

Article

Robust Trust-Aware Federated Learning for Privacy-Preserving Distributed Intelligence in Resource-Constrained IoT Systems: A Blockchain-Assisted Architecture

Manuel J. C. S. Reis ^{1,*} , Carlos Serôdio ²  and Frederico Branco ³ 

¹ Engineering Department and IEETA, University of Trás-os-Montes e Alto Douro, Quinta de Prados, 5000-801 Vila Real, Portugal

² Engineering Department and Center ALGORITMI, University of Trás-os-Montes e Alto Douro, Quinta de Prados, 5000-801 Vila Real, Portugal; cserodio@utad.pt

³ Engineering Department and INESC-TEC, University of Trás-os-Montes e Alto Douro, Quinta de Prados, 5000-801 Vila Real, Portugal; fbranco@utad.pt

* Correspondence: mcabral@utad.pt

Abstract

The rapid expansion of Internet of Things (IoT) ecosystems has intensified the need for distributed intelligence mechanisms that reduce direct raw-data exposure while remaining resilient to adversarial manipulation. Federated learning (FL) addresses part of this challenge by enabling collaborative model training without centralizing raw data, but it remains vulnerable to malicious client behavior, particularly model poisoning attacks that can substantially degrade global model quality. This paper investigates a blockchain-assisted, trust-aware FL framework for privacy-aware distributed intelligence in resource-constrained IoT systems, where the blockchain layer is used to support trust coordination, auditability, traceability, and tamper-resistant metadata recording rather than to directly improve predictive performance. The empirical study compares six aggregation strategies: FedAvg, coordinate-wise Median, Trimmed Mean, FLTrust, trust-aware weighted aggregation, and a hybrid trust-trimmed mean method. The primary evaluation is conducted on the UCI Human Activity Recognition (UCI HAR) dataset under Dirichlet-based non-IID client partitioning $\alpha = 0.1$ and $\alpha = 1.0$, partial client participation, and sign-flip model poisoning. Each configuration is evaluated over five independent runs. Under the severe 40% malicious-client stress test, FLTrust achieves the strongest mean robustness among the evaluated methods, reaching 0.4195 ± 0.1295 accuracy and 0.3108 ± 0.1294 macro-F1 for $\alpha = 0.1$, and 0.6472 ± 0.0725 accuracy and 0.6014 ± 0.0949 macro-F1 for $\alpha = 1.0$. In lower-intensity attack controls with 10% and 20% malicious clients, the trust-aware and hybrid trust-trimmed strategies are the most competitive, achieving the highest or near-highest mean performance without requiring a clean server-side reference set. Mechanism-level analysis shows that FLTrust is particularly effective in the severe setting because its reference-based scoring assigns near-zero weights to malicious clients, whereas trust-aware and hybrid methods rely on relative update consistency and are more affected by the interaction between poisoning and statistical heterogeneity. Globally, the results indicate that no single aggregation rule dominates all adversarial regimes. Instead, reference-based trust provides strong protection under severe poisoning, while trust-aware and hybrid aggregation offer competitive root-free alternatives under lower attack intensities. The findings also clarify that the privacy-preserving scope of the framework derives from FL-based data locality and does not constitute a formal cryptographic privacy guarantee.



Academic Editors: Zijian Bao, Yunhao Ling and Zhuangzhuang Zhang

Received: 3 June 2026

Revised: 8 July 2026

Accepted: 9 July 2026

Published: 13 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: federated learning; internet of things; trust-aware aggregation; robust aggregation; model poisoning attacks; statistical heterogeneity; blockchain-assisted architecture; privacy-aware distributed intelligence

1. Introduction

The rapid expansion of Internet of Things (IoT) ecosystems has led to an unprecedented increase in the volume of distributed data generated by sensors, wearable devices, industrial controllers, and smart infrastructure. These environments increasingly rely on machine learning models to support monitoring, classification, anomaly detection, and automated decision-making. However, conventional centralized learning approaches often require collecting raw data from edge devices into a central server, which introduces important privacy, security, bandwidth, and governance concerns. In many IoT settings, the transmission and central storage of raw data may be undesirable or even infeasible due to confidentiality requirements, regulatory constraints, or resource limitations [1,2].

Federated Learning (FL) has emerged as a promising paradigm to address these issues by allowing multiple devices or clients to collaboratively train a global model while keeping raw data local. Instead of sharing raw observations, each participant performs local training and transmits only model updates or gradients to a coordinating server. This approach can reduce direct data exposure and aligns well with the distributed nature of IoT deployments. The foundational formulation of FL through iterative model averaging was introduced by McMahan et al., whose FedAvg algorithm established the canonical server-coordinated FL setting and highlighted its communication advantages under decentralized data [3]. Subsequent surveys and monographs have emphasized that FL is particularly attractive in edge and IoT environments because it supports data minimization and distributed intelligence while partially mitigating the privacy risks associated with centralized collection [4,5].

Despite these advantages, FL does not eliminate privacy and security risks. Although raw data remain decentralized, model updates can still reveal sensitive information under inference-based attacks, and the collaborative training process itself can be manipulated by malicious or compromised participants. This distinction is important: FL should be understood as a privacy-preserving training paradigm in the sense of data locality, rather than as a complete privacy guarantee. Prior work has shown that privacy leakage, poisoning, backdoor insertion, and adversarial manipulation remain significant concerns in practical FL systems, particularly when large numbers of heterogeneous clients participate under weak trust assumptions [4,6,7].

Among these threats, model poisoning is one of the most critical for adversarial FL deployments. In poisoning attacks, malicious clients intentionally manipulate local model updates in order to degrade the global model, bias predictions, or destabilize convergence. This issue is especially relevant in IoT ecosystems, where devices may be heterogeneous, intermittently available, physically exposed, and more susceptible to compromise than tightly controlled cloud infrastructure. The literature on FL security consistently identifies poisoning and Byzantine behavior as central challenges for real-world adoption, particularly in edge and IoT contexts where device trust cannot be assumed a priori [1,7].

A further complication arises from the inherently non-IID nature of IoT data. In realistic deployments, different devices often observe distinct local distributions due to user behavior, sensor placement, hardware variation, mobility, or environmental context. This statistical heterogeneity already makes federated optimization more difficult even in benign

settings. Under adversarial conditions, the combination of non-IID data and poisoned updates can significantly amplify instability and reduce the effectiveness of standard aggregation methods such as FedAvg. This challenge has been repeatedly emphasized in the FL literature as one of the core reasons why robust and adaptive aggregation strategies are needed beyond simple weighted averaging [4,5].

To mitigate these risks, the robust distributed learning literature has proposed aggregation rules designed to tolerate Byzantine or adversarial clients. Seminal approaches include Krum, which selects gradients based on pairwise consistency under Byzantine assumptions, and coordinate-wise robust statistics such as median and trimmed mean, which explicitly suppress extreme updates [8,9]. These methods provide an important conceptual basis for robust FL, although their practical performance in non-IID IoT scenarios can vary substantially depending on participation rate, attack strength, and client distribution shift. In parallel, trust-aware defenses and similarity-based filtering methods such as FoolsGold have shown that leveraging client behavior or update similarity can improve resilience against coordinated poisoning and sybil-like attacks, especially when adversarial participants attempt to dominate the global update [10].

At the same time, the increasing interest in blockchain-assisted federated learning reflects a broader effort to reduce single points of failure, improve auditability, and support decentralized trust management in collaborative edge intelligence systems. Rather than replacing the statistical robustness problem, blockchain is typically positioned as an architectural support layer that can enhance coordination, provenance tracking, immutability of participation records, and accountability of update-related metadata. Recent surveys indicate that the combination of blockchain and FL is particularly relevant for IoT because both paradigms address complementary concerns: FL reduces raw-data centralization, while blockchain can strengthen trust, traceability, and decentralized governance. However, these benefits come with practical overhead and do not by themselves solve poisoning robustness or non-IID instability [11–13].

Motivated by these challenges, this paper investigates a blockchain-assisted, trust-aware federated learning framework for privacy-aware distributed intelligence in resource-constrained IoT environments. The blockchain component is not treated as the direct source of predictive-performance improvements. Instead, it is positioned as an architectural support layer for recording participation metadata, trust-score summaries, model-version identifiers, and audit information in a tamper-resistant manner. The empirical contribution of the paper is centered on the robustness of the aggregation layer under adversarial model poisoning and statistically heterogeneous client partitions.

The study is designed to provide a broader and more critical assessment than a comparison against FedAvg alone. It compares conventional averaging, classical robust statistical aggregation, reference-based trust aggregation, and root-free trust-aware aggregation under the same federated evaluation protocol. This design is important because different robust aggregation families rely on different assumptions. For example, coordinate-wise robust statistics aim to suppress extreme update components, FLTrust uses a trusted server-side reference signal, and root-free trust-aware strategies infer client reliability from the submitted updates themselves. Comparing these alternatives under common conditions allows their advantages and limitations to be interpreted more clearly.

The main contributions of this paper are summarized as follows:

1. We design and evaluate a blockchain-assisted FL framework for resource-constrained IoT-oriented classification, where blockchain is used as a trust, traceability, and audit-support layer rather than as the direct source of predictive-performance gains.
2. We provide a comparative robustness study of six aggregation strategies: FedAvg, Median, Trimmed Mean, FLTrust, trust-aware weighting, and hybrid trust-trimmed mean.

3. We examine the behavior of these aggregation strategies under adversarial model poisoning, partial client participation, and statistically heterogeneous client partitions, with the aim of clarifying how robustness depends on both attack intensity and non-IID severity.
4. We distinguish reference-based trust from root-free trust-aware aggregation by explicitly considering the role of a clean server-side reference set in FLTrust and by comparing it with strategies that do not assume such a root dataset.
5. We provide a mechanism-level analysis of trust scores, client aggregation weights, and trimming behavior, allowing the observed performance differences to be interpreted in terms of how each strategy modulates the influence of benign and malicious clients.
6. We explicitly delimit the scope of the framework by clarifying that privacy preservation derives from FL-based data locality, that formal cryptographic privacy mechanisms are not implemented, and that blockchain overhead in terms of storage, communication, latency, and energy consumption is not independently benchmarked in this study.

Importantly, the blockchain component is presented as an architectural trust and audit support layer and is not independently benchmarked as the direct source of predictive gains. The empirical comparison is intentionally centered on aggregation robustness under adversarial conditions.

The remainder of this paper is organized as follows. Section 2 reviews related work on federated learning for IoT, privacy-preserving distributed learning, robust aggregation under poisoning attacks, trust-aware FL, and blockchain-assisted FL architectures. Section 3 presents the proposed system model and aggregation strategies. Section 4 describes the experimental setup, datasets, attack model, evaluation protocol, and compared baselines. Section 5 reports the experimental results and mechanism-level analysis. Section 6 discusses the findings, limitations, and threats to validity. Finally, Section 7 concludes the paper and outlines directions for future work.

2. Related Work

2.1. Federated Learning for IoT and Resource-Constrained Edge Environments

Federated Learning (FL) has become an important paradigm for Internet of Things (IoT) and edge intelligence because it enables collaborative model training without requiring centralized access to raw device data. This property is particularly relevant in IoT scenarios, where devices often operate under privacy, bandwidth, storage, energy, and ownership constraints. Recent surveys identify FL as a natural fit for sensing, human activity recognition, industrial monitoring, healthcare telemetry, and cyber-physical systems, where data are generated close to the edge and may be impractical or undesirable to centralize [1,14,15].

However, FL deployment in resource-constrained IoT systems is considerably more challenging than in homogeneous data-center or mobile-device settings. IoT clients may differ in computational capacity, energy budget, connectivity, local sample size, and data distribution. These factors can produce intermittent participation, communication bottlenecks, unstable convergence, and uneven contribution quality [15,16]. Consequently, FL for IoT should be treated as a system-level co-design problem involving privacy-aware data locality, optimization stability, robustness, and resource-awareness. This motivates the use of domain-relevant sensor benchmarks and controlled non-IID partitioning when evaluating FL methods for IoT-oriented applications [14,17].

2.2. Statistical Heterogeneity in Federated Learning

A central challenge in FL is statistical heterogeneity, commonly referred to as non-IID data, where each client observes a local distribution that differs from the global population. This issue is especially pronounced in IoT environments because user behavior, sensor placement, hardware variation, mobility, and environmental context naturally induce client-specific distributions. Prior work has shown that non-IID data can degrade the stability and scalability of standard FL optimization pipelines [4,5,18].

FedAvg remains the canonical FL baseline due to its simplicity and communication efficiency, but its performance can deteriorate when client updates drift in different directions under heterogeneous local data. Methods such as FedProx and SCAFFOLD were proposed to mitigate client drift and improve convergence under statistical or system heterogeneity [5,19]. More broadly, survey work distinguishes several forms of heterogeneity, including label skew, feature skew, quantity skew, concept shift, and temporal drift [18]. These distinctions are important in adversarial settings because robust aggregation rules that perform well when benign updates are tightly clustered may become less effective when benign updates are already naturally dispersed.

2.3. Privacy, Security, and Poisoning Threats

Although FL is often described as privacy-preserving, this should be interpreted carefully. Keeping raw data local reduces direct raw-data exposure, but model updates may still leak information under inference attacks. Stronger confidentiality guarantees generally require additional mechanisms such as secure aggregation, differential privacy, secure multiparty computation, or trusted execution support [6,7,20]. Therefore, FL-based privacy in this work is understood primarily as data locality rather than as a formal cryptographic privacy guarantee.

In addition to privacy leakage, FL remains vulnerable to poisoning and Byzantine attacks. Malicious clients can manipulate local data, model updates, or training objectives to degrade global performance, bias predictions, or insert backdoors. Model poisoning is particularly relevant because the server often has limited visibility into local training behavior, especially when privacy-enhancing mechanisms reduce access to individual client updates [7,21]. Seminal attacks such as model-replacement backdoors and small Byzantine perturbations have shown that malicious updates do not need to be obviously extreme to be effective, which makes simple anomaly detection insufficient in adversarial FL scenarios [22,23].

2.4. Robust Aggregation and Trust-Aware Defenses

To reduce the influence of malicious clients, robust aggregation rules have been proposed as alternatives to conventional averaging. Krum selects updates based on pairwise consistency assumptions, while coordinate-wise Median and Trimmed Mean suppress extreme update components without explicitly modeling attacker behavior [8,9]. These methods provide important baselines for Byzantine-robust FL, but their effectiveness can vary under non-IID data because benign updates may no longer concentrate around a single stable central tendency.

Trust-aware defenses address this limitation by weighting client updates according to estimated reliability. FoolsGold, for example, down-weights clients with suspiciously similar update patterns, targeting sybil-like or colluding adversaries [10]. FLTrust introduces a different form of trust estimation by using a small trusted server-side root dataset to define a reference update direction; client updates are then weighted according to directional alignment and magnitude normalization [24]. This makes FLTrust a strong reference-based

baseline, but it also assumes the availability of clean server-side data, which may not always be realistic in privacy-sensitive IoT deployments.

More recent credibility-based methods explicitly consider the difficulty of distinguishing malicious updates from heterogeneous benign ones under non-IID data [25]. Overall, this literature suggests that robust statistics and trust estimation are complementary: robust aggregation can suppress extreme values, while trust mechanisms can modulate client influence based on behavioral consistency or reference alignment. This motivates comparative evaluation of purely statistical, reference-based trust, and root-free trust-aware aggregation strategies under the same adversarial and non-IID conditions.

2.5. Blockchain-Assisted Federated Learning

In parallel with robust FL research, blockchain-assisted FL has attracted attention as a way to improve decentralization, auditability, coordination, and trust management in collaborative learning systems. This integration is particularly relevant in IoT and Industrial IoT settings, where participants may belong to different stakeholders and where provenance, accountability, and tamper-resistant recording of participation events can be operationally valuable [11–13,26].

The role of blockchain in FL should nevertheless be stated conservatively. Blockchain does not inherently improve model accuracy, convergence, or poisoning robustness. Instead, it can support immutable logging of participation metadata, model-version identifiers, trust-score summaries, incentive records, access-control events, and audit trails [11,13]. In adversarial FL, such records may improve transparency and forensic traceability, but the statistical effect of poisoned updates still depends on the aggregation rule. Accordingly, recent work increasingly emphasizes that blockchain-assisted FL should be evaluated both as a system-trust architecture and as a learning pipeline, without assuming that architectural trust automatically implies predictive robustness [12,26].

The present study follows this conservative interpretation. Blockchain is positioned as an architectural layer for trust coordination, traceability, and auditability, while the empirical focus is placed on aggregation robustness under adversarial poisoning and non-IID client partitions. This positioning separates three concepts that are sometimes conflated: FL provides data locality, robust aggregation provides statistical resilience, and blockchain provides tamper-resistant coordination and audit support.

3. Proposed Framework and Methodology

3.1. System Overview

This work proposes a blockchain-assisted robust trust-aware federated learning framework for privacy-preserving distributed intelligence in resource-constrained IoT systems. The framework is designed for scenarios in which multiple IoT clients collaboratively train a shared model without transferring raw local data to a central repository. This follows the standard federated learning principle introduced by FedAvg, where clients train locally and the server aggregates local model updates into a global model [3].

Figure 1 presents the overall architecture of the proposed framework, highlighting the interaction between local client training, federated coordination, robust aggregation, adversarial behavior, and the blockchain-assisted trust and audit layer. The framework separates three roles that are often conflated in blockchain-assisted FL research. First, FL provides data locality by keeping raw client data within each device or domain. Second, the aggregation layer provides statistical robustness against unreliable or malicious model updates. Third, the blockchain-assisted layer provides tamper-resistant support for participation tracking, trust metadata recording, model-version traceability, and auditability.

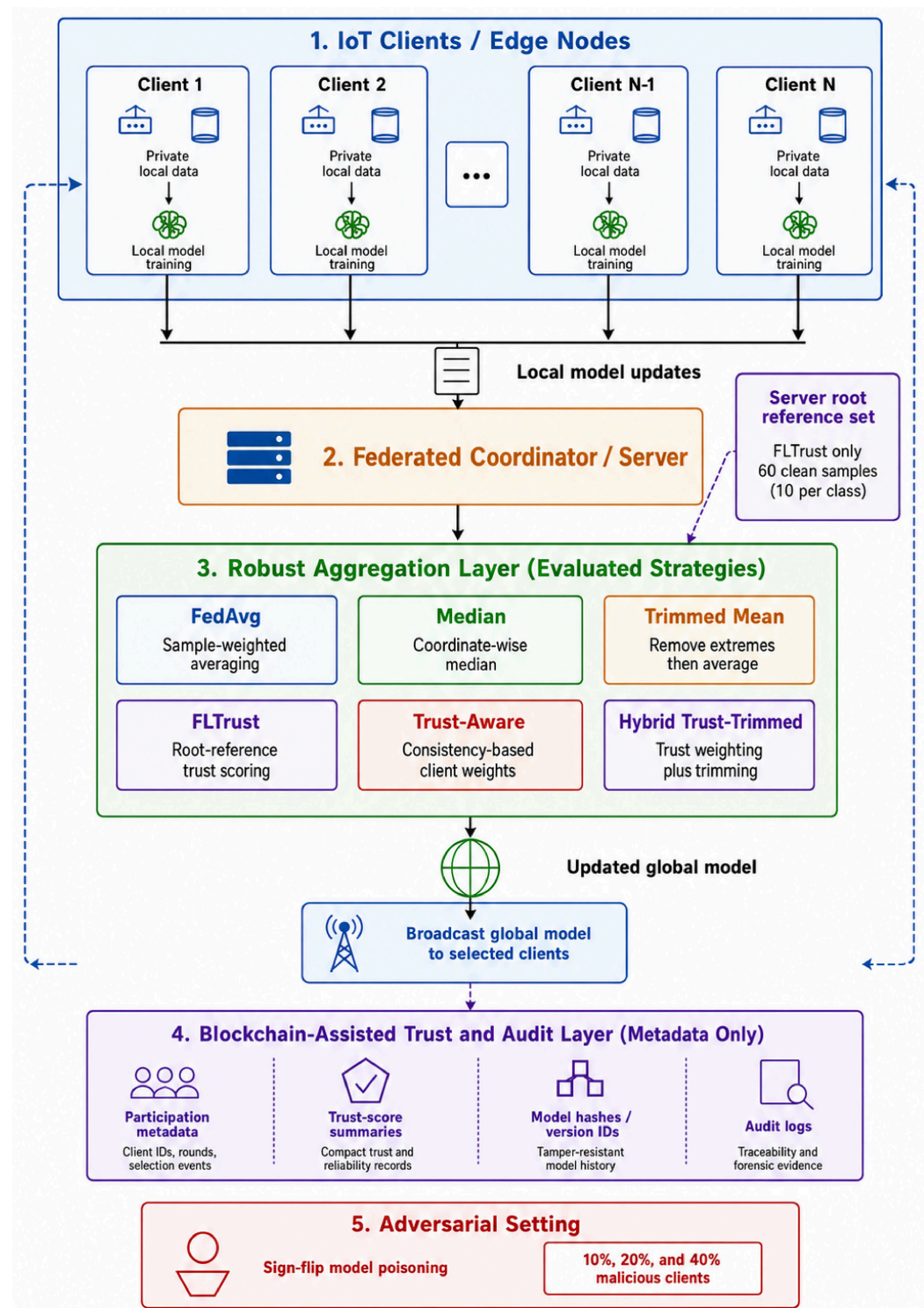


Figure 1. Blockchain-assisted robust federated learning (FL) architecture for IoT. IoT clients perform local training using private data and transmit only model updates to the federated server. The server applies one of the evaluated aggregation strategies (FedAvg, Median, Trimmed Mean, FLTrust, Trust-Aware, or Hybrid Trust-Trimmed) before broadcasting the updated global model to the next communication round. The blockchain-assisted layer records only trust- and audit-related metadata rather than raw data or model parameters. Solid black arrows indicate the primary data/model-update flow, dashed blue arrows denote the iterative FL communication cycle, and the dashed purple arrow indicates the server-side reference path used exclusively by FLTrust. Colored boxes identify the main functional layers of the proposed architecture and distinguish the evaluated aggregation strategies.

The proposed framework is therefore blockchain-assisted at the architectural level, privacy-aware through FL-based data locality, and robust through the evaluated aggregation mechanisms. The blockchain layer is not treated as the direct source of predictive-performance gains. Instead, the empirical evaluation focuses on whether different aggregation strategies can reduce the influence of poisoned client updates under adversarial and non-IID conditions.

In this study, six aggregation strategies are considered: standard FedAvg, coordinate-wise Median, Trimmed Mean, FLTrust, trust-aware weighted aggregation, and a hybrid trust-trimmed mean method. FedAvg is used as the conventional baseline. Median and Trimmed Mean represent classical robust statistical baselines. FLTrust is included as a reference-based trust baseline that relies on a small trusted server-side root dataset. The trust-aware and hybrid trust-trimmed methods represent root-free trust-based alternatives that estimate client reliability from update consistency rather than from a clean server reference set.

3.2. Federated Learning Setting and Threat Model

We consider a federated learning system composed of one coordinating server and a set of distributed IoT clients. Each client holds a private local dataset and trains a local model using its own data. Raw data are never transmitted to the server. At each communication round, a subset of clients is selected according to a partial participation policy. Selected clients receive the current global model, perform local training for a fixed number of local epochs, and return their model updates or updated model parameters to the server.

The server then aggregates the received client contributions to update the global model. This process is repeated for a fixed number of communication rounds. In the experimental implementation used in this study, partial participation is configured through the client fraction and minimum number of selected clients. The evaluation uses 30 federated clients, with 9 clients selected per communication round.

The general training loop follows four steps:

1. The server initializes the global model.
2. A subset of eligible clients is selected for the current round.
3. Each selected client performs local training using its private data.
4. The server aggregates the submitted updates using one of the evaluated aggregation rules.

This design reflects the standard FL setting while allowing the aggregation rule to be replaced by robust or trust-aware alternatives. The specific number of clients, participation rate, number of rounds, local training configuration, and dataset partitioning strategy used in the experiments are reported in Section 4.

The threat model assumes the presence of malicious clients participating in the federated training process. These clients are allowed to submit manipulated model updates with the objective of degrading the global model. The server does not directly observe local raw data and does not assume that all participating clients are trustworthy. This is consistent with adversarial FL scenarios where compromised or dishonest clients may behave in a Byzantine manner.

The poisoning attack considered in the main experiments is a sign-flip model poisoning attack. Under this attack, malicious clients invert the direction of their local update before submission, thereby attempting to push the global model away from the direction supported by benign clients. The evaluation includes a severe 40% malicious-client stress-test scenario and additional lower-intensity controls with 10% and 20% malicious clients. The 40% setting is therefore interpreted as an adversarial robustness stress test rather than as a typical IoT deployment assumption.

The goal of the defense is not to identify every malicious client perfectly, but to reduce the influence of harmful updates on the global aggregation step. This is important because prior Byzantine-robust FL work has shown that malicious clients can corrupt global model training even when the number of attackers is bounded [24].

3.3. Description of the Aggregation Strategies

3.3.1. FedAvg

FedAvg is used as the conventional baseline aggregation method. In FedAvg, the server computes a weighted average of the client model updates, typically proportional to the number of local training samples contributed by each client. This method is simple, communication-efficient, and widely used as a canonical FL baseline [3].

However, FedAvg assumes that client updates are broadly reliable. Under model poisoning, this assumption is violated. If malicious clients submit adversarial updates, their contributions are averaged together with benign updates, which can degrade the global model. This vulnerability motivates the use of robust aggregation strategies.

3.3.2. Coordinate-Wise Median

The coordinate-wise Median is included as a classical robust aggregation baseline. Instead of averaging each parameter coordinate across client updates, the server computes the median value for each coordinate. This reduces the influence of extreme values and can provide resilience against Byzantine or outlying updates when a sufficient fraction of clients is benign [9].

The advantage of Median is that it does not require explicit trust scores, attacker labels, or a trusted reference dataset. However, its performance may degrade under strong non-IID conditions because benign client updates can already be dispersed. In that case, coordinate-wise median aggregation may suppress useful heterogeneous contributions or fail to distinguish malicious updates from legitimate distribution-shifted ones.

3.3.3. Trimmed Mean

Trimmed Mean is also included as a robust statistical baseline. For each model parameter coordinate, the server removes a fixed fraction of the largest and smallest submitted values and averages the remaining values. This approach is designed to reduce the impact of extreme malicious updates while preserving more information than the coordinate-wise Median [9].

Like Median, Trimmed Mean does not require a trusted root dataset or explicit client identities. Its robustness depends on the trimming fraction, the number of selected clients, and the separation between benign and malicious update distributions. Under strong heterogeneity, aggressive trimming may remove useful benign updates, whereas insufficient trimming may leave poisoned contributions in the aggregation step.

3.3.4. FLTrust

FLTrust is included as a reference-based trust aggregation baseline [24]. Unlike purely statistical robust aggregation methods, FLTrust assumes that the server has access to a small trusted root dataset. The server uses this clean reference set to compute a trusted reference update direction. Client updates are then scored according to their directional alignment with the server reference update and normalized to reduce the influence of excessively large update magnitudes.

In this study, FLTrust is treated as a strong reference-based comparator rather than as the proposed method. Its inclusion is important because it provides a direct benchmark for assessing how much robustness can be gained when a clean server-side reference signal is available. At the same time, this assumption represents a methodological advantage and a

deployment limitation: in privacy-sensitive IoT scenarios, the availability of a representative clean root dataset at the server may not always be realistic. For this reason, the trust-aware and hybrid methods are also evaluated as root-free alternatives.

3.3.5. Trust-Aware Aggregation

The trust-aware aggregation method assigns each client a reliability score that modulates the influence of its submitted update. Instead of treating all selected clients equally, the server estimates a trust value based on the consistency or plausibility of the submitted update. Clients with lower trust receive lower aggregation weight, while clients with higher trust contribute more strongly to the global model.

In contrast to FLTrust, the trust-aware method considered here does not rely on a trusted server-side root dataset. Trust is updated dynamically across rounds using consistency information derived from the population of submitted updates. In the implementation used in this study, submitted updates are compared against a robust reference derived from the received client updates, and each selected client receives a consistency score. Client trust then evolves gradually across communication rounds according to bounded update rules.

The trust-aware strategy is intended to improve robustness against malicious clients while preserving the contribution of benign clients. This is particularly important in IoT environments, where client reliability may vary over time due to compromise, noise, resource constraints, or unstable operating conditions. However, because trust is inferred from relative update consistency, this strategy can be affected by severe statistical heterogeneity and high malicious-client ratios, where benign and malicious updates may become more difficult to separate.

3.3.6. Hybrid Trust-Trimmed Mean Aggregation

The hybrid trust-trimmed mean method combines two complementary robustness mechanisms: trust-aware weighting and trimmed aggregation. The motivation is that trust scores provide a client-level reliability signal, while trimming reduces the effect of extreme or suspicious update components. Robust aggregation based on trimmed mean has been shown to provide resilience against Byzantine failures in distributed learning by suppressing extreme coordinate-wise contributions [9].

In the proposed hybrid method, client updates are first evaluated through the trust-aware mechanism. Trust scores are then incorporated into the aggregation process, while trimming is used to reduce the influence of outlying update values. This makes the method more conservative than pure trust-aware weighting and more adaptive than a purely statistical trimmed mean.

The intended advantage of the hybrid design is that it addresses two different failure modes. If malicious clients submit extreme sign-flipped updates, trimming can reduce their coordinate-wise impact. If malicious clients submit less extreme but persistently inconsistent updates, trust weighting can reduce their influence over time. This layered defense is especially relevant in non-IID IoT settings, where benign updates may already be heterogeneous and simple outlier rejection may be insufficient.

Unlike FLTrust, the hybrid method does not assume a clean server-side root dataset. Its robustness therefore depends on the quality of the trust signal estimated from client-update consistency and on the ability of trimming to suppress harmful coordinate-wise extremes. This distinction is central to the experimental comparison, because it separates reference-based trust, which uses an external clean signal, from root-free strategies whose robustness must be inferred only from the submitted client updates.

3.4. Blockchain-Assisted Trust and Audit Layer

The blockchain-assisted component is included as an architectural support layer for decentralized trust management, auditability, and integrity of trust-related metadata. In the proposed framework, the blockchain layer can be used to record selected events such as client participation, aggregation metadata, trust-score evolution, model-version identifiers, and audit trails. This helps reduce dependence on a fully opaque centralized coordinator and supports traceability of the federated learning process.

The envisioned blockchain layer is lightweight and metadata-oriented. It is not intended to store raw client data, full model updates, or complete model parameters. Instead, a permissioned or consortium blockchain can be used to record compact information such as round identifiers, client participation events, hashes of submitted updates or aggregated models, trust-score summaries, model-version hashes, and audit logs. This design is more compatible with resource-constrained IoT settings because it avoids placing high-dimensional model parameters or sensitive local data directly on-chain.

Importantly, the blockchain layer is not claimed to directly improve predictive accuracy or poisoning robustness. Its role is complementary. The statistical robustness of the global model is provided by the aggregation mechanism, while the blockchain-assisted layer supports transparency, accountability, and tamper-resistant recording of trust-related information. This distinction is necessary because secure or immutable recording of updates does not, by itself, prevent poisoned updates from damaging the global model.

In the present study, the blockchain layer is specified at the architectural level and is not independently benchmarked in terms of storage overhead, communication overhead, latency, energy consumption, or consensus cost. These aspects are therefore treated as deployment considerations and are explicitly discussed as limitations. The empirical evaluation focuses on aggregation robustness under adversarial conditions.

3.5. Privacy-Aware Design Considerations

The framework is privacy-preserving in the federated learning sense: raw client data remain local and are not centralized. This reduces direct exposure of sensitive IoT data and is consistent with the privacy motivation of FL. However, the framework does not assume that FL alone provides complete privacy protection. Model updates can still leak information under certain inference attacks, and stronger privacy guarantees may require additional mechanisms such as secure aggregation or differential privacy. Secure aggregation protocols, for example, are designed to allow a server to aggregate high-dimensional client updates without observing individual contributions [6].

Accordingly, the term privacy-aware or privacy-preserving in this paper refers to data locality and reduced raw-data centralization, not to a formal cryptographic privacy guarantee. The current framework does not implement differential privacy, secure aggregation, secure multiparty computation, homomorphic encryption, or trusted execution environments. Therefore, it does not claim strict protection against strong privacy attacks such as model inversion, gradient leakage, or membership inference.

This scope is intentional. The primary focus of this study is adversarial robustness against malicious model updates. Privacy is addressed at the architectural level through local data retention, while robustness is evaluated empirically through poisoning experiments. Future work may integrate formal privacy-enhancing mechanisms with robust aggregation, although such integration may introduce additional trade-offs between confidentiality, observability, detection capability, and model accuracy.

3.6. Methodological Positioning

The methodology intentionally separates three concepts that are often conflated in blockchain-assisted FL research:

1. FL provides data locality by avoiding centralized raw-data collection.
2. Robust aggregation provides resilience against adversarial or unreliable client updates.
3. Blockchain assistance provides auditability, traceability, and trust-recording support.

This separation is important for scientific clarity. The experiments in this paper evaluate how different aggregation strategies behave under sign-flip poisoning, partial participation, and non-IID client partitions. The blockchain layer supports the broader trustworthy IoT architecture but is not treated as the direct causal source of the observed accuracy or macro-F1 differences.

This positioning makes the claims of the paper conservative and testable: the proposed framework is blockchain-assisted at the architectural level, privacy-aware through federated data locality, and robust only to the extent supported by the evaluated aggregation strategies under the tested adversarial conditions.

4. Experimental Setup

4.1. Datasets

The primary experimental evaluation is conducted on the UCI Human Activity Recognition (UCI HAR) dataset, which is used as the main benchmark for the federated learning experiments. UCI HAR is a widely adopted human-centric sensing dataset collected from smartphone inertial sensors and has been extensively used in activity recognition research. It contains tri-axial accelerometer and gyroscope measurements acquired from 30 subjects performing six daily activities, namely walking, walking upstairs, walking downstairs, sitting, standing, and lying. The dataset is particularly suitable for the present study because it reflects a realistic IoT-oriented classification setting based on distributed sensor observations and provides a stable and reproducible benchmark for evaluating privacy-preserving edge intelligence [27]. The use of this benchmark is also consistent with mobile and edge FL settings characterized by heterogeneous devices, decentralized sensing data, and communication constraints [28]. Such distributed learning environments must additionally account for Byzantine participants capable of submitting malicious or carefully crafted model updates [29].

The experimental campaign focuses on UCI HAR as the main evidence base because it provides a reproducible sensor-classification benchmark aligned with the target IoT application domain. This choice also allows controlled non-IID partitioning, repeated runs across random seeds, and direct comparison of multiple robust aggregation strategies under identical adversarial conditions. Auxiliary experiments with synthetic data and preliminary TON_IoT integration, if retained in the manuscript, are treated only as secondary checks or implementation evidence and are not used as the main basis for the comparative conclusions.

4.2. Data Partitioning and Non-IID Client Distribution

To emulate realistic federated learning conditions, the local data distribution across clients is intentionally made non-independent and non-identically distributed (non-IID). This is essential because, in practical IoT deployments, different devices often observe distinct local distributions due to user behavior, environmental variation, sensor placement, and context-specific data generation. The impact of non-IID data on federated optimization is well established and is one of the major factors affecting both convergence and robustness [18].

In the main experiments, client data are partitioned using a Dirichlet-based allocation strategy, which is a standard and widely used approach for simulating heterogeneous

client distributions in federated learning studies. Specifically, two heterogeneity regimes are considered:

- $\alpha = 0.1$, representing a strongly heterogeneous and highly skewed non-IID setting;
- $\alpha = 1.0$, representing a moderately heterogeneous but still non-IID setting.

Lower Dirichlet concentration parameters produce more extreme client-level class imbalance, thereby increasing statistical heterogeneity. This allows the experiments to explicitly examine how the aggregation methods behave under both severe and moderate distribution shift across clients.

The federated system uses 30 clients in total. At each communication round, only a subset of clients participates, reflecting realistic partial-participation conditions. This combination of non-IID partitioning and partial participation creates a more challenging evaluation environment than fully IID or full-participation settings and is particularly relevant for resource-constrained IoT deployments.

4.3. Federated Training Configuration

All evaluated methods share the same core federated training configuration to ensure a fair comparison. In each communication round, a subset of clients is selected to perform local training and return updated model parameters to the server for aggregation. The main UCI HAR campaign used the following configuration:

- Number of clients: 30;
- Client fraction: 0.3;
- Minimum client floor: 5;
- Clients selected per round: 9;
- Number of communication rounds: 20;
- Local epochs per selected client: 1;
- Batch size: 32;
- Learning rate: 0.001.

With 30 total clients and a client fraction of 0.3, the client selection rule yields 9 participating clients per round. The minimum client floor was set to 5, but it was not binding in the reported experiments because the fraction-based selection already exceeded this threshold.

The model architecture, optimizer, number of rounds, local training configuration, client participation rule, and attack configuration were kept consistent across all aggregation methods within each experimental condition. This ensures that observed performance differences can be attributed primarily to the aggregation rule rather than to changes in local optimization settings.

The number of communication rounds was intentionally limited to maintain a practical runtime budget while still allowing the aggregation strategies to exhibit meaningful convergence behavior under adversarial and non-IID conditions. Likewise, the local training depth was kept intentionally shallow (one local epoch per selected client) to better reflect resource-constrained edge learning scenarios, in which excessive local computation may be undesirable and where robustness must be assessed under realistic computational budgets.

4.4. Adversarial Attack Configuration

To evaluate robustness under malicious participation, the experiments introduce a sign-flip model poisoning attack. In this threat model, malicious clients invert the direction of their local model updates before sending them to the server. This causes the adversarial contribution to move against the dominant optimization direction of benign clients, thereby degrading global convergence and final predictive performance.

The main adversarial configuration uses a malicious client ratio of 40%. This setting is deliberately severe and is interpreted as a robustness stress test rather than as a typical real-world deployment assumption. To address this point explicitly, the experimental campaign also includes lower-intensity control scenarios with 10% and 20% malicious clients. These controls help distinguish behavior under moderate attack pressure from behavior under the severe stress-test condition.

The 40% malicious-client scenario is evaluated under both Dirichlet regimes, $\alpha = 0.1$ and $\alpha = 1.0$. The lower-intensity 10% and 20% attack controls are evaluated under the moderately heterogeneous $\alpha = 1.0$ setting. This design prioritizes the most difficult heterogeneity comparison for the severe stress test while using the lower-ratio controls to assess whether the same aggregation trends persist when adversarial pressure is reduced.

Sign-flip poisoning is a well-established and interpretable adversarial benchmark in federated learning because it directly tests whether the aggregation rule can resist strongly misaligned updates. It is particularly suitable for comparative evaluation because it creates a clear distinction between benign and malicious update directions without requiring complex attacker-side optimization.

4.5. Compared Aggregation Strategies

The experimental campaign compares the six aggregation strategies defined in Section 3.3 under identical data partitions, client participation rules, attack settings, training hyperparameters, and random seeds. Therefore, this section specifies their role in the experimental protocol rather than repeating their methodological definitions. FedAvg is used as the conventional non-robust baseline; coordinate-wise Median and Trimmed Mean are used as robust statistical baselines; FLTrust is used as a reference-based trust baseline; and trust-aware aggregation and hybrid trust-trimmed mean aggregation are used as root-free trust-based alternatives.

For FLTrust, the server root set contains 10 clean samples per class, resulting in 60 reference samples in total for the six-class UCI HAR task. To preserve comparability in the data available to the federated clients, these samples are removed from the client partitions for all compared methods. However, the resulting server-side reference update is used only by FLTrust. FedAvg, Median, Trimmed Mean, trust-aware aggregation, and the hybrid trust-trimmed mean method do not access this clean reference set. This design allows the experiments to distinguish the effect of a clean server-side reference signal from root-free robustness mechanisms.

All methods are evaluated using the same model architecture, optimizer, number of communication rounds, local training configuration, client selection policy, malicious-client ratio, attack type, evaluation metrics, and independent random seeds. Consequently, the comparisons reported in Section 5 are intended to isolate the influence of the aggregation rule rather than differences in training configuration.

4.6. Trust Configuration and Mechanism Logging

For the trust-aware and hybrid trust-trimmed mean aggregation strategies, client trust values are initialized and updated dynamically during training. In the reported experiments, the trust mechanism is governed by the following configuration:

- Initial trust value (τ_0): 0.8;
- Minimum trust bound ($\tau_{\min}$): 0.05;
- Maximum trust bound ($\tau_{\max}$): 1.0;
- Trust update coefficient (β): 0.35;
- Consistency sensitivity parameter (κ): 4.0.

At each communication round, the updates submitted by the selected clients are compared against a robust reference update derived from the coordinate-wise median of the received updates. A consistency score in the interval $[0, 1]$ is then assigned to each selected client based on its normalized deviation from this reference, with the sensitivity controlled by κ . Trust is subsequently updated using an exponential moving average (EMA) that combines the previous trust value with the current consistency score, after which the resulting value is clipped to the interval $[\tau_{\min}, \tau_{\max}]$.

In addition to final classification performance, the experiments log mechanism-level indicators to support interpretation of the aggregation behavior. These include average trust values, consistency scores, client aggregation weights, low-weight client fractions, malicious-versus-benign average weights when labels are available for analysis, FLTrust score summaries, and the fraction of update components removed by trimming. These indicators are used in Section 5 to explain why different aggregation strategies behave differently under severe and lower-intensity attack regimes.

4.7. Evaluation Metrics

Performance is evaluated using accuracy and macro-F1 score. Let C denote the number of classes, TP_c , FP_c , and FN_c denote the true positives, false positives, and false negatives for class c , and N denote the total number of test samples. Accuracy is defined as Equation (1):

$$\text{Accuracy} = \frac{\sum_{c=1}^C TP_c}{N}. \quad (1)$$

For each class c , precision, recall, and F1-score are defined as Equations (2)–(4), respectively:

$$\text{Precision}_c = \frac{TP_c}{TP_c + FP_c}, \quad (2)$$

$$\text{Recall}_c = \frac{TP_c}{TP_c + FN_c}, \quad (3)$$

$$F1_c = \frac{2 \cdot \text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}. \quad (4)$$

Macro-F1 is then computed as the unweighted mean of the class-wise F1-scores, as shown in Equation (5):

$$\text{Macro-F1} = \frac{1}{C} \sum_{c=1}^C F1_c. \quad (5)$$

Accuracy provides a global measure of correct predictions across all classes. However, because federated non-IID partitioning can induce class imbalance across clients and because adversarial disruption may affect classes unevenly, macro-F1 is particularly important. Macro-F1 gives equal weight to each class and therefore provides a more balanced assessment of performance under heterogeneous and potentially imbalanced conditions.

The primary reported results focus on final test performance after 20 communication rounds. Per-round behavior and mechanism-level aggregation statistics are used where relevant to support interpretation, but the main comparisons are based on final accuracy and macro-F1 over five independent runs.

4.8. Statistical Analysis

To improve the reliability of the comparative analysis, each experimental configuration is repeated over five independent random seeds. Results are reported as mean $\pm$ standard deviation. Where appropriate, pairwise comparisons are conducted using Welch's two-sample t -test, which does not assume equal variances between methods. The tests are used

as supporting evidence rather than as definitive proof of superiority, given the relatively small number of independent runs.

This statistical treatment directly addresses the variability expected in adversarial FL under non-IID partitioning, where random client partitions, malicious-client selection, and partial participation can substantially affect final performance. Accordingly, the results are interpreted cautiously, emphasizing both mean performance and variability across seeds.

4.9. Experimental Protocol and Scope

The experimental protocol is designed to isolate the effect of the aggregation rule under controlled adversarial and heterogeneous conditions. For each dataset and heterogeneity regime, all compared methods are evaluated under the same:

- Client partitioning scheme;
- Number of clients;
- Participation rate;
- Number of communication rounds;
- Local training configuration;
- Malicious client ratio;
- Attack type;
- Evaluation metrics;
- Random seeds.

This controlled setup ensures that the comparison remains focused on the aggregation strategy itself.

A practical consideration in adversarial federated learning is that performance can be sensitive to the specific random partitioning of data across clients, especially under strong non-IID conditions. This is particularly relevant for the $\alpha = 0.1$ regime, where client-level class skew is more severe and benign updates may naturally diverge more strongly. For this reason, the results should be interpreted not as universal absolute performance guarantees, but as comparative evidence under a reproducible and intentionally challenging evaluation protocol.

The experimental design is intentionally comparative rather than exhaustive. The goal is not to benchmark every possible adversarial defense or every IoT dataset, but to assess how conventional averaging, robust statistical aggregation, reference-based trust, and root-free trust-aware aggregation behave under controlled but practically meaningful IoT-oriented federated conditions. This scope allows the study to address the reviewers' request for stronger baselines and mechanism-level explanation while keeping the empirical claims conservative and reproducible.

5. Results and Mechanism Analysis

5.1. Main UCI HAR Results Under Severe Sign-Flip Poisoning

Table 1 summarizes the main comparative results obtained on the UCI HAR benchmark under the severe sign-flip poisoning stress test with 40% malicious clients and partial participation. Results are reported as mean $\pm$ standard deviation over five independent runs for the two Dirichlet non-IID regimes considered in this study, $\alpha = 0.1$ and $\alpha = 1.0$. The comparison includes six aggregation strategies: FedAvg, Median, Trimmed Mean, FLTrust, trust-aware aggregation, and hybrid trust-trimmed mean.

The results substantially refine the interpretation of the original study. Under the severe 40% malicious-client stress test, FLTrust achieves the highest mean accuracy and macro-F1 in both heterogeneity regimes. For $\alpha = 0.1$, FLTrust reaches 0.4195 ± 0.1295 accuracy and 0.3108 ± 0.1294 macro-F1, compared with 0.1695 ± 0.0072 and 0.0483 ± 0.0018 , respectively, for FedAvg. This corresponds to gains of approximately +0.2500 accuracy and

+0.2625 macro-F1 over FedAvg. Welch's t -test indicates that these gains are statistically significant at the 5% level for both accuracy ($p = 0.0124$) and macro-F1 ($p = 0.0105$).

Table 1. UCI HAR performance under 40% sign-flip poisoning. Best results are shown in bold.

Partition	Method	Accuracy	Macro-F1
$\alpha = 0.1$	FedAvg	0.1695 ± 0.0072	0.0483 ± 0.0018
$\alpha = 0.1$	Median	0.2812 ± 0.0600	0.1667 ± 0.0711
$\alpha = 0.1$	Trimmed Mean	0.1859 ± 0.0382	0.0649 ± 0.0358
$\alpha = 0.1$	FLTrust	0.4195 ± 0.1295	0.3108 ± 0.1294
$\alpha = 0.1$	Trust-aware	0.1695 ± 0.0072	0.0483 ± 0.0018
$\alpha = 0.1$	Hybrid trust-trimmed mean	0.2414 ± 0.0754	0.1339 ± 0.0850
$\alpha = 1.0$	FedAvg	0.4566 ± 0.2690	0.3521 ± 0.2995
$\alpha = 1.0$	Median	0.5263 ± 0.2582	0.4233 ± 0.3026
$\alpha = 1.0$	Trimmed Mean	0.4422 ± 0.2471	0.3368 ± 0.2703
$\alpha = 1.0$	FLTrust	0.6472 ± 0.0725	0.6014 ± 0.0949
$\alpha = 1.0$	Trust-aware	0.5271 ± 0.2712	0.4342 ± 0.3163
$\alpha = 1.0$	Hybrid trust-trimmed mean	0.5565 ± 0.2818	0.4635 ± 0.3291

In the same $\alpha = 0.1$ regime, Median also improves over FedAvg, reaching 0.2812 ± 0.0600 accuracy and 0.1667 ± 0.0711 macro-F1. These gains are statistically significant relative to FedAvg for both accuracy ($p = 0.0136$) and macro-F1 ($p = 0.0203$). Trimmed Mean provides only limited improvement, while the trust-aware strategy collapses to the same final mean performance as FedAvg in this severe regime. The hybrid trust-trimmed method improves over FedAvg, but the difference remains only marginal with five runs ($p = 0.1000$ for accuracy and $p = 0.0873$ for macro-F1).

For $\alpha = 1.0$, the same general ranking is observed, with FLTrust again achieving the highest mean performance. However, variability across runs is much larger for FedAvg, Median, trust-aware aggregation, and the hybrid method. Consequently, although FLTrust improves over FedAvg by approximately +0.1905 accuracy and +0.2493 macro-F1, the pairwise differences are not statistically significant with five independent runs ($p = 0.1921$ for accuracy and $p = 0.1387$ for macro-F1). This high variability confirms that adversarial FL performance under partial participation and non-IID data can be strongly affected by random partitioning, client selection, and malicious-client sampling.

These results indicate that the severe 40% attack regime is particularly favorable to reference-based trust. FLTrust benefits from a clean server-side reference update, which provides an external anchor for scoring client updates. In contrast, the root-free trust-aware and hybrid methods infer reliability from the submitted client updates themselves. This makes them less dependent on clean server-side data, but also more vulnerable when the malicious-client fraction is high and benign updates are already dispersed by non-IID partitioning.

5.2. Lower-Intensity Attack Controls

To clarify whether the 40% malicious-client setting should be interpreted as a severe stress test rather than as a typical operating condition, additional control experiments were conducted with 10% and 20% malicious clients under the moderately heterogeneous $\alpha = 1.0$ partition. Table 2 summarizes these results.

The lower-intensity controls reveal a different pattern from the severe 40% stress test. With 10% malicious clients, the trust-aware and hybrid methods achieve the highest mean performance, with trust-aware aggregation reaching 0.8809 ± 0.0317 accuracy and 0.8759 ± 0.0370 macro-F1. Trimmed Mean also performs strongly, whereas FLTrust is substantially lower in this setting.

Table 2. UCI HAR performance under 10% and 20% sign-flip poisoning with $\alpha = 1.0$. Best results are shown in bold.

Malicious Ratio	Method	Accuracy	Macro-F1
0.1	FedAvg	0.8645 ± 0.0502	0.8567 ± 0.0599
0.1	Median	0.8292 ± 0.0769	0.8159 ± 0.0942
0.1	Trimmed Mean	0.8725 ± 0.0387	0.8688 ± 0.0434
0.1	FLTrust	0.5765 ± 0.0162	0.5115 ± 0.0198
0.1	Trust-aware	0.8809 ± 0.0317	0.8759 ± 0.0370
0.1	Hybrid trust-trimmed mean	0.8775 ± 0.0328	0.8720 ± 0.0387
0.2	FedAvg	0.8210 ± 0.0494	0.7968 ± 0.0768
0.2	Median	0.8287 ± 0.0808	0.8148 ± 0.0987
0.2	Trimmed Mean	0.7920 ± 0.0368	0.7627 ± 0.0585
0.2	FLTrust	0.5920 ± 0.0366	0.5271 ± 0.0390
0.2	Trust-aware	0.8597 ± 0.0249	0.8544 ± 0.0282
0.2	Hybrid trust-trimmed mean	0.8598 ± 0.0517	0.8489 ± 0.0679

A similar pattern appears with 20% malicious clients. Trust-aware aggregation and the hybrid trust-trimmed method again achieve the strongest mean results, with nearly identical accuracy values. The hybrid method obtains the highest mean accuracy, 0.8598 ± 0.0517 , while trust-aware aggregation obtains the highest mean macro-F1, 0.8544 ± 0.0282 . These results show that root-free trust-aware mechanisms can be highly competitive when the attack intensity is low to moderate.

The contrast between Tables 1 and 2 is central to the interpretation of the study. FLTrust is the strongest method under the severe 40% malicious-client stress test, where a clean server-side reference update provides a decisive robustness advantage. However, when the malicious-client ratio is reduced to 10% or 20%, trust-aware and hybrid aggregation become the best or near-best methods without relying on a clean server root dataset. Therefore, the results do not support a universal dominance claim for any single aggregation rule. Instead, they show that the most suitable defense depends on attack intensity, statistical heterogeneity, and whether a trusted reference dataset is available.

5.3. Mechanism-Level Analysis

To better understand why the aggregation strategies behave differently across attack regimes, mechanism-level indicators were logged during training. Table 3 summarizes the most relevant indicators under the severe 40% malicious-client stress test. These indicators include the mean aggregation weight assigned to malicious and benign clients, the fraction of low-weight clients, the zero-score fraction in FLTrust, and the trimming fraction in the hybrid method.

Table 3. Mechanism-level aggregation indicators under the severe 40% malicious-client stress test. Values are averaged over five independent runs.

Partition	Method	Malicious Weight	Benign Weight	Main Mechanism Indicator
$\alpha = 0.1$	FLTrust	0.0043	0.1937	Zero-score fraction: 0.4100
$\alpha = 0.1$	Trust-aware	0.1338	0.0926	Trust gap: 0.1138
$\alpha = 0.1$	Hybrid trust-trimmed mean	0.1006	0.1102	Trimmed fraction: 0.2222
$\alpha = 1.0$	FLTrust	0.0000	0.1986	Zero-score fraction: 0.4067
$\alpha = 1.0$	Trust-aware	0.0977	0.1174	Trust gap: 0.0532
$\alpha = 1.0$	Hybrid trust-trimmed mean	0.0804	0.1245	Trimmed fraction: 0.2222

The mechanism indicators help explain the strong performance of FLTrust in the severe stress test. For $\alpha = 0.1$, FLTrust assigns an average weight of only 0.0043 to malicious clients, compared with 0.1937 for benign clients. For $\alpha = 1.0$, the average malicious-client weight is effectively zero, while benign clients receive an average weight of 0.1986. In

both regimes, approximately 40% of selected updates receive zero FLTrust score, indicating that the server reference update sharply separates many malicious contributions from benign ones.

The root-free trust-aware and hybrid methods show a more nuanced behavior. Under $\alpha = 0.1$, trust-aware aggregation assigns a higher average aggregation weight to malicious clients than to benign clients, which explains its collapse to FedAvg-level final performance in this severe regime. This occurs because severe non-IID dispersion makes relative consistency-based trust estimation difficult when the malicious-client fraction is high. The hybrid method partially mitigates this effect: its malicious and benign average weights are closer, and coordinate-wise trimming removes approximately 22.2% of extreme update components. Nevertheless, this is insufficient to match the reference-based separation achieved by FLTrust.

Under $\alpha = 1.0$, the root-free methods behave more favorably. Trust-aware aggregation assigns a lower average weight to malicious clients than to benign clients, and the hybrid method strengthens this separation further, with malicious and benign average weights of 0.0804 and 0.1245, respectively. This helps explain why the hybrid method becomes more competitive in the moderately heterogeneous setting, although the variability across seeds remains high.

Globally, the mechanism-level results support the main empirical interpretation: reference-based trust is especially effective under severe poisoning because it provides an external clean anchor for update scoring, while root-free trust-aware and hybrid aggregation depend more strongly on the separability of benign and malicious updates within the submitted client population.

Figure 2 visualizes the aggregation-weight separation between benign and malicious clients for the main trust-based methods under the $\alpha = 1.0$, 40% malicious-client stress-test setting.

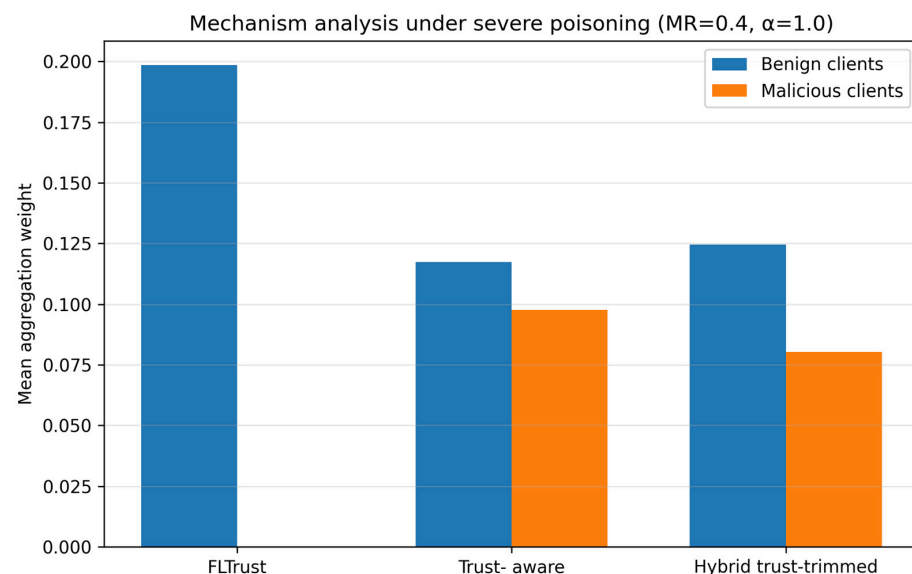


Figure 2. Mean aggregation weights assigned to benign and malicious clients under severe sign-flip poisoning with $\alpha = 1.0$.

5.4. Summary of Empirical Findings

Taken together, the experiments support four main findings. First, FedAvg is highly vulnerable under severe sign-flip poisoning, particularly when statistical heterogeneity is strong. This confirms the need for robust aggregation in adversarial FL deployments.

Second, classical robust statistical methods provide partial protection, but their behavior is not uniformly strong. Median improves substantially over FedAvg in the severe

$\alpha = 0.1$ regime, whereas Trimmed Mean is less effective under the tested configurations. This suggests that coordinate-wise robust statistics can help, but their effectiveness depends on the interaction between trimming or selection behavior, partial participation, and non-IID client dispersion.

Third, FLTrust provides the strongest mean robustness under the severe 40% malicious-client stress test. The mechanism analysis shows that this advantage is associated with its ability to assign near-zero weights to malicious clients using a clean server-side reference update. However, this advantage comes with an important assumption: FLTrust requires a trusted root dataset at the server.

Fourth, trust-aware and hybrid trust-trimmed aggregation remain important root-free alternatives. They do not dominate FLTrust under severe poisoning, but they achieve the best or near-best results under lower-intensity 10% and 20% attack controls without requiring clean server-side reference data. This makes them relevant in privacy-sensitive or decentralized IoT scenarios where maintaining a representative trusted root dataset at the server may not be feasible.

Globally, the results support a more nuanced robustness claim than the original manuscript. Robust trust-aware aggregation can improve FL resilience in adversarial IoT settings, but no single aggregation rule dominates all regimes. The most suitable defense depends on malicious-client ratio, statistical heterogeneity, partial participation, and the availability of clean reference data. The blockchain-assisted layer remains an architectural support mechanism for coordination, traceability, and auditability, while the observed predictive-performance differences are driven by the aggregation strategies themselves.

6. Discussion, Limitations, and Threats to Validity

The results provide a more nuanced interpretation than the original experimental campaign. Under the evaluated UCI HAR sign-flip poisoning scenarios, the main conclusion is not that a single aggregation strategy is universally superior, but rather that different robustness mechanisms are advantageous under different adversarial regimes. In particular, the severe 40% malicious-client setting favors reference-based trust, whereas the lower-intensity 10% and 20% malicious-client controls favor the root-free trust-aware and hybrid trust-trimmed strategies. This regime-dependent behavior is important because it avoids overstating the contribution of the proposed hybrid method and instead positions the study as a comparative analysis of robust aggregation under adversarial and non-IID IoT-oriented FL conditions.

A first key observation is that FLTrust achieves the strongest mean robustness under the severe 40% malicious-client stress test. This is especially clear in the strongly heterogeneous $\alpha = 0.1$ regime, where FLTrust substantially outperforms FedAvg and the improvement is statistically significant for both accuracy and macro-F1. The mechanism-level analysis explains this behavior: FLTrust uses a clean server-side reference update and assigns near-zero weights to malicious clients. This external reference signal gives FLTrust a strong advantage when the malicious-client fraction is high and when benign client updates are dispersed by non-IID partitioning. In practical terms, the result shows that reference-based trust can be highly effective under severe poisoning, provided that a representative and clean server root dataset is available.

However, this advantage should be interpreted together with the assumptions of FLTrust. Unlike Median, Trimmed Mean, trust-aware aggregation, and the hybrid trust-trimmed mean, FLTrust requires a trusted server-side reference set. In the present implementation, this reference set contains 10 clean samples per class, resulting in 60 reference samples for the six-class UCI HAR task. Although these samples were removed from the federated client partitions for all methods to preserve data-partition fairness, the refer-

ence update itself was used only by FLTrust. This gives FLTrust an important robustness advantage, but it also represents a deployment limitation. In privacy-sensitive or highly decentralized IoT settings, maintaining a clean and representative server-side root dataset may not always be feasible.

A second key observation is that the root-free trust-aware and hybrid trust-trimmed strategies are most competitive under lower-intensity poisoning. In the 10% and 20% malicious-client controls with $\alpha = 1.0$, trust-aware aggregation and the hybrid method achieve the highest or near-highest mean performance without requiring clean server-side reference data. This suggests that, when the adversarial fraction is moderate and benign client updates retain sufficient structural alignment, consistency-based trust estimation and trimmed aggregation can provide effective robustness while preserving the root-free character of the FL setting. This is relevant for IoT deployments where server-side reference data may be unavailable, undesirable, or difficult to keep representative over time.

The comparison between severe and lower-intensity attack regimes highlights an important practical trade-off. FLTrust is more effective under severe poisoning because it relies on an external clean anchor for update scoring. In contrast, the trust-aware and hybrid methods infer reliability from the submitted client updates themselves. This makes them more deployable in settings without trusted root data, but also more vulnerable when malicious clients represent a large fraction of the selected participants. Under severe poisoning and strong non-IID fragmentation, benign and malicious updates may become difficult to separate using relative consistency alone. This explains why the trust-aware method collapses to FedAvg-level performance under $\alpha = 0.1$ and 40% malicious clients, while the hybrid method provides only partial improvement.

The role of statistical heterogeneity is therefore central. Under $\alpha = 0.1$, client distributions are highly skewed, and benign updates may naturally diverge even in the absence of attacks. This weakens the assumptions behind both coordinate-wise robust statistics and root-free trust estimation, because an update that appears atypical may be either malicious or simply generated from a legitimate but highly skewed local distribution. Under $\alpha = 1.0$, benign updates are less fragmented, allowing trust-aware and hybrid mechanisms to separate harmful and useful contributions more effectively. This confirms that adversarial robustness in FL cannot be evaluated independently of non-IID severity, partial participation, and client-selection randomness.

Classical robust aggregation methods show mixed behavior in the campaign. Median provides meaningful protection under the severe $\alpha = 0.1$ stress test and significantly improves over FedAvg, confirming the usefulness of coordinate-wise robust statistics against extreme poisoned updates. Trimmed Mean, however, is less effective under the tested settings. This may be due to the interaction between the trimming ratio, the number of selected clients per round, and the dispersion of benign updates under non-IID data. These findings reinforce the broader point that robust statistics are valuable baselines, but their effectiveness is not automatic under heterogeneous IoT FL conditions.

The blockchain layer should also be interpreted carefully. The empirical gains reported in the results arise from the aggregation strategies, not from blockchain itself. In the proposed architecture, blockchain is used as a supporting trust and auditability layer. It can provide tamper-resistant logging of participation metadata, trust-score summaries, model-version identifiers, and audit trails. This can strengthen transparency, traceability, and accountability in multi-stakeholder IoT FL deployments. However, immutable recording of metadata does not by itself prevent poisoned updates from damaging the global model. The statistical defense against poisoning remains the responsibility of the aggregation layer.

This distinction is particularly important in resource-constrained IoT environments. IoT devices are often bandwidth-limited, energy-sensitive, intermittently connected, and

heterogeneous in computation and storage capability. For this reason, the blockchain layer proposed in this work is metadata-oriented: it is intended to store compact records such as hashes, round identifiers, participation events, trust summaries, and audit information, rather than raw data, full model updates, or complete model parameters. This design is more realistic for constrained deployments, but the present study does not independently benchmark blockchain storage overhead, communication overhead, latency, energy consumption, consensus cost, or scalability. These aspects remain important deployment considerations for future work.

The privacy scope of the framework is also intentionally limited. The term privacy-preserving is used in the federated learning sense: raw client data remain local and are not centralized. This reduces direct raw-data exposure, but it does not provide a formal cryptographic privacy guarantee. The current implementation does not include differential privacy, secure aggregation, secure multiparty computation, homomorphic encryption, or trusted execution environments. Consequently, the framework should not be interpreted as providing strict protection against model inversion, gradient leakage, membership inference, or other strong privacy attacks. Future work should investigate how formal privacy mechanisms interact with trust-aware and robust aggregation, since additional privacy protection may reduce update observability and complicate malicious-client detection.

Several limitations and threats to validity should be acknowledged. First, although the campaign increases the number of independent runs from three to five, this remains a modest sample size for adversarial FL experiments. The use of mean $\pm$ standard deviation and Welch's *t*-tests improves transparency, but statistical conclusions should still be interpreted cautiously. Second, the lower-intensity 10% and 20% malicious-client controls were conducted under the $\alpha = 1.0$ partition regime. They clarify behavior under moderate heterogeneity but do not fully characterize low-ratio attacks under extreme $\alpha = 0.1$ heterogeneity. Third, the primary evidence is based on UCI HAR. This dataset is appropriate for human-centric IoT sensing, but the results should not be assumed to generalize automatically to all IoT domains, such as industrial telemetry, smart-grid monitoring, or cybersecurity traffic.

Additional limitations relate to the attack model and compared defenses. The main adversarial setting uses sign-flip model poisoning, which is interpretable and useful for stress-testing aggregation rules, but it does not cover all realistic attack strategies. Backdoor attacks, adaptive poisoning, colluding sybil clients, stealthy model replacement, and attacks aware of the defense mechanism may lead to different behavior. Similarly, the comparison includes important baselines—FedAvg, Median, Trimmed Mean, FLTrust, trust-aware aggregation, and hybrid trust-trimmed mean—but it does not exhaust the full space of robust FL defenses. Future studies should evaluate the framework against a broader range of adaptive attacks, datasets, participation regimes, and defense mechanisms.

Despite these limitations, the results provide useful practical guidance. When a clean and representative server-side reference set is available, FLTrust can offer strong protection under severe poisoning. When such a root dataset is unavailable or undesirable, trust-aware and hybrid trust-trimmed aggregation provide competitive root-free alternatives, especially under low-to-moderate malicious-client ratios. Therefore, the appropriate defense should be selected according to the expected threat level, heterogeneity regime, and deployment assumptions regarding trusted reference data.

Globally, the study supports a layered and conservative interpretation of the proposed framework. Federated learning contributes data locality, robust aggregation contributes statistical resilience against poisoned updates, and blockchain contributes traceability and audit support. The observed robustness gains are not caused by blockchain directly, and the framework does not eliminate all adversarial or privacy risks. Rather, it provides a structured architecture and empirical comparison showing how different aggregation

mechanisms behave under controlled adversarial IoT-oriented FL conditions. This regime-aware interpretation is scientifically stronger than a universal superiority claim and better reflects the practical complexity of secure, privacy-aware distributed intelligence in resource-constrained IoT systems.

7. Conclusions

This paper investigated a blockchain-assisted robust federated learning framework for privacy-aware distributed intelligence in resource-constrained IoT systems. The study addressed the combined challenges of statistical heterogeneity, partial client participation, and adversarial model poisoning. The blockchain layer was positioned as an architectural support mechanism for trust coordination, traceability, metadata recording, and auditability, while the empirical contribution focused on the robustness of the aggregation layer. In the revised experimental campaign, six aggregation strategies were compared under a controlled UCI HAR federated learning protocol: FedAvg, Median, Trimmed Mean, FLTrust, trust-aware aggregation, and hybrid trust-trimmed mean.

The results show that FedAvg is highly vulnerable under severe sign-flip poisoning, particularly when client data are strongly non-IID. Classical robust statistical baselines provide partial protection, with Median improving substantially over FedAvg in the most severe heterogeneity regime. However, the strongest mean robustness under the 40% malicious-client stress test is achieved by FLTrust. This behavior is explained by the mechanism-level analysis, which shows that FLTrust assigns near-zero weights to malicious clients by using a clean server-side reference update. Thus, when a trusted and representative root dataset is available at the server, reference-based trust can provide strong protection against severe poisoning.

At the same time, the lower-intensity attack controls show that FLTrust is not uniformly dominant across all adversarial regimes. Under 10% and 20% malicious-client ratios, the trust-aware and hybrid trust-trimmed strategies achieve the highest or near-highest mean performance without requiring a clean server-side reference set. This is an important practical distinction. FLTrust offers strong robustness when trusted root data are available, whereas the trust-aware and hybrid strategies provide competitive root-free alternatives for settings where such reference data may be unavailable, undesirable, or difficult to maintain.

A central conclusion of this work is therefore regime-dependent rather than universal. No single aggregation strategy dominates all tested conditions. The most suitable defense depends on the malicious-client ratio, the severity of non-IID partitioning, the stability of benign client updates, and the availability of trusted server-side reference data. This finding strengthens the scientific interpretation of the study by avoiding an overgeneralized superiority claim and instead showing how different robust aggregation mechanisms behave under controlled adversarial IoT-oriented FL conditions.

The blockchain-assisted component should also be interpreted precisely. The observed predictive-performance differences are driven by the aggregation strategies, not by blockchain itself. In the proposed architecture, blockchain supports system-level trust-worthiness by enabling tamper-resistant recording of participation metadata, trust-score summaries, model-version identifiers, and audit trails. It does not directly prevent poisoned updates from damaging the global model, nor is it independently benchmarked in this work in terms of storage, latency, energy, consensus, or communication overhead. These aspects remain important directions for deployment-oriented research.

The privacy scope of the framework is similarly bounded. The framework is privacy-aware in the federated learning sense that raw client data remain local and are not centralized. However, it does not implement formal privacy-enhancing mechanisms such as differential privacy, secure aggregation, secure multiparty computation, homomorphic

encryption, or trusted execution environments. Consequently, the framework should not be interpreted as providing strict cryptographic protection against attacks such as model inversion, gradient leakage, or membership inference.

Future work should extend the present study in several directions. First, broader attack families should be considered, including adaptive poisoning, model-replacement attacks, backdoors, colluding sybil clients, and attacks explicitly designed to evade trust estimation. Second, the evaluation should be expanded to additional IoT datasets and application domains beyond UCI HAR. Third, larger multi-seed campaigns and broader non-IID regimes should be used to further characterize statistical variability. Fourth, the interaction between robust aggregation and formal privacy mechanisms should be investigated. Finally, the blockchain-assisted layer should be evaluated from a systems perspective, including on-chain/off-chain design, storage footprint, communication overhead, latency, energy cost, consensus behavior, and scalability.

Globally, the findings support a conservative but meaningful robustness claim: robust aggregation can improve federated learning resilience in adversarial, resource-constrained IoT settings, but its effectiveness depends strongly on the threat regime and deployment assumptions. The proposed framework contributes to this direction by separating data locality, aggregation robustness, and blockchain-assisted auditability into distinct architectural roles, thereby providing a clearer basis for developing robust, auditable, and privacy-aware distributed intelligence systems for IoT environments.

Author Contributions: Conceptualization, M.J.C.S.R. and C.S.; methodology, F.B.; software, F.B. and C.S.; validation, M.J.C.S.R., C.S. and F.B.; investigation, M.J.C.S.R.; resources, M.J.C.S.R. and F.B.; data curation, F.B.; writing—original draft preparation, M.J.C.S.R.; writing—review and editing, C.S. and F.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The code, configuration files, and aggregated experimental results supporting the findings of this study are openly available in Zenodo at <https://doi.org/10.5281/zenodo.19947613> URL (accessed on 8 July 2026). The original datasets used in the experiments are not redistributed and should be obtained from their official sources, as described in the repository documentation.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Nguyen, D.C.; Pham, Q.-V.; Pathirana, P.N.; Ding, M.; Seneviratne, A.; Lin, Z.; Dobre, O.; Hwang, W.-J. Federated Learning for Smart Healthcare: A Survey. *ACM Comput. Surv.* **2022**, *55*, 1–37. [\[CrossRef\]](#)
2. Briggs, C.; Fan, Z.; Andras, P. A Review of Privacy-Preserving Federated Learning for the Internet-of-Things. In *Federated Learning Systems: Towards Next-Generation AI*; Rehman, M.H.U., Gaber, M.M., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 21–50, ISBN 978-3-030-70604-3.
3. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A. Y Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017*; PMLR: Cambridge, MA, USA, 2017; pp. 1273–1282.
4. Kairouz, P.; McMahan, H.B.; Avent, B.; Bellet, A.; Bennis, M.; Nitin Bhagoji, A.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. Advances and Open Problems in Federated Learning. *Found. Trends Mach. Learn.* **2021**, *14*, 1–210. [\[CrossRef\]](#)
5. Li, T.; Sahu, A.K.; Talwalkar, A.; Smith, V. Federated Learning: Challenges, Methods, and Future Directions. *IEEE Signal Process. Mag.* **2020**, *37*, 50–60. [\[CrossRef\]](#)
6. Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H.B.; Patel, S.; Ramage, D.; Segal, A.; Seth, K. Practical Secure Aggregation for Privacy-Preserving Machine Learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017*; Association for Computing Machinery: New York, NY, USA, 2017; pp. 1175–1191.

7. Mothukuri, V.; Parizi, R.M.; Pouriyeh, S.; Huang, Y.; Dehghantanha, A.; Srivastava, G. A Survey on Security and Privacy of Federated Learning. *Future Gener. Comput. Syst.* **2021**, *115*, 619–640. [\[CrossRef\]](#)
8. Blanchard, P.; El Mhamdi, E.M.; Guerraoui, R.; Stainer, J. Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent. In *Proceedings of the Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
9. Yin, D.; Chen, Y.; Kannan, R.; Bartlett, P. Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates. In *Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018*; PMLR: Cambridge, MA, USA, 2018; pp. 5650–5659.
10. Fung, C.; Yoon, C.J.M.; Beschastnikh, I. *The Limitations of Federated Learning in Sybil Settings*; USENIX Association: Berkeley, CA, USA, 2020; pp. 301–316.
11. Ali, M.; Karimipour, H.; Tariq, M. Integration of Blockchain and Federated Learning for Internet of Things: Recent Advances and Future Challenges. *Comput. Secur.* **2021**, *108*, 102355. [\[CrossRef\]](#)
12. Al Asqah, M.; Moulahi, T. Federated Learning and Blockchain Integration for Privacy Protection in the Internet of Things: Challenges and Solutions. *Future Internet* **2023**, *15*, 203. [\[CrossRef\]](#)
13. Jiang, Y.; Ma, B.; Wang, X.; Yu, G.; Yu, P.; Wang, Z.; Ni, W.; Liu, R.P. Blockchain Federated Learning for Internet of Things: A Comprehensive Survey. *ACM Comput. Surv.* **2024**, *56*, 1–37. [\[CrossRef\]](#)
14. Nguyen, D.C.; Ding, M.; Pathirana, P.N.; Seneviratne, A.; Li, J.; Vincent Poor, H. Federated Learning for Internet of Things: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1622–1658. [\[CrossRef\]](#)
15. Imteaj, A.; Thakker, U.; Wang, S.; Li, J.; Amini, M.H. A Survey on Federated Learning for Resource-Constrained IoT Devices. *IEEE IoT J.* **2021**, *9*, 1–24. [\[CrossRef\]](#)
16. Li, H.; Ge, L.; Tian, L. Survey: Federated Learning Data Security and Privacy-Preserving in Edge-Internet of Things. *Artif. Intell. Rev.* **2024**, *57*, 130. [\[CrossRef\]](#)
17. Ferrag, M.A.; Friha, O.; Maglaras, L.; Janicke, H.; Shu, L. Federated Deep Learning for Cyber Security in the Internet of Things: Concepts, Applications, and Experimental Analysis. *IEEE Access* **2021**, *9*, 138509–138542. [\[CrossRef\]](#)
18. Zhu, H.; Xu, J.; Liu, S.; Jin, Y. Federated Learning on Non-IID Data: A Survey. *Neurocomputing* **2021**, *465*, 371–390. [\[CrossRef\]](#)
19. Karimireddy, S.P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; Suresh, A.T. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In *Proceedings of the 37th International Conference on Machine Learning, Virtual, 13–18 July 2020*; PMLR: Cambridge, MA, USA, 2020; pp. 5132–5143.
20. Liu, F.; Zheng, Z.; Shi, Y.; Tong, Y.; Zhang, Y. A Survey on Federated Learning: A Perspective from Multi-Party Computation. *Front. Comput. Sci.* **2023**, *18*, 181336. [\[CrossRef\]](#)
21. Xia, G.; Chen, J.; Yu, C.; Ma, J. Poisoning Attacks in Federated Learning: A Survey. *IEEE Access* **2023**, *11*, 10708–10722. [\[CrossRef\]](#)
22. Bagdasaryan, E.; Veit, A.; Hua, Y.; Estrin, D.; Shmatikov, V. How To Backdoor Federated Learning. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, Virtual, 26–28 August 2020*; pp. 2938–2948.
23. Baruch, G.; Baruch, M.; Goldberg, Y. A Little Is Enough: Circumventing Defenses for Distributed Learning. In *Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019*; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32.
24. Cao, X.; Fang, M.; Liu, J.; Gong, N.Z. FLTrust: Byzantine-Robust Federated Learning via Trust Bootstrapping. In *Proceedings of the 2021 Network and Distributed System Security Symposium, Virtual, 21–25 February 2021*.
25. Zhai, K.; Ren, Q.; Wang, J.; Yan, C.; Zhai, K.; Ren, Q.; Wang, J.; Yan, C. Byzantine-Robust Federated Learning via Credibility Assessment on Non-IID Data. *MBE* **2022**, *19*, 1659–1676. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Shawkat, M.; El-desoky, A.; Ali, Z.H.; Salem, M. Blockchain and Federated Learning Based on Aggregation Techniques for Industrial IoT: A Contemporary Survey. *Peer-To-Peer Netw. Appl.* **2025**, *18*, 192. [\[CrossRef\]](#)
27. Reyes-Ortiz, J.; Anguita, D.; Ghio, A.; Oneto, L.; Parra, X. Human Activity Recognition Using Smartphones Dataset; University of California, Irvine Machine Learning Repository. 2013. Available online: <https://archive.ics.uci.edu/dataset/240/human+activity+recognition+using+smartphones> (accessed on 8 July 2026).
28. Lim, W.Y.B.; Luong, N.C.; Hoang, D.T.; Jiao, Y.; Liang, Y.-C.; Yang, Q.; Niyato, D.; Miao, C. Federated Learning in Mobile Edge Networks: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2031–2063. [\[CrossRef\]](#)
29. Mhamdi, E.M.E.; Guerraoui, R.; Rouault, S. The Hidden Vulnerability of Distributed Learning in Byzantium. In *Proceedings of the 35th International Conference on Machine Learning (ICML 2018), Stockholm, Sweden, 10–15 July 2018*.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.