

AF-IoTS: An Adaptive Trust-Aware Federated Architecture for Scalable Intrusion Detection in Heterogeneous IoT Environments

Hawraa Imad Al-Abedi¹ 

Department of Forensic Evidence, Faculty of Medical Sciences, Jabir Ibn Hayyan University for Medical and Pharmaceutical Sciences, Najaf, 54001, Iraq.

hawraa.i.naji@jmu.edu.iq

Abstract

The Internet of Things (IoT) generates massive, privacy-sensitive traffic across heterogeneous, resource-constrained devices, making centralized intrusion detection systems (IDS) increasingly impractical due to scalability, latency, and privacy limitations. Existing federated learning (FL) based IDS solutions partially mitigate these issues but typically assume static training schedules, uniform participation, and reliable clients, and therefore remain vulnerable to non-IID data distributions, device heterogeneity, and adversarial behaviors such as poisoning and Byzantine attacks. This paper proposes Adaptive Federated IoT Security (AF-IoTS). This architectural framework integrates federated learning with three coordinated mechanisms: (i) a trust model that combines behavioral deviation, loss, uncertainty, availability, and resource ratio into an exponential moving-average trust score; (ii) a diversity-aware client-selection utility that jointly considers Trust, resources, and data diversity under communication and cardinality budgets; and (iii) a trust-weighted robust aggregation rule with update clipping, driven by a closed-loop adaptive controller that regulates the participation rate. The proposed workflow is formalized through explicit mathematical formulations and an end-to-end algorithm, and its behavior is analyzed qualitatively across smart cities, smart healthcare, and industrial IoT scenarios. Expected outcomes include preserving data locality, improving robustness against unreliable and malicious clients, and providing a coherent architectural basis for future large-scale empirical validation on standard IoT security datasets.

Keywords: Adaptive security architecture; Distributed security architecture; Edge intelligence; Federated learning for security; Security orchestration.

Article history: Received: 27 Jul 2026, Accepted: 3 Sep 2026, Published: 15 Sep 2026.

This article is open-access under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Internet of Things (IoT) is a growing field in which applications use billions of connected devices. IoT devices are now widely used in smart cities, hospitals, industrial settings and many more applications [1]. This expansion has brought connectivity and intelligent services to things and devices; however, it has also brought many security challenges to the fore due to device heterogeneity, large-scale deployment, and dynamic operating environments [2].

Many current network security solutions still rely on traditional IDS. However, they are

increasingly not applicable to large-scale IoT environments. Centralized IDS architectures have scalability limitations and high communication overhead, making them poorly suited to resource-constrained IoT environments [3, 4, 17]. In addition, collating data in a centralized repository creates significant privacy issues, especially when IoT devices generate sensitive data.

Federated learning (FL) is a promising scheme for collaboratively training models without exchanging raw data, ensuring privacy and reducing communication costs. Therefore, to meet decentralization and privacy requirements, FL has been widely used for IoT intrusion detection.

Nevertheless, recent research shows that most existing FL-based IDS solutions use static learning settings and fixed aggregation strategies, which can limit their effectiveness in dynamic IoT environments [8] and [19].

In addition, current FL-based IDS architectures generally lack trust-aware coordination mechanisms for untrustworthy or malicious actors. They can be vulnerable to poisoning and Byzantine attacks [9]. In addition, real-world IoT deployments [10] may suffer reduced detection accuracy and system robustness due to non-IID data distributions and heterogeneous device capabilities. In the next generation of IoT for 6G networks, ultra-dense connectivity, mobility and low-latency requirements will impose greater challenges [11].

Motivated by these challenges, we propose adaptive federated IoT security (AF-IoTS), an adaptive, privacy-preserving security framework that integrates federated learning with dynamic control mechanisms, trust-aware coordination, and cross-layer security orchestration. The proposed framework is designed to operate in a heterogeneous, evolving environment, providing scalable intrusion detection that preserves data privacy and remains resilient under adversarial conditions [12].

The novelty of AF-IoTS is not claimed for any single learning model, but for the integration of four coupled mechanisms into one federated architecture:

1. A behavior- and metadata-driven trust model with temporal memory,
2. A diversity-aware utility function that jointly weighs resources and non-IID data heterogeneity,
3. A trust-weighted robust aggregation rule with update clipping.
4. A closed-loop adaptive controller that regulates client participation from global performance feedback.

Prior FL-based IDS designs address these concerns in isolation; AF-IoTS formulates them jointly and links them through explicit mathematical relations and a single end-to-end workflow, which distinguishes this contribution from combining existing methods.

2. Related work

Recent studies on IoT security show that intrusion detection systems (IDS) have attracted attention as a primary defence against cyber-attacks because they operate in large-scale, distributed systems [3]. IDS solutions identify harmful actions by monitoring network traffic and system behaviour. They are especially relevant to IoT deployments that continuously generate data and are connected [13, 14]. Despite their effectiveness, traditional IDS are mainly centralized and assume relatively stable network states, which limits their applicability in heterogeneous and dynamic IoT systems.

Several studies have explored distributed and edge-based IDS solutions for IoT environments to address the limitations of centralized architectures. Bringing detection closer to data sources can reduce latency and improve scalability [28, 32]. Nevertheless, limited interconnectivity means many edge-based IDS solutions share only partial data or make isolated local decisions, limiting their ability to achieve global situational awareness (GSA) against complex and evolving attack scenarios [16].

Federated learning (FL) enables distributed training of shared models on clients that collaborate to detect intrusions while keeping raw data private [5], [6], [7], and [36]. Newly developed FL-based IDS frameworks have shown strong detection performance while protecting privacy in IoT environments [18], [34]. Nonetheless, numerous IDS solutions that are based on FL continue to utilize fixed training schedules and aggregation strategies, which limits their applicability in highly dynamic IoT environments [19].

In addition to this adaptivity issue, multiple studies have cited the vulnerability of FL-based IDS systems to unreliable or malicious participants. In adversarial scenarios, poisoned updates or Byzantine behavior from compromised clients can greatly undermine the quality of the global model [20]. The intrusion detection signature (IDS) frameworks that rely on FL are still susceptible to such attacks in the absence of any explicit trust-aware coordination mechanisms. This is especially true for open, large-scale Internet of Things (IoT) deployments [21].

The non-IID characteristics of IoT data present a core problem for federated intrusion detection, as does adversarial behavior. Traffic characteristics, attack patterns, and normal behaviour profiles usually differ widely across devices, locations, and application domains. This prevents convergence and lowers detection accuracy [22]. The systems' inconsistencies, including varying compute capability, energy availability, and network connectivity on IoT and edge nodes, render data-handling issues more difficult [25]. These limitations become even more critical in next-generation Internet of Things (IoT) systems foreseen for 6G networks, with ultra-dense connectivity, mobility, and stricter latency requirements, which further complicate them and broaden the attack surface [11] and [23].

To summarize the methods and reported outcomes of the works reviewed above: [3, 14] employ deep learning classifiers on centralized datasets and report high detection accuracy but assume raw-data aggregation, which is not privacy-preserving. [18] applies standard FL with FedAvg on IoT security datasets and demonstrates that collaborative training preserves data locality, while the survey in [7] highlights that most FL-based IoT solutions still rely on fixed schedules and uniform client weighting, which degrade under non-IID conditions. [20], [21], [26] and [27] focus on Byzantine and poisoning defenses (e.g., FLTrust, FreqFed, precision-guided filtering) and report improved robustness against a specific attacker model. Still, they treat aggregation robustness in isolation from client heterogeneity and participation control. [24, 25] address client scheduling and heterogeneity-aware training and show convergence improvements on non-IID splits. However, they do not incorporate a trust component.

The techniques and results are therefore complementary rather than substitutable, and this observation motivates the integrated framework proposed in the following sections. While federated learning-based intrusion detection and robust learning have made much progress, the literature mostly treats learning, robustness, and response coordination as separate design concerns. Creating a single adaptable security structure that

accommodates non-IID data characteristics, along with trust-aware aggregation and cross-layer orchestration for 6G-oriented IoT, remains an open research avenue. This gap motivates the proposed Adaptive Federated IoT Security (AF-IoTS) Framework.

3. Problem Statement and Design Objectives

3.1 Problem Statement

IoT intrusion detection must operate under severely distributed, heterogeneous, and resource-constrained conditions in which centralized monitoring and data pooling are usually impractical or privacy-sensitive [14]. Federated Learning (FL) eliminates the need to share raw IoT data. Nevertheless, many FL-IDS designs still assume relatively stable training settings that do not align with real IoT environments [19].

In real-world scenarios, the deployment of the Internet of Things (IoT) often features statistical non-uniformity (non-IID data) [20]. This is evident because local traffic patterns and attack manifestations often vary across devices and sites [21]. This data non-uniformity can adversely affect the global model's convergence and accuracy [22]. Diversity in the network's computation, battery, and other conditions, such as participation and training efficiency, adds to the non-IID problem [25]. Federated Learning-based security systems can also be vulnerable to malicious behavior such as local model poisoning and Byzantine-style attacks that can distort the aggregation process and weaken detection robustness [20].

Current studies show that strong aggregation, anomaly filtering, or validation mechanisms, often adopted as effective poisoning defenses, can withstand changing attacker strategies and data. The main research problem of the study was the lack of an integrated adaptive framework to address the following in this paper jointly:

1. Dynamic IoT conditions.
2. Multi-dimensional heterogeneity.
3. Adversarial robustness in federated intrusion detection [20] and [26]

3.2 Design Objectives

To address this problem, AF-IoTS is designed with the following objectives [12].

- 1-Privacy-Preserving Collaborative Detection: Support collaborative intrusion detection without disclosing raw IoT data, while achieving competitive detection performance [7, 18].
2. Dynamic Adaptation: Be effective under changing traffic, mobility, and threat patterns by supporting dynamic adjustment of training and coordination (e.g., schedules, participation, aggregation) [11, 19].
3. Robustness Against Poisoning and Byzantine Behaviors: Deploy robust aggregation and trust-aware defenses to alleviate the impact of malicious or unreliable clients on the global model [20, 26, 27].
4. Heterogeneity-aware efficiency: Use client scheduling and resource-aware coordination [25] to improve training efficiency and stability on heterogeneous IoT-edge resources.
5. 6G-Readiness and Scalability: The framework should scale to ultra-dense IoT deployments and meet the low-latency requirements envisioned in 6G [11] and [12].

These objectives guide the architectural decisions, described in Section 4, that translate the identified requirements into a layered and adaptive federated security design.

3.3 Scope and Assumptions

AF-IoTS targets IoT environments where clients train local detection models at device/edge nodes and synchronize updates to an aggregator periodically while maintaining data locality [7] and [22]. The framework assumes clients may be unreliable or adversarial. It must therefore be handled through robust, trust-aware coordination rather than naive aggregation [20, 26]. The paper describes an architecture-level solution and design objectives relevant to 6G-era IoT, which serves as a basis for later algorithmic instantiations and empirical evaluation [11] and [12].

4. Proposed AF-IoTS Architecture

4.1 Architectural Design Rationale

The proposed Adaptive Federated IoT Security (AF-IoTS) architecture is designed to secure large-scale IoT environments characterized by heterogeneity, decentralization, and dynamic threat conditions.

AF-IoTS is designed to be resilient to single points of failure and to enable distributed intelligence near data sources. Moreover, next-generation IoT systems should be equipped with security mechanisms that are adaptive by design rather than static configurations that deteriorate under changing conditions. In this respect, AF-IoTS presents a multi-layer, decentralized architecture involving federated learning with adaptive control and trust-aware orchestration across the IoT, edge, and orchestration layers.

4.2 High-Level Architecture Overview

The proposed AF-IoTS architecture includes four interconnected layers:

1. IoT Device Layer.
2. Edge Intelligence Layer.
3. Federated Coordination Layer.
4. Security Orchestration and Analytics Layer.

This hierarchical organization separates statistical computations, enabling fast intrusion detection when required. This architectural decomposition is a critical enabler, providing scalable and resilient security in distributed IoT and edge computing environments. Illustration: Fig. 1 illustrates the high-level architecture of the suggested AF-IoTS framework, along with the interaction of data flow and model flow. Feedback loops and closed-loop adaptive control systems help manage stakeholder engagement, aggregation, and response facilitation. The architecture has four layers, as shown in Fig. 1; these layers work together and perform different tasks for learning and security.

Fig. 1 illustrates the layered architecture of AF-IoTS. From bottom to top: the IoT Device Layer performs local feature extraction and local model training on constrained devices; the Edge Intelligence Layer aggregates contextual signals from nearby devices, hosts adaptive-control decisions, and mediates communication with the coordinator; the Federated Coordination Layer runs trust-aware client selection and trust-weighted robust aggregation; and the Security Orchestration and Analytics Layer correlates alerts across layers and issues coordinated response policies. The vertical arrows represent trust and performance feedback flowing upward. In contrast, policy and

model updates flow downward, forming the closed-loop control referenced in Section 5.1.

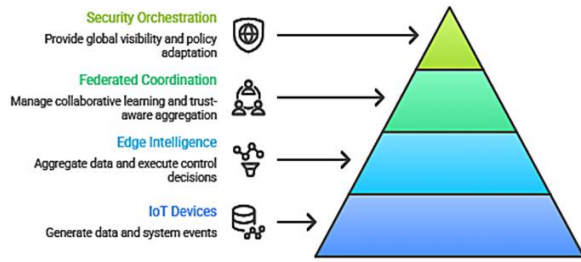


Fig. 1 Hierarchical architecture of the proposed AF-IoTS.

4.3 IoT Devices Layer

The devices in the device layer include various devices with heterogeneous sensing and actuating capabilities that generate network traffic and system events. Because of privacy limitations and scarce resources, raw data are processed locally, where lightweight intrusion detection components extract features and train local models. In modern IoT deployments, training locally on the device or edge is essential to reduce communication overhead and keep valuable data private.

4.4 Edge Intelligence Layer

The Edge Intelligence Layer mediates between IoT devices and other high-level coordination components. Edge nodes gather contextual information, track local network dynamics, and make adaptation control decisions to influence participation and response actions. The Edge Intelligence Layer in AF-IoTS reduces detection latency and enhances response in dynamic IoT environments through localized processing.

4.5 Federated Coordination Layer

Fig. 1 shows that trust-aware aggregation is driven by feedback collected across layers to minimize the impact of unreliable or malicious participants. The Federated Coordination Layer orchestrates cooperative learning by aggregating model updates from distributed clients. Unlike classical FL-based IDS frameworks that use static aggregation, AF-IoTS employs trust-aware aggregation rules that dynamically weight clients' contributions based on reliability and consistency metrics. This approach mitigates the impact of poisoning and Byzantine

attacks, which are considered the main threats for federated learning systems.

4.6 Security Orchestration and Analytics Layer

The Security Orchestration and Analytics Layer provide global visibility and long-term capabilities to manage security. This layer correlates alerts, learning feedback, and contextual information from all layers to support policy adaptation and coordinated response strategies. In AF-IoTS, centralized orchestration combined with decentralized detection enhances situational awareness while avoiding the disadvantages of fully centralized security architectures.

4.7 Architectural Advantages

Unlike existing IoT security architectures that treat adaptivity and trust-awareness as standalone components, AF-IoTS integrates them across all layers. This cross-layer integration improves robustness, scalability and Readiness for next-generation IoT and 6G environments.

5. Adaptive Mechanisms and Federated Security Workflow

5.1 Adaptive Control Loop

Operationalization of adaptivity through closed-loop control, which continuously observes the security feedback, learning performance and environmental context. These control loops enable dynamic tuning of system parameters, including client participation, training frequency, and aggregation strategies, which have been found crucial for performance retention in dynamic IoT settings. The feedback from detection outcomes and network status allows the adaptive control loop within AF-IoTS to counter evolving threats rather than static solution configurations.

5.2 Mathematical Formulation of Trust, Diversity-Aware Selection, and Adaptive Aggregation.

In this section, we formally formulate the adaptive mechanisms proposed in AF-IoTS, which include trust assessment and the adaptive control loop. This formulation converts the architectural design into a formal system model that can be used for analysis without sharing raw data.

5.2.1 Client Metrics and Update Representation.

Note on equation numbering. The equations of 5.2 are labeled sequentially (1) through (10) in reading order across the subsections 5.2.1–5.2.7. In-text references to a specific equation use its number; groups of related equations are referenced by their subsection number (e.g., "the equations of 5.2.3").

Suppose c is a set of clients. In the federated round t , each client $i \in c$ transmits the local model update:

$$\Delta i(t) = w_i(t) - w_i(t) \quad (1)$$

Along with a set of non-sensitive indicators (metadata):

$$m_i(t) = \{a_i(t), r_i(t), l_i(t), u_i(t)\}$$

where:

- $a_i(t) \in [0,1]$: Availability and commitment-to-participate rating
- $r_i(t) \in [0,1]$: Normalized computing/energy resources ratio.
- $l_i(t) \geq 0$: Loss value after local training
- $u_i(t) \geq 0$: Local uncertainty or anomaly score (if any) degree.

5.2.2 Update Deviation and Behavioral Risk

The server computes the average of the updates to check the consistency of the client's update with the collective behavior:

$$\bar{\Delta}(t) = \frac{1}{|S_t|} \sum_{j \in S_t} \Delta_j(t) \quad (2)$$

where S_t is the set of clients participating.

The update deviation of client i is defined as:

$$\rho_i(t) = \frac{||\Delta_i(t) - \bar{\Delta}(t)||}{||\bar{\Delta}(t)|| + \epsilon} \quad (3)$$

With $\epsilon > 0$ avoid division by zero.

$\rho_i(t)$ is a behavioral indicator for detecting anomalous or inconsistent updates.

5.2.3 Risk-Aware Trust Modeling with Temporal Memory

After normalizing the indicators to the domain $[0,1]$, we define the risk score for the client:

Where $\alpha_k \geq 0$ and with $\sum_k \alpha_k = 1$

Then, we define the instantaneous trust score:

$$T_i(t) = \exp(-R_i(t)) \quad (4)$$

To introduce temporal memory and feedback, we update Trust using an Exponential Moving Average (EMA).

$$T_i^{EMA}(t) = \lambda T_i^{EMA}(t-1) + (1 - \lambda) T_i(t) \quad (5)$$

Where:

$$0 < \lambda < 1$$

5.2.4 Diversity-Aware Utility Function.

In addition to trust, AF-IoTS addresses data diversity and contextual representation. The participation utility of client i is defined as:

Where:

$D_i(t)$ is the client's data diversity metric, e.g. variance of feature or class distribution with respect to the current global set.

$\gamma_1 + \gamma_2 = 1$, where $\gamma_1, \gamma_2 \geq 0$

This definition allows trading off computational resources with statistical diversity of heterogeneous (non-IID) data.

5.2.5 Trust- and Diversity-Aware Client Selection.

The selection of the clients in a certain round is formulated as a constrained utility maximization problem:

$$\max_{S_t \subseteq c} \sum_{i \in S_t} T_i^{EMA}(t) \cdot U_i(t) \quad (6)$$

Subject to the constraints:

$$\sum_{i \in S_t} c_i(t) \leq B(t), \quad |S_t| \leq K \quad (7)$$

where:

$c_i(t)$: Communication/ time cost.

$B(t)$: Available budget.

K : Maximum number of clients per round.

5.2.6 Trust- and Diversity-Aware Robust Aggregation

The global model is then updated by trust-weighted aggregation of the selected clients:

$$\omega(t+1) = \omega(t) + \sum_{i \in S_t} \omega_i(t) \Delta_i^{clip}(t) \quad (8)$$

Where:

$$\omega_i(t) = \frac{T_i^{EMA}(t) n_i}{\sum_{j \in S_t} T_j^{EMA}(t) n_j} \quad (9)$$

The update is clipped to limit the effect of anomalous updates:

$$\Delta_i^{clip}(t) = \Delta_i(t) \cdot \min\left(1, \frac{T}{||\Delta_i(t)||_2}\right) \quad (10)$$

5.2.7 Adaptive Control Loop

Finally, the dynamics of the participation rate $\pi(t)$ are adapted according to a global performance indicator $\rho(t)$. This results in a closed control loop that can adapt to changing conditions and threats.

5.2.8 Parameter Selection Strategy

The parameters of the above formulations are not fixed by convention; they are chosen through the following strategy so that the framework remains reproducible and practically deployable. The risk-weight vector ($\alpha_1.. \alpha_5$) is normalized to sum to one and initialized uniformly ($\alpha_k = 0.2$); the weights are then re-tuned per deployment domain (smart city, healthcare, industrial IoT) using grid search on a small held-out validation subset, guided by the relative importance of loss stability versus resource availability in that domain.

The EMA smoothing factor $\lambda \in (0,1)$ is selected in $[0.7, 0.9]$: higher values give stronger temporal memory (more conservative reaction to short-term anomalies) and lower values give faster adaptation; a default of $\lambda = 0.8$ is used unless the environment is highly non-stationary. The utility weights (γ_1, γ_2) sum to one and default to $(0.4, 0.6)$, placing slightly more weight on data diversity than on resources when the deployment is known to be non-IID. The clipping threshold τ is set to the median L2-norm of clean updates observed in the first few rounds. The controller gain η , the participation-rate bounds (π_{min}, π_{max}), and the round budgets ($B(t), K$) are chosen from the communication and energy budget of the target deployment rather than from the model. This strategy makes each parameter a design choice tied to observable system quantities rather than an arbitrary constant.

5.3 Algorithmic Workflow of the AF-IoTS Framework

Algorithm 1. AF-IoTS: Adaptive Federated Security Pipeline with Trust and Diversity
 Algorithm 1 summarizes the end-to-end workflow of the AF-IoTS framework for operationalizing the proposed adaptive mechanisms. The algorithm formalizes the interaction among trust evaluation, diversity-aware client selection, robust aggregation, and adaptive control, as described in the mathematical formulation above. It illustrates how

Trust feedback and control loops are embedded into the federated learning procedure without sharing raw data.

Algorithm 1: Trust- and Diversity-Aware Adaptive Federated Security Workflow (AF-IoTS).

Input:

Global model $w(t)$, client set C

Budgets: $B(t)$ (communication/time), K (max clients)

Parameters: $\alpha_1.. \alpha_5, \gamma_1, \gamma_2, \lambda, \tau, \eta, \pi_{min}, \pi_{max}$

Performance metric $P(t)$ (server-side), $\epsilon > 0$

Output:

Updated global model $w(t+1)$, updated participation rate $\pi(t+1)$

// :1Step A: Determine candidate pool size using adaptive control

:2 $\pi(t+1) \leftarrow \text{clip}(\pi(t) + \eta \cdot (P(t) - P(t-1)), \pi_{min}, \pi_{max})$

:3CandidatePool $\leftarrow \text{sample_candidates}(C, \text{size} = \text{ceil}(\pi(t+1) \cdot |C|))$

// :4Step B: Collect local updates and metadata (no raw data)

:5for each client i in CandidatePool in parallel do

:6($\Delta_i(t), a_i(t), r_i(t), \ell_i(t), u_i(t), n_i, c_i(t), D_i(t)$) $\leftarrow \text{local_train_and_report}(w(t))$

:7end for

:8 $St \leftarrow \{i \in \text{CandidatePool} \mid \text{report received and } c_i(t) \text{ feasible}\}$

// :9Step C: Compute update deviation and instantaneous Trust

:10 $\bar{\Delta}(t) \leftarrow (1/|St|) \cdot \sum_{j \in St} \Delta_j(t)$

:11for each client i in St do

:12 $\rho_i(t) \leftarrow \|\Delta_i(t) - \bar{\Delta}(t)\|^2 / (\|\bar{\Delta}(t)\|^2 + \epsilon)$

:13Normalize metrics to $[0,1]$: ($\tilde{a}_i, \tilde{r}_i, \tilde{\ell}_i, \tilde{u}_i, \tilde{\rho}_i$)

:14 $R_i(t) \leftarrow \alpha_1 \cdot \tilde{\rho}_i + \alpha_2 \cdot \tilde{u}_i + \alpha_3 \cdot \tilde{\ell}_i + \alpha_4 \cdot (1 - \tilde{a}_i) + \alpha_5 \cdot (1 - \tilde{r}_i)$

:15 $T_i(t) \leftarrow \exp(-R_i(t))$

:16 $T_i\text{EMA}(t) \leftarrow \lambda \cdot T_i\text{EMA}(t-1) + (1-\lambda) \cdot T_i(t)$

:17 $U_i(t) \leftarrow \gamma_1 \cdot \tilde{r}_i + \gamma_2 \cdot D_i(t)$

:18 $\text{Score}_i(t) \leftarrow T_i\text{EMA}(t) \cdot U_i(t)$

:19end for

// :20Step D: Select trusted & diverse clients under budgets

:21 $S^*t \leftarrow \text{argmax}_{\{S \subseteq St\}} \sum_{i \in S} \text{Score}_i(t)$

s.t. $\sum_{i \in S} c_i(t) \leq B(t)$ and $|S| \leq K$

// :22Implementation can be greedy by $\text{Score}_i(t)$ with budget checks

// :23Step E: Trust-aware robust aggregation

:24for each client i in S^*t do

:25 $\Delta_{i_clip}(t) \leftarrow \Delta_i(t) \cdot \min(1, \tau / \Delta_i(t)^2)$

:26end for

```

:27 $\omega_i(t) \leftarrow (T_i \text{EMA}(t) \cdot n_i) / \sum_{j \in S^*} (T_j \text{EMA}(t) \cdot n_j)$ 
:28 $w(t+1) \leftarrow w(t) + \sum_{i \in S^*} \omega_i(t) \cdot \Delta_i\_clip(t)$ 
:29return  $w(t+1), \pi(t+1)$ 
    
```

AF-IoTS Algorithm 1 provides the workflow, and the participation ratio is adaptively determined according to the global performance feedback. Candidate clients are then ranked by trustworthiness and diversity, using local metadata and update behavior. Selected based on communication constraints, the trusted clients' updates are aggregated by trust-aware robust weighting. The framework's closed-loop process allows it to adapt $\square$ dynamically to changing IoT conditions and adversarial phenomena.

Computational Complexity of Algorithm 1

Let N be the size of the candidate pool selected in each round, and d be the dimensionality of the model-update vector. $\Delta_i(t)$ Moreover, let K be the maximum number of clients admitted per round. In a single federated round:

1. Computing the mean update $\bar{\Delta}(t)$ is $O(Nd)$.
2. Computing the deviation $\rho_i(t)$, the normalized risk $R_i(t)$, the instantaneous Trust $T_i(t)$, the EMA trust $T_i^{EMA}(t)$, the utility $U_i(t)$, and the score for every candidate is $O(N)$ after the mean is available.
3. Greedy Trust- and diversity-aware selection under budget constraints is $O(N \log N)$ when implemented by sorting candidates by score and admitting until $B(t)$ Alternatively, K is reached.
4. Trust-weighted robust aggregation over the selected set is $O(Kd)$. The dominant cost per round is therefore $O(Nd + N \log N + Kd)$ at the coordinator, which reduces to $O(Nd)$ whenever the model dimension d dominates. On the client side, local training preserves the underlying model cost. It adds only $O(d)$ for the reported update. The coordinator's memory footprint is $O(Nd)$ for the candidate updates plus $O(N)$ for the trust vector. T_i^{EMA} . Communication cost per round is $O(Kd)$ transmitted and $O(Kd)$ received. These costs are linear in the pool size and the model dimension, which is compatible with the resource-constrained, high-density regime targeted by the framework.

5.4 Context-Aware Client Selection

In AF-IoTS, client selection is based on contextual indicators such as device reliability, resource availability, and recent security observations. This context-aware participation mitigates the effects of unstable or resource-constrained devices. It improves convergence stability and overall detection accuracy. Within the framework, intelligent client scheduling is a key mechanism for addressing the heterogeneity and efficiency challenges inherent to IoT-edge federated learning systems.

5.5 Trust-Aware Aggregation

AF-IoTS employs trust-aware aggregation mechanisms that dynamically weight client updates based on historical consistency and anomaly indicators to alleviate the impact of malicious or unreliable participants. The trust-aware aggregation mechanism in AF-IoTS is designed to reduce the impact of poisoning and Byzantine attacks in federated learning. AF-IoTS enhances robustness by integrating trust evaluation into the aggregation process without requiring access to raw client data.

5.6 Adaptive Intrusion Detection

The AF-IoTS architecture combines federated learning with adaptive intrusion detection schemes. These schemes dynamically adjust detection sensitivity based on contextual and threat-related feedback. This feature allows a trade-off between false positives and false negatives as traffic patterns and attack intensity change. This adaptability is especially crucial in various Internet-of-Things settings, which often lack generalizable detection thresholds.

5.7 Dynamic Response Coordination

The AF-IoTS system enables collaborative, prearranged responses across three layers: IoT, edge, and orchestration. Coordination mechanisms to limit attack propagation in a large-scale distributed system are essential.

5.8 Handling Non-IID Data and System Heterogeneity

Non-IID distributions and differences in device capabilities hinder federated intrusion detection. To overcome these issues, AF-IoTS uses adaptive learning rates, aggregation weights, and

participation policies based on observed data and system properties. Heterogeneity-adaptive handling in AF-IoTS aims to improve robustness and convergence in federated learning deployments.

5.9 End-to-End Federated Security Workflow

The complete AF-IoTS process includes independent monitoring of sensor glitches, context evaluation, client selection, federated training, aggregation without trust level, intrusion detection, and coordinated response in a feedback loop.

6. Discussion, Use Cases, and Comparative Analysis

6.1 Discussion of the Proposed AF-IoTS Framework

The AF-IoTS framework addresses major limitations of existing IoT intrusion detection solutions, such as non-adaptivity, lack of Trust Awareness, and a lack of cross-layer coordination [11, 19]. Recent studies show that static security mechanisms are becoming ineffective in dynamic, large-scale IoT systems where traffic patterns, device behavior, and the nature of threats are continuously changing. The AF-IoTS incorporates adaptive control, federated learning, and trust-aware coordination within a unified architectural framework, aligning with emerging design principles for intelligent and resilient IoT security systems [12, 41].

Beyond the architectural description, three analytical observations are relevant. First, coupling the trust score with the EMA smoothing factor λ introduces a controllable trade-off between adversarial reactivity and stability under transient noise: increasing λ suppresses reaction to isolated deviations but slows the response to a persistent attacker, and this trade-off is exposed as a design knob rather than hidden inside the aggregation rule.

Second, the diversity-aware utility U_i shifts selection pressure toward under-represented data distributions, which is expected to reduce the well-documented accuracy degradation of FedAvg on non-IID splits, at the cost of admitting clients whose updates would otherwise be down-weighted; the trust term bounds this trade-off T_i^{EMA} that multiplies U_i . Third, update clipping with threshold τ upper-

bounds the influence any single client can exert per round, which is a formal robustness guarantee against unbounded poisoning even before Trust converges. These properties are architectural consequences of the formulation in Section 5.2 rather than post-hoc claims, and they define the empirical hypotheses that a later experimental study will validate.

6.2 Practical Use Cases

Note. The following use cases are presented as prospective, architecture-level analyses, not as empirical case studies. Their role is to show that the mechanisms of Section 5 are meaningful under each domain's constraints and to identify the deployment parameters that a subsequent empirical study will need to instantiate.

6.2.1 Smart Cities

Smart city infrastructures consist of a wide variety of heterogeneous IoT devices deployed in transportation, surveillance, and public services [11, 15]. As seen in earlier works [30, 31], centralized or static IDS solutions still cannot handle the scalability and latency issues in these environments. By enabling distributed, privacy-preserving intrusion detection that dynamically adapts learning and response policies to localized threats, AF-IoTS is well-suited for smart-city implementations [11, 19].

6.2.2 Smart Healthcare Systems

Modern health care systems generate sensitive medical data with high privacy and reliability requirements [35]. Federated learning is considered a prominent solution for privacy-preserving healthcare analytics in IoT applications [35] and [37]. AF-IoTS further improves this capability by introducing trust-aware mechanisms that reduce the impact of unreliable or compromised devices, which is important for safety-critical healthcare [9] and [39].

6.2.3 Industrial IoT

Security techniques that can adjust to the environment have been developed and adapted to various types of industrial environments. With its ability to support adaptive detection and coordinated response at the edge, AF-IoTS can offer a resilient

security foundation for industrial IoT deployments [29] and [33].

6.3 Comparative Analysis with Existing Approaches

Traditional intrusion detection system (IDS) solutions face scalability and single-point-of-failure issues in large-scale IoT environments [3] and [14]. Table 1 presents an architectural, qualitative comparison between AF-IoTS and two reference categories: classical centralized IDS and standard FL-based IDS. The comparison is qualitative because the compared classes represent different design philosophies rather than a single reproducible benchmark; the intent is to make the architectural differences explicit before empirical validation is performed in future work.

Despite being privacy-preserving and decentralized, existing FL-based intrusion detection system frameworks use static aggregation and lack measures against dynamic and unreliable participants [18], [19], and [38]. In contrast, the proposed architecture incorporates adaptivity, trust-

aware aggregation, and cross-layer orchestration, bringing qualitative benefits in flexibility, robustness, and Readiness for next-generation IoT and 6G scenarios [11], [12] and [40].

To further contextualize AF-IoTS against recently published work, Table 2 provides an extended architectural comparison against four representative FL security frameworks from our reference list: FLTrust [21], FreqFed [26], the precision-guided approach of

Kumar et al. [27], and FLAD [42]. For each framework, we report four design dimensions (trust-awareness, non-IID handling, adaptivity, and cross-layer integration) alongside the framework's target domain and its main reported contribution, as described in the original publication. This comparison is drawn from the papers themselves, not new experiments. It shows that while each existing framework addresses a subset of the four design dimensions, AF-IoTS is the only framework in this set that integrates all four dimensions in a single architecture.

Table 1: Qualitative architectural comparison between AF-IoTS, centralized IDS, and standard FL-based IDS.

#	Aspect	Centralized IDS	FL-based IDS	AF-IoTS (Proposed)
1	Data privacy	Requires raw data aggregation	Preserves data locality	Privacy-preserving by design
2	Scalability	Limited by central bottlenecks	Improved scalability	Designed for ultra-dense IoT
3	Adaptivity to Dynamic Conditions	Static configurations	Limited or static adaptivity	Closed-loop adaptive control
4	Handling Non-IID Data	Poor generalization	Partially addressed	Explicitly considered
5	Trust-Aware Aggregation	Not supported	Rarely supported	Integrated trust feedback
6	Resilience to Poisoning/ Byzantine Attacks	Vulnerable	Limited defenses	Trust-aware and robust
7	Cross-Layer Coordination	Absent	Typically, isolated	Cross-Layer orchestration
8	Edge Intelligence Support	Centralized processing	Partial edge usage	Native edge intelligence
9	6G Readiness	Not aligned	Limited consideration	Designed for 6G IoT
10	Architectural Integration	Monolithic	Learning-centric	Unified adaptive architecture

Table 2: Extended architectural comparison between AF-IoTS and four recent FL security frameworks along four design dimensions.

6.4 Limitations and Future Research Directions

Although AF-IoTS provides many benefits, it is at the architectural and conceptual level, and comprehensive experimental validation is future work. Future research should focus on implementing practical adaptive algorithms and developing large-scale simulations to assess performance under a variety of attack models and network conditions [38, 39]. In addition, adopting emerging technologies such as AI-native security services or digital twins is a promising way to enhance adaptive IoT security further [12] and [40].

6.5 Analytical Justification of the Trust and Clipping Mechanisms

To complement the architectural discussion above, this subsection establishes two formal properties of AF-IoTS that follow directly from the mathematical formulation of 5.2. These properties give a principled, evaluation-independent justification for the robustness claims made about the framework.

Property 1: (Bounded Client Influence via Update Clipping).

The clipping rule of 5.2.6, $\Delta_i^{clip}(t) = \Delta_i(t) \cdot \min\left(1, \frac{\tau}{\|\Delta_i(t)\|_2}\right)$, guarantees that the L2-norm of every clipped update satisfies $\|\Delta_i^{clip}(t)\|_2 \leq \tau$, irrespective of the magnitude of the raw local update $\Delta_i(t)$. Since the aggregation weight satisfies $\omega_i(t) \in [0, 1]$ by construction (it is a normalized quantity), the contribution of any individual client to the global update is bounded by $\|\omega_i(t) \cdot \Delta_i^{clip}(t)\|_2 \leq \tau$. Consequently, an adversary that submits an arbitrarily large poisoned update cannot shift the global model by more than τ per round through that single client, regardless of the attacker's strategy. This gives a per-round, worst-case robustness bound that holds before the trust mechanism has had time to converge, protecting the earliest rounds of training when trust scores are least informative.

Property 2: (Exponential Trust Decay for Persistent Adversaries).

Consider an adversarial client i that maintains a persistently high-risk score, $R_i(t) \geq R_{max}$, for every round $t \geq t_0$ (for example, by repeatedly

submitting updates that deviate significantly from the collective mean). By the instantaneous trust definition

$T_i(t) = \exp(-R_i(t))$ of 5.2.3, the client's instantaneous trust is upper-bounded by $T_{min} = \exp(-R_{max})$ for all $t \geq t_0$. Unrolling the EMA recursion

$T_i^{EMA}(t) = \lambda T_i^{EMA}(t-1) + (1-\lambda)T_i(t)$ over k successive rounds yields $T_i^{EMA}(t_0 + k) \leq \lambda \cdot T_i^{EMA}(t_0) + (1-\lambda^k) \cdot T_{min}$. Hence T_i^{EMA} converges monotonically toward T_{min} at a geometric rate governed by λ ; for the default choice $\lambda = 0.8$ recommended in 5.2.8, roughly 89% of the decay is achieved within ten rounds (since $1 - 0.8^{10} \approx 0.893$). This bounds the number of rounds during which a persistent adversary can retain meaningful aggregation weight and quantifies the trade-off between adversarial reactivity and stability against transient noises exposed through the smoothing factor λ .

Taken together, Property 1 provides an immediate, non-asymptotic robustness guarantee at every round. In contrast, Property 2 provides an asymptotic guarantee against persistent adversaries. The two mechanisms are complementary: clipping bounds the damage before Trust converges, and the EMA trust decay progressively excludes persistent adversaries from the aggregation. These analytical properties are architectural consequences of the formulation and do not depend on any specific dataset or attacker distribution, which makes them a suitable evaluation-independent basis for the design contribution presented in this paper.

7. Conclusion

This paper has presented AF-IoTS, an architectural framework for privacy-preserving and adaptive intrusion detection in large-scale and heterogeneous IoT environments. The framework couples federated learning with four coordinated mechanisms: a trust-aware coordination mechanism, a diversity-aware client-selection utility, a trust-weighted robust aggregation rule with update clipping, and a closed-loop adaptive controller, all integrated within a single layered architecture that departs from conventional centralized or static security solutions. Through this

integration, AF-IoTS provides a systematic design that preserves privacy while remaining scalable and resilient under dynamic threats and unreliable participants.

The discussions and prospective use-case analyses conducted in this work indicate that AF-IoTS applies to IoT domains such as smart cities, healthcare systems, and industrial IoT, which impose combined privacy, reliability, and low-latency requirements. Unlike current FL-based IDS methods, AF-IoTS explicitly addresses heterogeneity, non-IID data distribution, and poisoning resistance at the architectural level. This effort deliberately focuses on architectural design rather than a specific algorithmic implementation, providing a coherent and logically consistent foundation for trustable and adaptive IoT security.

Building upon this foundation, future research will focus on engineering concrete adaptive algorithms, conducting large-scale experiments under precise attacker models, and extending the framework toward emerging IoT use cases enabled by 6G. This paper's contribution is architectural and analytical rather than empirical. The study introduces an integrated four-layer framework that unifies federated learning with a trust model, a diversity-aware selection utility, robust aggregation, and a closed-loop adaptive controller.

The paper further provides explicit mathematical formulations for all components, together with the end-to-end algorithm and its computational cost. The paper also presents a qualitative architectural comparison against centralized and standard FL-based IDS categories, complemented by a prospective analysis across three IoT deployment domains. The next stage of this line of work involves empirical validation on benchmark IoT security datasets, quantitative comparisons under specified attacker models, and statistical dispersion analysis (including mean, standard deviation, and confidence intervals across independent runs).

Acknowledgment

The author expresses gratitude to the Department of Forensic Evidence, Faculty of Medical Sciences,

Jabir Ibn Hayyan University for Medical and Pharmaceutical Sciences, Najaf, Iraq, for institutional support for this project. The author expresses gratitude to the unknown reviewers of BAJEST for their careful reading and constructive comments, which significantly improved the clarity and precision of the manuscript.

Conflict of Interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

References

- [1] Hassija, V., Chamola, V., Saxena, V., Jain, D., Goyal, P., & Sikdar, B. (2019). A survey on IoT security: Application areas, security threats, and solution architectures. *IEEE Access*, 7, 82721-82743.
- [2] Khanam, S., Ahmedy, I. B., Idris, M. Y. I., Jaward, M. H., & Sabri, A. Q. B. M. (2020). A survey of security challenges, attack taxonomy and advanced countermeasures in the Internet of Things. *IEEE Access*, 8, 219709-219743.
- [3] Ferrag, M. A., Maglaras, L., Moschogiannis, S., & Janicke, H. (2020). Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications*, 50, 102419.
- [4] Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: Techniques, datasets and challenges: cybersecurity, 2(1), 20.
- [5] Lim, W. Y. B., Luong, N. C., Hoang, D. T., Jiao, Y., Liang, Y.-C., Yang, Q., Niyato, D., & Miao, C. (2020). Federated learning in mobile edge networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(3), 2031-2063.
- [6] Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., & Gao, Y. (2021). A survey on federated learning. *Knowledge-Based Systems*, 216, 106775.
- [7] Khan, L. U., Saad, W., Han, Z., Hossain, E., & Hong, C. S. (2021). Federated learning for Internet of Things: Recent advances, taxonomy, and open challenges. *IEEE Communications Surveys & Tutorials*, 23(3), 1759-1799.
- [8] AbdulRahman, S., Tout, H., Ould-Slimane, H., Mourad, A., Talhi, C., & Guizani, M. (2021).

- A survey on federated learning: The journey from centralized to distributed on-site learning and beyond. *IEEE Internet of Things Journal*, 8(7), 5476-5497.
- [9] Mothukuri, V., Parizi, R. M., Pouriyeh, S., Huang, Y., Dehghantanha, A., & Srivastava, G. (2021). A survey on security and privacy of federated learning. *Future Generation Computer Systems*, 115, 619-640.
- [10] Li, Q., Diao, Y., Chen, Q., & He, B. (2022). Federated learning on non-IID data silos: An experimental study. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)* (pp. 965-978).
- [11] De Alwis, C., Kalla, A., Pham, Q.-V., Kumar, P., Dev, K., Hwang, W.-J., & Liyanage, M. (2021). Survey on 6G frontiers: Trends, applications, requirements, technologies and future research. *IEEE Open Journal of the Communications Society*, 2, 836-886.
- [12] Jiang, W., Han, B., Habibi, M. A., & Schotten, H. D. (2021). The road towards 6G: A comprehensive survey. *IEEE Open Journal of the Communications Society*, 2, 334-366.
- [13] Sultana, N., Chilamkurti, N., Peng, W., & Alhadad, R. (2019). Survey on SDN-based network intrusion detection system using machine learning approaches. *Peer-to-Peer Networking and Applications*, 12(2), 493-501.
- [14] Aldweesh, A., Derhab, A., & Emam, A. Z. (2020). Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues: knowledge-Based Systems, 189, 105124.
- [15] Aloqaily, M., Otoum, S., Al Ridhawi, I., & Jararweh, Y. (2019). An intrusion detection system for connected vehicles in smart cities. *Ad Hoc Networks*, 90, 101842.
- [16] Hasan, M., Islam, M. M., Zarif, M. I. I., & Hashem, M. M. A. (2019). Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches—*Internet of Things*, 7, 100059.
- [17] Imteaj, A., Thakker, U., Wang, S., Li, J., & Amini, M. H. (2022). A survey on federated learning for resource-constrained IoT devices. *IEEE Internet of Things Journal*, 9(1), 1-24.
- [18] Mothukuri, V., Khare, P., Parizi, R. M., Pouriyeh, S., Dehghantanha, A., & Srivastava, G. (2022). Federated-learning-based anomaly detection for IoT security attacks. *IEEE Internet of Things Journal*, 9(4), 2545-2554.
- [19] Popoola, S. I., Ande, R., Adebisi, B., Gui, G., Hammoudeh, M., & Jogunola, O. (2021). Federated deep learning for collaborative intrusion detection in heterogeneous networks. In *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)* (pp. 1-6).
- [20] Fang, M., Cao, X., Jia, J., & Gong, N. Z. (2020). Local model poisoning attacks to Byzantine-robust federated learning, in *Proceedings of the 29th USENIX Security Symposium* (pp. 1605-1622).
- [21] Cao, X., Fang, M., Liu, J., & Gong, N. Z. (2021). FLTrust: Byzantine-robust federated learning via trust bootstrapping. *Network and Distributed System Security Symposium (NDSS)*.
- [22] Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3), 50-60.
- [23] Guo, F., Yu, F. R., Zhang, H., Li, X., Ji, H., & Leung, V. C. M. (2021). Enabling massive IoT toward 6G: A comprehensive survey. *IEEE Internet of Things Journal*, 8(15), 11891-11915.
- [24] Cui, Y., Cao, K., Cao, G., Qiu, M., & Wei, T. (2022). Client scheduling and resource management for efficient training in heterogeneous IoT-edge federated learning. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 41(8), 2407-2420.
- [25] Ye, M., Fang, X., Du, B., Yuen, P. C., & Tao, D. (2023). Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Computing Surveys*, 56(3), 79.
- [26] Fereidooni, H., Pegoraro, A., Rieger, P., Dmitrienko, A., & Sadeghi, A.-R. (2024). FreqFed: A frequency analysis-based approach for mitigating poisoning attacks in federated learning. *Network and Distributed System Security Symposium (NDSS)*.
- [27] Kumar, K. N., Mohan, C. K., & Machiry, A. (2024). Precision-guided approach to mitigate data poisoning attacks in federated learning. In *Proceedings of the 14th ACM Conference on Data and Application Security and Privacy (CODASPY)*.
- [28] Zhou, Z., Chen, X., Li, E., Zeng, L., Luo, K., & Zhang, J. (2019). Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proceedings of the IEEE*, 107(8), 1738-1762.
- [29] Khan, W. Z., Ahmed, E., Hakak, S., Yaqoob, I., & Ahmed, A. (2019). Edge computing: A

- survey. *Future Generation Computer Systems*, 97, 219-235.
- [30] Ren, J., Zhang, D., He, S., Zhang, Y., & Li, T. (2019). A survey on end-edge-cloud orchestrated network computing paradigms: Transparent computing, mobile edge computing, fog computing, and cloudlet. *ACM Computing Surveys*, 52(6), 1-36.
- [31] Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., Kong, J., & Jue, J. P. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. *Journal of Systems Architecture*, 98, 289-330.
- [32] Deng, S., Zhao, H., Fang, W., Yin, J., Dustdar, S., & Zomaya, A. Y. (2020). Edge intelligence: The confluence of edge computing and artificial intelligence. *IEEE Internet of Things Journal*, 7(8), 7457-7469.
- [33] Gartner. (2022). Market guide for security orchestration, automation and response solutions. Gartner Research.
- [34] Liu, Y., Dong, Y., Wang, H., Jiang, H., & Xu, Q. (2022). Distributed fog computing and federated-learning-enabled secure aggregation for IoT devices. *IEEE Internet of Things Journal*, 9(21), 21025-21037.
- [35] Xu, J., Glicksberg, B. S., Su, C., Walker, P., Bian, J., & Wang, F. (2021). Federated learning for healthcare informatics. *Journal of Healthcare Informatics Research*, 5(1), 1-19.
- [36] Kairouz, P., et al. (2021). Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1-2), 1-210.
- [37] Nguyen, D. C., Ding, M., Pathirana, P. N., Seneviratne, A., Li, J., & Poor, H. V. (2021). Federated learning for Internet of Things: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 23(3), 1622-1658.
- [38] Ferrag, M. A., Friha, O., Maglaras, L., Janicke, H., & Shu, L. (2021). Federated deep learning for cyber security in the Internet of Things: Concepts, applications, and experimental analysis. *IEEE Access*, 9, 138509-138542.
- [39] Zhang, Z., Cao, X., Jia, J., & Gong, N. Z. (2022). FLDetector: Defending federated learning against model poisoning attacks via detecting malicious clients in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 2545-2555).
- [40] Campos, E. M., Saura, P. F., González-Vidal, A., Hernández-Ramos, J. L., Bernabé, J. B., Baldini, G., & Skarmeta, A. (2022). Evaluating federated learning for intrusion detection in Internet of Things: Review and challenges. *Computer Networks*, 203, 108661.
- [41] Xu, D., Li, T., Li, Y., Su, X., Tarkoma, S., Jiang, T., Crowcroft, J., & Hui, P. (2021). Edge intelligence: Empowering intelligence to the edge of the network. *Proceedings of the IEEE*, 109(11), 1778-1837.
- [42] Doriguzzi-Corin, R., & Siracusa, D. (2024). FLAD: Adaptive federated learning for DDoS attack detection. *Computers & Security*, 137, 103597