ZTA-FEDGUARD: A ZERO-TRUST DYNAMIC TRUST-WEIGHTED FEDERATED LEARNING FRAMEWORK FOR IOT INTRUSION DETECTION UNDER NON-IID DATA AND POISONING ATTACKS

NGO VAN NAM[1*]
[1]*Faculty of Engineering Technology, Hung Vuong University, Phu Tho Province, Vietnam*
Email address: ngovannam@hvu.edu.vn
**Corresponding author: Ngo Van Nam*
http://doi.org/10.51453/3093-3706/2026/1480

ARTICLE INFO

Received: 15/02/2026
Revised: 20/4/2026
Published: 28/4/2026

KEYWORDS

Federated learning;
Intrusion detection;
Zero Trust Architecture;
IoT security;
Poisoning attack;
Non-IID data;
Robust aggregation;
Trust-weighted aggregation.

ABSTRACT

The rapid proliferation of the Internet of Things (IoT) has increased the demand for intrusion detection systems (IDSs) capable of learning from multiple distributed data domains without centralizing sensitive network traffic. Federated learning is a suitable paradigm because it enables edge gateways, organizations, or administrative domains to train local models and share only model updates. However, real-world IoT environments do not satisfy the ideal assumptions of conventional federated learning. Device-level data are often non-independent and non-identically distributed (non-IID), network behavior evolves over time, and some devices or gateways may be compromised and may send malicious updates. This paper proposes **ZTA-FedGuard**, a dynamic trust-weighted federated learning framework based on Zero Trust principles for IoT intrusion detection. The central idea of ZTA-FedGuard is that no model update should receive implicit trust by default. Each client update must be continuously evaluated by geometric consistency, robust update-norm deviation, and its impact on a trusted calibration buffer. The resulting anomaly score is converted into a dynamic trust value and used in a gating mechanism to restrict or remove the influence of suspicious clients before aggregation. Unlike robust aggregation methods that rely mainly on coordinate-wise statistics, ZTA-FedGuard introduces the concept of **least aggregation privilege** into federated learning, thereby turning each training round into an access-control process over the global IDS model. The paper presents the threat model, mathematical formulation, algorithmic design, security analysis, deployment requirements, limitations, and future research directions. The work is positioned as a methodological and theoretical contribution focused on threat modeling, secure aggregation control, and deployability in adversarial IoT environments.

1. INTRODUCTION

The widespread adoption of IoT devices has significantly expanded the attack surface of modern networked systems. Smart cameras, industrial sensors, programmable logic controllers, smart-home devices, residential routers, and edge gateways continuously generate network flows that contain security-relevant behavioral traces. At the same time, many IoT devices have limited computational resources, irregular software-update cycles, weak security configurations, and heterogeneous deployment contexts. These characteristics make IoT environments attractive targets for network scanning, vulnerability exploitation, device compromise, botnet propagation, and distributed denial-of-service campaigns.

Modern network intrusion detection systems increasingly rely on machine learning to recognize abnormal traffic behavior. Traditional centralized training requires packet traces, flow records, or endpoint logs to be collected into a central data repository. In practice, this approach faces substantial barriers because network data may reveal user behavior, internal addresses, business processes, infrastructure configuration, or compliance-sensitive information. Communication overhead and cross-organizational data ownership further reduce the feasibility of raw-data sharing.

Federated learning partially addresses these limitations by enabling multiple clients to train models on local data and send only model updates to an aggregation server. FedAvg was introduced as a communication-efficient aggregation mechanism for decentralized data; the original formulation explicitly recognizes that edge-device data can be sensitive, large-scale, unbalanced, and non-IID [8]. In cybersecurity, federated learning is particularly attractive because it preserves a degree of data locality while enabling shared learning across multiple observation domains.

Nevertheless, replacing centralized training with federated learning does not automatically produce a secure IDS. First, IoT traffic is strongly non-IID. An IP camera, an industrial controller, and a residential router may observe different protocols, packet sizes, connection frequencies, and attack patterns. Second, IoT behavior is not static. Firmware updates, device replacement, new cloud services, seasonal usage patterns, and adaptive adversarial tactics can all cause concept drift. Third, federated learning creates a new attack surface: compromised clients can poison local data, flip labels, amplify gradients, or send carefully crafted updates intended to bias the global model. Prior work on poisoning attacks against federated-learning-based IoT IDSs shows that the distributed learning process itself can be exploited if the aggregation server over-trusts registered clients [5].

This paper argues that the design of federated IDSs for IoT must move from the assumption that “registered clients are trustworthy” to a posture in which **model updates are untrusted by default**. The Zero Trust Architecture described in NIST SP 800-207 emphasizes the removal of implicit trust based on network location and its replacement with continuous evaluation of subjects, assets, and resources [7]. When this principle is mapped to federated learning, each model update can be viewed as a request to access and modify the global model. The server should not grant aggregation influence merely on the basis of client identity or participation history. Instead, it must verify, score, and constrain the influence of every update in every communication round.

The contributions of this paper are threefold. First, it formalizes a zero-trust threat model for federated-learning-based IoT intrusion detection in which clients may be statistically heterogeneous, partially poisoned, and time-varying. Second, it proposes ZTA-FedGuard, a dynamic trust-weighted aggregation mechanism that combines three signal groups: update-direction deviation, robust update-norm deviation, and impact on a trusted calibration buffer. Third, it analyzes the security properties, deployment conditions, limitations, and future research agenda of the proposed framework as a conceptual and methodological contribution to secure collaborative intrusion detection.

2. RELATED WORKS

2.1 Machine-learning-based intrusion detection in IoT environments

Email: sjttu@tqu.edu.vn

An intrusion detection system aims to identify network behaviors that may violate security policy. In IoT settings, an IDS must often process high-volume traffic, heterogeneous device behavior, and resource constraints. Recent surveys on machine-learning-based IDSs emphasize that an effective system should not only achieve high accuracy but also reduce false alarms, detect novel attack variants, adapt to environmental changes, and remain deployable under operational constraints [2].

A central challenge in IoT intrusion detection is the gap between benchmark data and distributed operational data. In real deployments, each gateway observes only a narrow slice of the traffic space. Data at one client may consist primarily of camera flows, data at another client may reflect industrial sensors, and data at a third client may describe an unstable residential network. Consequently, models trained under centralized or IID assumptions may underestimate the difficulty of distributed deployment. Common challenges include class imbalance, rare traffic events, behavioral drift, incomplete labels, and delayed feedback from security analysts.

2.2 Federated learning for intrusion detection

Federated learning is viewed as a privacy-preserving collaboration mechanism because raw data do not need to leave local domains. In the IDS problem, a client may be an IoT gateway, an edge server, an organizational branch, or an independent operational domain. The aggregation server coordinates learning by distributing the global model, receiving local updates, and aggregating those updates into a new global model. The goal is to exploit distributed security knowledge without directly collecting sensitive traffic or logs.

FedAvg is the foundational aggregation mechanism in federated learning. In the original formulation, the server computes a data-size-weighted average of local model parameters, and McMahan et al. explicitly motivate the method for decentralized data that may be sensitive, large-scale, unbalanced, and non-IID [8]. A delta-update expression is often used as an algebraically equivalent implementation view when $\Delta_{k,t} = w_{k,t+1} - w_t$, but it should not be confused with the original parameter-averaging notation. FedProx extends this setting by adding a proximal regularization term, $\mu/2||w - w_t||^2$, to each client's local objective; this term restricts excessive drift from the current global model and permits clients to perform variable or partial local work under system and statistical heterogeneity [9]. In IoT intrusion detection, these approaches enable learning across multiple domains, but they still leave open a critical security question: should the server assume that returned updates are honest?

In cybersecurity settings, a client is not only a data source but also a potentially compromised entity. A compromised gateway may submit a malicious update while retaining a valid identity. Therefore, federated IDS design must jointly address two objectives: preserving data locality and protecting the integrity of the learning process. If the second objective is neglected, the global model may become a propagation point for corrupted behavior throughout the defense system.

2.3 Poisoning attacks and robust aggregation

Poisoning attacks manipulate the learning process to bias the resulting model. In federated learning, poisoning can occur at either the data level or the model level. At the data level, a malicious client may flip labels, inject noisy samples, or insert trigger-bearing samples. At the

model level, the client may submit amplified, reversed, or optimized updates designed to evade defense mechanisms. For IDSs, poisoning is especially dangerous because a biased model may miss attack traffic and create a persistence window for adversaries.

Robust aggregation mechanisms have been proposed to reduce the influence of Byzantine clients. Krum does not perform a simple majority vote; it selects a single update whose sum of squared distances to its $n - f - 2$ nearest neighboring updates is minimal, where f denotes the assumed upper bound on Byzantine clients. Multi-Krum generalizes this idea by selecting a subset of such updates before averaging [10]. Coordinate-wise Median and Trimmed Mean rely on robust coordinate-level statistics to reduce the influence of outliers [11]. Other approaches use divergence-based or robust-estimation techniques to constrain anomalous updates [13]. However, these mechanisms often assume that malicious updates are statistically distinguishable from honest updates.

The limitations of purely statistical robust aggregation are pronounced in non-IID IoT IDS environments. An honest client observing a rare device type may produce an update that differs from the majority. Conversely, an adversarial client may craft an update that remains close to the statistical center while still degrading detection for a targeted attack class. Thus, reliance on a single statistical criterion can create two risks: discarding legitimate minority-domain signals or accepting well-camouflaged malicious updates.

2.4 Zero Trust as a design principle for federated learning

Zero Trust Architecture is not a single product but a security design principle. NIST SP 800-207 defines Zero Trust as an approach in which no subject or resource is implicitly trusted merely because it resides inside a network perimeter; every access request must be continuously evaluated according to context, policy, and risk state [7].

Applying this principle to federated learning leads to a different perspective: a model update is an action that can influence a critical asset, namely the global IDS model. If an update is aggregated, it may change decision boundaries and affect the entire defense system. Accordingly, the right to influence the global model should be managed as a security privilege. Each client should receive only the amount of influence justified by its current trust state, and that trust state must be recomputed in every communication round.

In ZTA-FedGuard, a local update is treated as an access request to the global IDS model. The server grants only a bounded amount of influence after evaluating geometric consistency, statistical deviation, and impact on a trusted calibration buffer. This interpretation transfers the Zero Trust concepts of continuous verification, least privilege, and risk-adaptive access control into the aggregation layer of federated learning.

3. PROPOSED METHODOLOGY

3.1 Problem statement and threat model

Consider a federated IDS consisting of an aggregation server and a set of edge clients indexed by k. Each client k owns a local flow-record dataset $D_k = (x_i, y_i)$, where x_i is a feature

Email: sjttu@tqu.edu.vn

vector and $y_i \in 0,1$ denotes benign or attack traffic. The objective is to learn a global classifier parameterized by w, without transmitting raw records from clients to the server.

In each communication round t, the server sends the current model w_t to the selected clients. Each client performs local training and returns a local model or an equivalent update. In the original FedAvg notation, the global model is obtained as the data-size-weighted average $w_{t+1} = \Sigma_{k\in K}(n_k/n)w_{k,t+1}$, where $n = \Sigma_{j\in K} n_j$. The delta-update form shown below is an equivalent rewrite used in this paper for security scoring because $\Delta_{k,t} = w_{k,t+1} - w_t$:$\Delta_{k,t}$.

$$w_{t+1} = w_t + \Sigma_{k=1}^{K}\left(n_k/\Sigma_{j=1}^{K} n_j\right)\Delta_{k,t}$$

where n_k is the number of local samples at client k and $\Delta_{k,t}$ denotes the local model displacement relative to w_t. This delta notation is therefore an implementation-level representation, not a replacement of the original FedAvg parameter-averaging definition. The critical security assumption remains the same: returned client contributions are presumed sufficiently trustworthy to be averaged according to data size. In adversarial IDS environments, this assumption is unsafe because a malicious client may submit $\Delta_{k,t}$ that does not reflect honest optimization over local traffic data.$\Delta_{k,t}$

This paper considers four practical assumptions. First, data across clients are non-IID because each gateway observes different devices, protocols, and behavior patterns. Second, a subset $A \subset 1, \dots, K$ of clients may be malicious or poisoned. An adversary may flip local labels, inject misleading samples, amplify updates, or optimize updates to reduce model reliability. Third, the server maintains a small trusted calibration bufferD_c. This buffer is not used for centralized training; it is used only to evaluate update impact and tune deployment sensitivity. Fourth, the adversary does not control the aggregation server and cannot modify the trusted calibration buffer.

The defender's objective is to maintain stable detection capability while limiting the influence of suspicious clients. Unlike ordinary classification, IDS deployment must balance missed attacks and false alarms. Missed attacks create persistence opportunities for adversaries, whereas excessive false alarms reduce analyst trust and overload operations. Therefore, the aggregation mechanism must account for both statistical accuracy and operational security context.

3.2 Overview of ZTA-FedGuard

ZTA-FedGuard is built around three principles: **continuous verification**, **least aggregation privilege**, and **dynamic trust with decay and recovery**. Continuous verification means that every client update is scored in every round, regardless of whether the client previously participated legitimately. Least aggregation privilege means that even accepted clients influence the global model only according to both data size and trust weight, rather than receiving unconditional aggregation authority. Dynamic trust means that suspicious behavior reduces future influence, while consistent behavior can gradually restore influence over time.

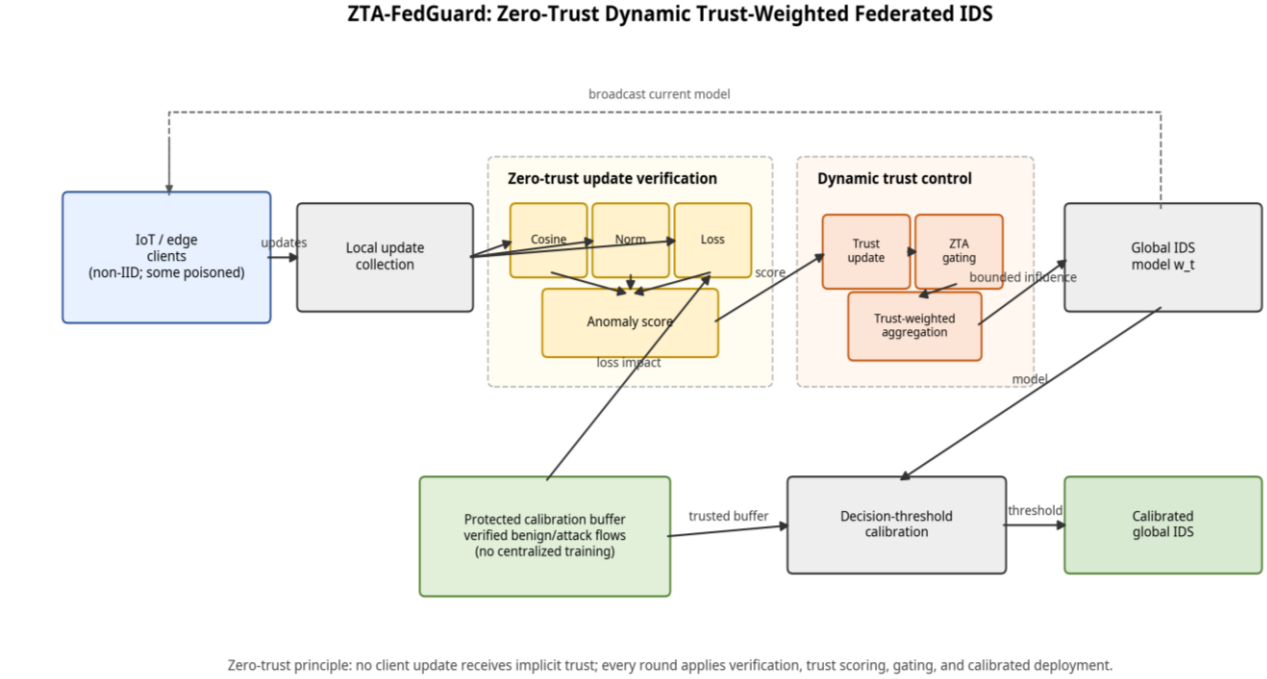

Figure 1: Architecture of ZTA-FedGuard

The aggregation server verifies each update through geometric consistency, update-norm deviation, and calibration-buffer impact. It then updates client trust, performs zero-trust gating, aggregates accepted updates by trust-weighted influence, and calibrates IDS thresholds for deployment.

3.3 Multi-signal update scoring

In round t, the server receives local updates $\Delta_{1,t}, \ldots, \Delta_{K,t}$. The first signal is geometric inconsistency. The server computes a robust update center m_t, which can be implemented as the coordinate-wise median of updates, and then measures cosine-based deviation:

$$S_{k,t}^{cos} = \left|1 - cos\left(\Delta_{k,t}, m_t\right)\right|$$

This signal detects updates whose direction deviates from the majority. In poisoning attacks, the adversary often needs to push the model in a direction that weakens the decision boundary; therefore, update direction is an important indicator. However, under non-IID data, an honest update may also deviate in direction if the client observes rare behavior. ZTA-FedGuard therefore does not use this signal alone to reject clients.

The second signal is robust update-norm deviation. $Let\ r_{k,t} = \left|\Delta_{k,t}\right|$. The server computes a robust z-score using the median absolute deviation (MAD):

$$S_{k,t}^{norm} = \left|\left(r_{k,t} - median(r_t)\right)/(1.4826 \cdot MAD(r_t) + \varepsilon)\right|$$

This signal identifies unusually large updates and is especially useful against update amplification or model-replacement attacks. The factor 1,4826 is approximately $1/\Phi^{-1}(3/4)$, where Φ is the cumulative distribution function of the standard normal distribution; it makes MAD a consistent estimator of the standard deviation when the underlying update distribution is close to

Email: sjttu@tqu.edu.vn

normal. In non-IID IoT settings, however, update distributions may be skewed, heavy-tailed, or multimodal, so this constant should be treated as a calibration heuristic rather than a universal statistical guarantee. The ε term prevents division by very small values. Importantly, update norm is not interpreted as absolute evidence of maliciousness but as one component of an overall risk score.

The third signal is impact on the trusted calibration buffer. Let $L(w; D_c)$ denote the loss on the calibration buffer. The server evaluates whether applying an update increases this loss:

$$S_{k,t}^{loss} = max\left(0, L\left(w_t + \Delta_{k,t}; D_c\right) - L(w_t; D_c)\right)$$

The calibration buffer is not a centralized training dataset. It is a small set of verified samples that serves as a consequence-testing anchor. This design is consistent with Zero Trust thinking: regardless of what a client claims about its data, the server evaluates the effect of the update on a trusted reference set.

The combined anomaly score is computed as a bounded weighted composition:

$$S_{k,t} = \alpha S_{k,t}^{cos} + \beta tanh\left(S_{k,t}^{norm}/3\right) + \gamma tanh\left(S_{k,t}^{loss}/3\right)$$

where $\alpha + \beta + \gamma = 1$. These weights are policy parameters and can be configured according to the environment. A system that prioritizes resistance against direction-shifting attacks may increase α, whereas a system with a high-quality calibration buffer may increase γ. The bounding function limits the influence of extreme values and prevents a single signal from dominating the entire decision.

3.4 Dynamic trust and zero-trust gating

Each client has a trust value $\tau_{k,t} \in [0,1]$. At initialization, clients should not receive absolute trust; instead, they should be assigned a moderate trust value to reflect the principle of no default trust. After scoring, trust is updated as:

$$\tau_{k,t} = \lambda \tau_{k,t-1} + (1 - \lambda)\left(1 - S_{k,t}\right)$$

where $\lambda \in [0,1]$controls system memory. If λ is large, trust changes slowly, reducing responsiveness to sudden attacks. If λ is small, the system responds quickly but may become sensitive to legitimate fluctuations. Selecting λ is therefore a policy decision that depends on the volatility of the IoT environment.

A client is admitted into the aggregation set G_t only if it satisfies the minimum trust threshold and its current anomaly score does not exceed the policy threshold. The ZTA-FedGuard aggregate update is defined as:

$$\Delta_t^{ZTA} = \Sigma_{k \in G_t}\left[\left(n_k \tau_{k,t+1}\right)/\left(\Sigma_{j \in G_t} n_j \tau_{j,t+1}\right)\right]\Delta_{k,t}$$

This formulation limits the influence of suspicious clients without necessarily removing them permanently from the system. This property is important in non-IID environments, where honest clients may temporarily appear anomalous because of device changes, operational events, or

rare but legitimate traffic. Trust recovery allows a client to regain influence if later updates become more consistent with the security policy.

3.5 Algorithmic summary

The ZTA-FedGuard workflow can be summarized as a sequence of zero-trust operations in which every training round is treated as an access-control process over the global IDS model.

Table 1. Operational Steps and Zero Trust Interpretation of the Proposed

Step	Operation	Zero Trust Interpretation
1	The server distributes the current global IDS model to clients.	Clients receive only the resources needed for local training.
2	Clients train on local data and submit updates.	Raw network traffic remains inside local administrative domains.
3	The server computes cosine, norm, and loss-impact scores.	Every update is verified; client identity alone is insufficient for trust.
4	The server updates trust and applies gating.	Trust is neither implicit nor permanent.
5	Accepted updates are aggregated by trust-weighted influence.	Least aggregation privilege constrains suspicious influence.
6	IDS thresholds are calibrated by trusted calibration data.	Deployment sensitivity is adjusted to operational requirements.

The algorithm can also be represented in terms of inputs, verification, scoring, gating, aggregation, and output.

Table 2. Functional Components of the Proposed Trust-Aware Federated Learning Algorithm

Component	Description
Input	Global model w_t, update set $\Delta_{k,t}$, local data sizes n_k, trust values $\tau_{k,t}$, and trusted calibration buffer D_c.
Verification	Compute $S_{k,t}^{cos}$, $S_{k,t}^{norm}$, and $S_{k,t}^{loss}$ for each client.
Scoring	Compute $S_{k,t}$ using a weighted and bounded composition.
Trust update	Compute $\tau_{k,t+1}$ using dynamic memory.
Gating	Retain only clients whose trust is sufficiently high and anomaly score sufficiently low.
Aggregation	Compute Δ_t^{ZTA} using trust-weighted aggregation.
Output	Update $w_{t+1} = w_t + \Delta_t^{ZTA}$ and store the new trust ledger.

3.6 Role of the trusted calibration buffer

Email: sjttu@tqu.edu.vn

The trusted calibration buffer distinguishes ZTA-FedGuard from many purely statistical aggregation mechanisms. In operational IDSs, this buffer can be built from analyst-verified alerts, high-confidence benign flows, attack samples from honeypots, analyzed incident records, or controlled internal test data. The buffer does not need to be large, but it should be diverse and strongly protected.

The main risks are staleness, bias, and poisoning of the buffer itself. If the buffer represents only one device type, honest updates from clients observing other devices may be unfairly penalized. Therefore, the buffer should be periodically audited, refreshed in response to concept drift, and managed through a dedicated security process. Architecturally, it should be treated as a high-value security asset because it directly influences the trust-scoring mechanism.

4. RESEARCH RESULTS

4.1 Security analysis

ZTA-FedGuard strengthens resistance to poisoning by requiring an adversary to pass multiple checks simultaneously. A malicious update must not only remain directionally close to the robust center but also keep its norm within a normal range and avoid degrading performance on the trusted calibration buffer. This increases the adversary's optimization cost compared with defenses based on a single statistic.

For update-amplification attacks, the norm signal is particularly important. An update with abnormal magnitude increases $S_{k,t}^{norm}$, leading to reduced trust. For attacks that shift the decision boundary while maintaining moderate magnitude, the cosine and loss-impact signals provide complementary detection capability. For temporarily poisoned clients or clients that become unusual because of legitimate environmental changes, dynamic trust helps avoid premature permanent exclusion.

ZTA-FedGuard is not a complete defense against every federated-learning attack. An adaptive adversary may attempt to craft updates that stay near the statistical center, maintain a normal norm, and evade the trusted calibration buffer simultaneously. Semantic backdoor attacks may also be difficult to detect if the trigger is absent from the calibration buffer. In addition, if malicious clients form a majority or collude to reshape the robust center, geometric signals may be weakened.

Another risk is privacy leakage from model updates. Federated learning reduces the need for raw-data sharing, but it does not automatically guarantee absolute privacy. Gradients or parameters may still be analyzed through membership inference or data-reconstruction techniques. Privacy protection must therefore be integrated carefully. Standard secure aggregation prevents the server from observing individual client updates; as a result, it conflicts with the per-client cosine, norm, and loss-impact scoring required by ZTA-FedGuard. In high-assurance deployments, this conflict must be resolved explicitly through secure-verification mechanisms such as verifiable computation, zero-knowledge proofs, trusted execution environments, or secure aggregation variants that expose only policy-approved scoring evidence. Differential privacy may also be used, but its noise budget must be evaluated against IDS sensitivity. This layered design is consistent with the broader

direction of Zero Trust for collaborative AI, where privacy, continuous authentication, and privilege control are treated as foundational requirements [3].

4.2 Methodological comparison

The following table summarizes the conceptual differences between ZTA-FedGuard and several common aggregation approaches. The comparison focuses on design mechanisms and does not imply an empirical performance ranking.

Table 3. Evaluation of Federated Aggregation Methods under Non-IID

Method	Aggregation Principle	Strength	Limitation in Non-IID IoT IDS
FedAvg	Data-size-weighted averaging of local model parameters.	Simple, communication-efficient, and grounded in the original decentralized-data formulation.	Sensitive to malicious or amplified updates because data-size weight is not a security trust signal.
Krum	Selects one update with the minimum distance sum to its n−f−2 nearest neighbors.	Provides explicit Byzantine-tolerance reasoning under a bounded adversary assumption.	May discard legitimate minority-domain signals and can be weakened when non-IID clients are far from the statistical center.
Coordinate-wise Median	Uses the median for each parameter coordinate.	Reduces the effect of coordinate-level outliers.	Does not evaluate the security impact of an update on IDS behavior.
Trimmed Mean	Removes coordinate-level extreme values and averages the remainder.	Can resist boundary noise in each coordinate.	Depends on the trimming ratio and can be evaded by sophisticated poisoning.
ZTA-FedGuard	Uses multi-signal scoring, dynamic trust, least-privilege gating, and trusted calibration evidence.	Combines geometric, statistical, and contextual IDS-impact checks as a security decision.	Requires a protected calibration buffer, tuned thresholds, and special design if secure aggregation hides individual updates.

Compared with traditional robust aggregation, ZTA-FedGuard does not treat aggregation as a purely statistical operation. Instead, aggregation is treated as a controlled security decision. This perspective aligns with IDS deployment, where even small model deviations may have significant operational consequences. It also facilitates integration into security operations center workflows, because each down-weighting or gating decision can be logged as a security event for analyst review.

4.3 Practical deployment recommendations

A practical ZTA-FedGuard deployment should include three layers of control. The first layer is technical control over the federated-learning process, including client authentication,

channel protection, update verification, training-session management, and audit logging. The second layer is calibration-data control, including trusted-sample selection, buffer protection, change auditing, and behavioral diversity assurance. The third layer is operational control, including integration of trust scores into SOC processes, alerting on anomalous clients, and coordination with incident response.

The aggregation server should maintain a trust ledger for clients. This ledger should store not only $\tau_{k,t}$, but also the score components $S_{k,t}^{cos}$, $S_{k,t}^{norm}$, and $S_{k,t}^{loss}$, gating decisions, and reasons for down-weighting. Such information helps analysts distinguish among compromised clients, misconfigured clients, clients affected by concept drift, and clients observing rare but legitimate traffic.

Gating should not be understood as permanent blocking. In IoT environments, legitimate volatility may temporarily move a client away from the majority. Therefore, policy should support temporary privilege reduction, requests for additional verification, or transition to enhanced monitoring. Only when additional operational evidence exists should the system isolate a client or revoke its right to participate in federated learning.

The communication overhead of ZTA-FedGuard does not require raw data or large additional artifacts to be transmitted. The main cost is incurred at the server when scoring updates and evaluating calibration-buffer impact. For small and medium IDS models, this cost is manageable. For deep models with many parameters, layer sampling, block-wise scoring, or update embeddings may be required to reduce computational overhead. A deployment that mandates secure aggregation must not simply enable it as an independent privacy layer, because hiding individual updates disables direct per-client trust scoring. Such a deployment requires a compatible verification architecture, for example a TEE-protected scoring enclave, zero-knowledge proof of bounded update properties, or verifiable computation that proves policy compliance without exposing full model updates.

4.4 Limitations of the proposed framework

The first limitation is dependence on the quality of the trusted calibration buffer and statistical calibration constants. If the buffer is too small, class-biased, unrepresentative of new devices, or poisoned, the loss-impact score may introduce bias. Similarly, the MAD scaling factor 1,4826 is theoretically justified under approximate normality, but IoT update distributions in non-IID environments can be skewed, heavy-tailed, or multimodal. A robust deployment should therefore validate the scaling factor empirically, consider domain-specific quantile calibration, and maintain multiple buffers by device class, network domain, or label-confidence level. This issue is especially important in IDSs because both benign and attack behaviors evolve over time.

The second limitation is adversarial adaptivity. An adversary that understands the scoring mechanism may optimize updates to reduce all three score components simultaneously. To address this risk, the system should vary policies over time, add randomized checks, combine model-layer-specific tests, and monitor long-term behavioral sequences. Detection of collusion among multiple clients also requires further study.

The third limitation is the safety-availability trade-off. Overly strict gating may exclude honest clients during periods of data volatility, whereas overly permissive gating may accept malicious updates. ZTA-FedGuard should therefore be viewed as a tunable policy framework rather than a fixed parameter set applicable to all environments.

5. CONCLUSION AND FUTURE DEVELOPMENT

This paper proposed ZTA-FedGuard, a dynamic trust-weighted federated learning framework based on Zero Trust principles for IoT intrusion detection under non-IID data and poisoning risk. The framework treats every local update as an access request to the global model and grants influence only after multi-signal verification. The three scoring components are cosine inconsistency, robust update-norm deviation, and impact on a trusted calibration buffer. The anomaly score is transformed into dynamic trust and then used for gating and trust-weighted aggregation.

The key contribution of ZTA-FedGuard is the introduction of no-default-trust and least-privilege principles into the aggregation layer of federated learning. This approach fits IoT IDS settings, where clients are both distributed learning sources and potential points of compromise. The work has been presented as a methodological and theoretical contribution. Future studies can extend the framework through empirical evaluation across multiple data domains, analysis against adaptive adversaries, and standardization of operational procedures for international publication and deployment.

Several future research directions are particularly important. First, ZTA-FedGuard should be extended to deep IDS models and multi-layer architectures. Deep models provide richer representation capacity, but they also complicate update scoring. Future studies may examine anomaly scores by layer, attention block, prototype, or hidden representation. Second, concept-drift-aware continual learning should be integrated. In IoT, normal behavior changes continuously; therefore, client trust must distinguish legitimate drift from poisoning. A potential solution is to combine class-wise prototype memory, local drift tests, and conditional trust-recovery policies. Third, backdoor defense in federated IDSs should be studied in greater depth. Backdoors may not reduce general performance but may cause misclassification under specific triggers. ZTA-FedGuard can be extended with trigger-oriented checks, activation analysis, or class-consistency tests. Fourth, privacy-preserving verification must be treated as a first-class research problem rather than a post-hoc add-on. Conventional secure aggregation hides individual updates and therefore prevents the server from computing per-client cosine, norm, and calibration-loss signals directly. Future work should design ZTA-FedGuard-compatible protocols based on zero-knowledge proofs, verifiable computation, secure enclaves, or hybrid protocols that disclose only bounded trust evidence while protecting update contents. Fifth, operational standards for Zero-Trust federated IDSs should be developed, including procedures for creating calibration buffers, maintaining trust ledgers, integrating with SIEM/SOAR, handling down-weighted clients, and auditing automated decisions.

REFERENCES

1. W. Gharibi, “Machine Learning and Cybersecurity—Trends and Future Challenges,” *Electronics*, 2025. https://doi.org/10.3390/electronics14204007

2. A. Hozouri, A. Mirzaei, and M. Effatparvar, "A comprehensive survey on intrusion detection systems with advances in machine learning, deep learning and emerging cybersecurity challenges," *Discover Artificial Intelligence*, 2025. https://link.springer.com/article/10.1007/s44163-025-00578-1

3. K. Li et al., "Zero-Trust Foundation Models: A New Paradigm for Secure and Collaborative Artificial Intelligence for Internet of Things," *arXiv*, 2025. https://arxiv.org/html/2505.23792v1

4. M. B. Bankó et al., "Advancements in Machine Learning-Based Intrusion Detection in IoT: Research Trends and Challenges," *Algorithms*, 2025. https://doi.org/10.3390/a18040209

5. T. D. Nguyen, P. Rieger, M. Miettinen, and A.-R. Sadeghi, "Poisoning Attacks on Federated Learning-based IoT Intrusion Detection System," *DISS Workshop / NDSS*, 2020. https://www.ndss-symposium.org/wp-content/uploads/2020/04/diss2020-23003-paper.pdf

6. H. Peng, C. Wu, and Y. Xiao, "FD-IDS: Federated Learning with Knowledge Distillation for Intrusion Detection in Non-IID IoT Environments," *Sensors*, 2025. https://doi.org/10.3390/s25144309

7. S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture," *NIST Special Publication 800-207*, 2020. https://doi.org/10.6028/NIST.SP.800-207

8. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," *AISTATS / PMLR*, 2017. https://proceedings.mlr.press/v54/mcmahan17a.html

9. T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated Optimization in Heterogeneous Networks," *MLSys*, 2020. https://proceedings.mlsys.org/paper/2020/hash/1f5fe83998a09396ebe6477d9475ba0c-Abstract.html

10. P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent," *NeurIPS*, 2017. https://papers.nips.cc/paper/6617-machine-learning-with-adversaries-byzantine-tolerant-gradient-descent

11. D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates," *ICML / PMLR*, 2018. https://proceedings.mlr.press/v80/yin18a.html

12. M. Fang, S. Nabavirazavi, Z. Liu, W. Sun, S. Iyengar, and H. Yang, "Do We Really Need to Design New Byzantine-robust Aggregation Rules?" *NDSS*, 2025. https://www.ndss-symposium.org/ndss-paper/do-we-really-need-to-design-new-byzantine-robust-aggregation-rules/

13. C.-J. Li, P.-H. Huang, Y.-T. Ma, H. Hung, and S.-Y. Huang, "Robust Aggregation for Federated Learning by Minimum γ-Divergence Estimation," *Entropy*, 2022. https://doi.org/10.3390/e24050686