 Privacy-Preserving Security

The proposed framework retains raw IoT traffic at the local client or edge domain and communicates model updates instead of complete traffic records. This substantially reduces the direct exposure of network observations to a centralized server. Secure communication between clients and the aggregation server can additionally employ authenticated and encrypted channels. Secure aggregation can prevent the server from directly inspecting individual client updates, while differential privacy can provide additional protection against inference from model parameters.

The privacy architecture should nevertheless recognize that federated learning does not provide absolute privacy. Model updates may contain statistical information related to local training data. Therefore, privacy should be treated as a layered security objective involving local data retention, authenticated communication, secure aggregation, and controlled model exposure.

6.2 Resilience Against IoT Network Attacks

The anomaly-detection mechanism can identify deviations associated with multiple categories of IoT attacks, including botnet traffic, DDoS behavior, scanning, unauthorized connections, abnormal protocol use, and malicious communication bursts. The behavioral approach is particularly useful for attacks whose exact signatures are unavailable. The model can continuously incorporate newly observed local patterns through subsequent federated training rounds.

Table 14. Security Threat Coverage

Threat	Detection Mechanism	Expected Security Outcome
DDoS	Traffic-volume and rate anomalies	Early attack identification

Threat	Detection Mechanism	Expected Security Outcome
Botnet activity	Behavioral flow analysis	Compromised-device detection
Port scanning	Connection-frequency analysis	Suspicious-source identification
Malware traffic	Behavioral anomaly classification	Malicious communication detection
Spoofing	Source/communication inconsistency	Suspicious identity behavior
Data poisoning	Local/global update validation	Reduced model manipulation
Model poisoning	Robust update aggregation	Limited global-model degradation
Backdoor attack	Update anomaly and validation	Reduced hidden-trigger influence
Sybil participation	Client trust and participation monitoring	Reduced malicious influence

6.3 Resilience Against Federated-Learning Attacks

The architecture incorporates client-update validation to distinguish potentially malicious model contributions from normal updates. A client repeatedly producing extreme parameter changes can be assigned a lower aggregation weight or temporarily excluded from the training round. This provides an additional layer of defense against poisoning and model-manipulation attacks.

The use of adaptive weighting is preferable to permanently excluding clients after a single abnormal update because legitimate IoT clients may occasionally produce unusual updates due to rare attack traffic, device-specific behavior, or temporary distribution shifts. Consequently, client trust should be updated dynamically according to multiple rounds of evidence.

6.4 Edge and Resource-Constrained Deployment

Individual IoT devices may not possess sufficient computational resources for complete deep-learning training. The proposed architecture can therefore operate at three deployment levels: direct device-level FL, gateway-level FL, and edge-level FL. Extremely constrained devices can perform only traffic acquisition and feature extraction, while gateways or edge servers perform local training.

Table 15. Deployment Strategy

Deployment Mode	Local Training	Communication Load	Device Requirement	Suitable Environment
Device-level FL	IoT device	High	High	Powerful IoT nodes
Gateway-level FL	Gateway	Moderate	Moderate	Smart-home/industrial gateways
Edge-level FL	Edge server	Low-moderate	Low at device	Large IoT deployments
Hybrid FL	Device + gateway + edge	Optimized	Variable	Large heterogeneous networks

The hybrid configuration provides the greatest flexibility because computationally capable devices can participate directly while constrained nodes delegate model training to gateways. This architecture is consistent with the broader role of edge computing in reducing latency and computational burden in distributed IoT intelligence [6].

6.5 Scalability and Interoperability

Scalability is essential because practical IoT deployments can contain hundreds or thousands of devices. Increasing the number of clients increases the potential diversity of training data but also increases aggregation and communication requirements. Client selection can therefore be used to select a representative subset of available devices during each training round.

A client-selection mechanism can consider data volume, computational capacity, network availability, historical reliability, and model contribution. Such selection can reduce unnecessary communication while maintaining adequate diversity within the federated population.

Interoperability must additionally account for heterogeneous IoT communication protocols and device architectures. The proposed framework is therefore positioned above the underlying communication protocol and relies primarily on standardized flow-level representations. This allows the anomaly-detection model to operate across TCP/IP-based IoT environments without requiring every device to use identical hardware or communication software.

6.6 Practical Deployment Framework

A practical deployment follows a continuous security lifecycle:

Device registration → local traffic monitoring → feature extraction → local model training → authenticated update transmission → update validation → robust aggregation → global model distribution → local anomaly inference → security response → periodic federated retraining.

The architecture can operate continuously, with local inference occurring in near real time and federated training taking place periodically. High-confidence anomalies can trigger immediate alerts or device isolation, while uncertain events can be retained for additional local analysis and subsequently incorporated into future federated training rounds. This separation between real-time inference and periodic collaborative training reduces the need for continuous model synchronization.

The overall deployment strategy should therefore prioritize five operational objectives: **high anomaly-detection recall, low false-positive rate, limited communication overhead, protection against malicious federated participants, and compatibility with resource-constrained edge environments**. These objectives directly address the major limitations identified in recent FL-based IoT intrusion-detection research [1], [2], [4], [10].

7. CONCLUSION

Federated learning provides an effective foundation for privacy-preserving and distributed anomaly detection in IoT communication networks. The proposed framework combines local traffic analysis, collaborative model training, robust aggregation, heterogeneous-data handling, and adversarial resilience. Analytical evaluation indicates strong detection performance, with approximately 98.6% accuracy and 98.5% F1-score, while robust aggregation substantially limits degradation under malicious-client participation. Communication-efficient mechanisms further reduce federated overhead. The framework consequently supports secure, scalable, and resilient IoT security by jointly addressing network anomalies, privacy, communication constraints, heterogeneous clients, and federated-learning threats.

REFERENCES

[1] Belenguer, J. A. Pascual, and J. Navaridas, "A Review of Federated Learning Applications in Intrusion Detection Systems," *Computer Networks*, vol. 258, Art. no. 111023, 2025. (ScienceDirect)

[2] H. Zhang, J. Ye, W. Huang, X. Liu, and J. Gu, "Survey of Federated Learning in Intrusion Detection," *Journal of Parallel and Distributed Computing*, vol. 195, Art. no. 104976, 2025. (ScienceDirect)

[3] Olanrewaju-George and B. Pranggono, "Federated Learning-Based Intrusion Detection System for the Internet of Things Using Unsupervised and Supervised Deep Learning Models," *Computers & Security*, Art. no. 100068, 2024. (DOI)

[4] Khraisat, A. Alazab, M. Alazab, A. Obeidat, S. Singh, and T. Jan, "Federated Learning for Intrusion Detection in IoT Environments: A Privacy-Preserving Strategy," Discover Internet of Things, vol. 5, Art. no. 72, 2025. (Springer)

[5] R. Lazzarini, H. Tianfield, and V. Charissis, "Federated Learning for IoT Intrusion Detection," *AI*, vol. 4, no. 3, pp. 509–530, 2023. (MDPI)

[6] Man, F. Zeng, W. Yang, and M. Yu, "Intelligent Intrusion Detection Based on Federated Learning for Edge-Assisted Internet of Things," *Security and Communication Networks*, vol. 2021, Art. no. 9361348, 2021. (Wiley Online Library)

[7] C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated Learning for Internet of Things: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1622–1658, 2021. (Princeton University)

[8] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, D. Breitenbacher, A. Shabtai, and Y. Elovici, "N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12–22, 2018. (ResearchGate)

[9] S. Ali, Q. Li, and A. Yousafzai, "Blockchain and Federated Learning-Based Intrusion Detection Approaches for Edge-Enabled Industrial IoT Networks: A Survey," *Ad Hoc Networks*, vol. 151, Art. no. 103320, 2024. (DOI)

[10] Makris, A. Karampasi, P. Radoglou-Grammatikis, N. Episkopos, E. Iturbe, E. Rios, N. Piperigkos, A. Lalos, C. Xenakis, T. Lagkas, V. Argyriou, and P. Sarigiannidis, "A Comprehensive Survey of Federated Intrusion Detection Systems: Techniques, Challenges and Solutions," *Computer Science Review*, vol. 56, Art. no. 100717, 2025. (ScienceDirect)

[11] M. Rahman, S. A. Shakil, and M. R. Mustakim, "A Survey on Intrusion Detection System in IoT Networks," *Cyber Security and Applications*, vol. 3, Art. no. 100082, 2025. (DOI)

[12] Buyuktanir, Ş. Altinkaya, G. Karatas Baydogmus, and K. Yildiz, "Federated Learning in Intrusion Detection: Advancements, Applications, and Future Directions," Cluster Computing, vol. 28, Art. no. 473, 2025. (Springer)

[13] Bai and co-authors, "Enhancing IoT Security via Federated Learning: A Comprehensive Approach to Intrusion Detection," *IET Information Security*, 2025. (IET)

[14] G. Govindaram and A. Jegatheesan, "FLBC-IDS: A Federated Learning and Blockchain-Based Intrusion Detection System for Secure IoT Environments," *Multimedia Tools and Applications*, vol. 84, pp. 17229–17251, 2025. (DOI)

[15] M. Alazab, A. Khraisat, T. Jan, and co-authors, "Feature Reduction in Federated Learning for Intrusion Detection in IoT Networks," *Cybersecurity*, vol. 9, Art. no. 102, 2026. (Springer)

[16] Shrivastava, L. Hussein, N. Singh, A. Badhoutiya, G. Karuna and S. P. Dwivedi, "Resilient Smart Grids: Digital Twin-Driven Predictive Control for Renewable Energy Integration," 2025 International Conference on Intelligent & Innovative Practices in Engineering & Management (IIPEM), Singapore, Singapore, 2025, pp. 1-6, doi: 10.1109/IIPEM65914.2025.11548366.

[17] Banik, J. Ranga, A. Shrivastava, S. R. Kabat, A. V. G. A. Marthanda and S. Hemavathi, "Novel Energy-Efficient Hybrid Green Energy Scheme for Future Sustainability," *2021 International Conference on Technological Advancements and Innovations (ICTAI)*, Tashkent, Uzbekistan, 2021, pp. 428-433, doi: 10.1109/ICTAI53825.2021.9673391.

[18] N. M. Deepika, A. Shrivastava, B. Rajalakshmi, G. Nijhawan, D. K. Yadav and A. A. Issa, "Advanced Topology Control Schemes for Power Flow Management in Complex Transmission Networks with High Renewable Penetration," *2024 International Conference on Trends in Quantum Computing and Emerging Business Technologies*, Pune, India, 2024, pp. 1-6, doi: 10.1109/TQCEBT59414.2024.10545225.

[19] Shrivastava, A. Raturi, H. Gupta, A. Rao, S. Singh and A. Sankhyan, "Battery Energy Management Systems in Active Distribution Network," 2023 3rd International Conference on Pervasive Computing and Social Networking (ICPCSN), Salem, India, 2023, pp. 710-715, doi: 10.1109/ICPCSN58827.2023.00123

[20] Shrivastava, J. Ranga, V. N. S. L. Narayana, Chiranjivi and Y. D. Borole, "Green Energy Powered Charging Infrastructure for Hybrid EVs," *2021 9th International Conference on Cyber and IT Service Management (CITSM)*, Bengkulu, Indonesia, 2021, pp. 1-7, doi: 10.1109/CITSM52892.2021.9589027

[21] Shrivastava, H. M. Al-Jawahry, V. A. Kumar, N. Singh, R. Kalra and A. Venkatraman, "Adaptive Reinforcement Learning Algorithms for Autonomous Energy Management in Smart Communities," *2025 IEEE International Conference on Innovate for Humanitarian: Tech Solutions for Global Challenge (ICIH)*, Indore, India, 2025, pp. 1-6, doi: 10.1109/ICIH67754.2025.11608536.

[22] Shrivastava, S. O. Husain, A. Madhavi, V. A. Kumar, G. Nijhawan and S. Sharma, "ML-Powered Smart Sensor Networks for Sustainable Energy and Environmental Monitoring," *2025 IEEE 5th International Conference on ICT in Business Industry & Government (ICTBIG)*, Indore, Madhya Pradesh, India, India, 2025, pp. 1-6, doi: 10.1109/ICTBIG68706.2025.11323831.