# 3T-ZTFL: A Three-Tier Zero Trust Federated Learning Framework for Privacy-Preserving and Byzantine-Resilient Intrusion Detection in Resource-Constrained IoT Critical Infrastructure

**King David Kamara[1]*, Mohamed Syed Fofanah[2]*, Maurice Sesay[3]*, Mohamed Koroma[4]***

**[*1]Faculty of Engineering, Department of Computer Science, Freetown Polytechnic, Freetown, Sierra Leone, West Africa**

**[*2,3,4],School of Technology, Department of Computer Science and Information Technology, Njala University, Sierra Leone, West Africa**

***Corresponding Author**

**DOI: https://doi.org/10.51584/IJRIAS.2026.11100060**

**Received: 18 October 2026; Accepted: 23 October 2026; Published: 05 November 2026**

## ABSTRACT

New and rapidly evolving trends in IoT devices that rely on resources have broadened the threat surface beyond the perimeter defenses to include what's known as critical infrastructure. Zero Trust Architecture (ZTA) is one solution to this problem, while federated learning (FL) is the other; both solutions can't be readily integrated with endpoints that have only kilobytes of RAM in storage and intermittent connectivity. We introduce the three-tier Zero Trust federated learning architecture, 3T-ZTFL, which distributes the verification across the device tier, edge tier and coordination tier. The lightweight anomaly detector (368 KB, under our 1MB limit, is based on depth-wise separable convolutions, magnitude pruning and 8-bit quantization), the edge aggregates models updates and implements micro-segmentation policies, and the coordination tier keeps the global model and trust knowledge base. Byzantine resilience is achieved with gradient auditing, and privacy is achieved with a client-side differential privacy which is configurable by the operators. In all four benchmark datasets (N-BaIoT, Edge-IIoTset, CICIoT23 and IoT-23), the percentage of the clients correctly detected is between 97.5% and 98.7%, and can be achieved even with 50% malicious clients. The inference time is on average 82 ms on a Raspberry Pi 4 and the privacy cost (1.1 percentage points of accuracy at $\varepsilon = 5.0$) is negligible. Particularly, gradient auditing can enhance the accuracy of clean data by 1.3–1.4 percentage points, and also be Byzantine robust, without any conflict between the two in terms of performance. The results indicate that such a successful, secure and privacy preserving detection is possible on the hardware representative of the critical infrastructure. There is a need for field validation.

**Keywords:** Zero Trust Architecture; Federated Learning; IoT Security; Differential Privacy; Byzantine Resilience; Critical Infrastructure Protection

### Highlights

- 3T-ZTFL unifies Zero Trust access control with federated learning for IoT security.
- Detection accuracy of 97.5%–98.7% across four datasets; model size of 368 KB.
- Client-side DP at $\varepsilon = 5.0$ costs only 1.1 percentage points of accuracy.
- Gradient auditing preserves >92% accuracy with 50% malicious clients.
- Ablation quantifies each component's contribution to accuracy and efficiency.
- Verification load is distributed across device (5 ms), edge (15 ms), and coordination tiers.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

# INTRODUCTION

The Internet of Things (IoT) has progressed from a prototype to the mainstay of today's society over the last ten years. From electricity grids that adjust load on demand using connected sensors, to water utilities that monitor chemical levels across thousands of sites, to hospitals that monitor patient conditions with networked devices, to transport systems that coordinate thousands of endpoints that are spread throughout the system. While the security implications are real, the efficiency, responsiveness and cost savings are equally real, and weren't the point of perimeter-based defenses.

The Mirai botnet of 2016 is the most recent example of the potential of insecure IoT devices when in the wrong hands. According to Antonakakis et al. 2017, Mirai was used to attack approximately 600,000 machines to attack large Internet services in North America and Europe. It wasn't the devices that were being aimed at – it was the ammunition. This is the same for the latter campaigns: the Gorilla botnet for example, worked for years instead of days (Weyns et al., 2026). With the smartness of the infrastructure, Mirai and the attacks that ensued have illustrated how easily it can be taken down.

The cornerstone of enterprise and industrial network security has always been perimeter defense, and there has been an implicit assumption of static, physical and logical assets in the network. This is no longer the case. Critical infrastructure networks today are global, inter-administerial and filled with dozens of devices from dozens of vendors, each having a diverse security approach. There needs to be seamless operations that are continuous and as the boundary itself fades so does the protection it offers (Spencer & Pizio, 2024). Zero Trust Architecture (ZTA) comes as a solution. All entities must be verified and authorized for access, if there is any doubt at all, as there is with critical infrastructure devices, which can be hacked and the credentials can be stolen (Gumbo & Almulhem, 2026; Liu et al., 2024). But in the case of continual verification, it is costly. There are many IoT devices that are embedded systems with limited computation, memory and energy, which cannot handle the authentication, authorization and policy evaluation required for an Industrial IoT (IIoT) intrusion detection system (Albalawi et al., 2026).

Machine learning is one way to making Zero Trust possible for IoT. In addition to the ability to differentiate between normal and abnormal traffic, ML models have also the capability to identify the behaviours of the devices which do not correspond to normal access patterns (Chen et al., 2026; Pathak & Panteli, 2026). However, there are some drawbacks to using ML. This requires representative data for models and that data may include sensitive operational data in critical infrastructure, which can be seen as a high-value target and a probable violation of the data protection law (Murugan et al., 2025). One way to solve this problem is to follow the federated learning (FL) where local models collect knowledge without absorbing raw data in the central model (McMahan et al., 2017). Local training and sharing of model updates help to maintain data at the local level whilst allowing for collective learning.

However, there has been a substantial amount of work done in the area of access control (Zero Trust access control), privacy-preserving federated learning (PPFL) and lightweight model design for constrained devices (LMDC) which have grown somewhat independent. The focus of the Zero Trust research is on Zero Trust architecture design and policy enforcement (Gumbo & Almulhem, 2026). In Federated learning for intrusion detection, federated training and aggregation, and privacy mechanisms (Rezaei et al., 2025) are the key points. The most significant challenges in lightweight ML are the deployment of an edge device and the compression of its model (Albalawi et al., 2026). Coupling all three components has received few studies and the challenges of an IoT deployment in critical infrastructure are often overlooked (Liu et al., 2024).

This creates a vacuum which is experienced by the operator. When an operator is interested in federated learning for privacy, they will need to come up with a way to integrate it with Zero Trust access control, and won't be able to do that with the current tools, as it's not a straightforward mapping. Energy, water, healthcare and transportation are key elements of the modern society, disruption of any of these would pose a threat to public safety, economic stability and national security (Michalec, 2026). Security solutions to be deployed on the actual devices that execute these sectors are needed which would be deployable, privacy-preserving and effective.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

This work has one main research question and four sub questions. The big challenge is: what is a privacy preserving machine learning solution that can be deployed with Zero Trust access control to detect threats in critical infrastructure that is IoT enabled without consuming computational power of the constrained devices? The sub-questions follow. RQ1: What happens to Zero Trust verification continuous authentication, micro-segmentation, and dynamic access decisions when they are spread across all layers between the device, edge and coordination layers, and what is the result of the detection? RQ2: Is it possible to use federated learning to maintain high accuracy in detecting the accuracy of the operational data while preserving privacy? RQ3: What are the advantages of architecture and model optimization for threat detection in the case of IoT devices with limited processing, memory and energy resources? RQ4: How to assure strong client opposition but still ensure good data cleanliness?

Four goals are proposed to address these questions: (1) design a resource-adaptive Zero Trust architecture for dealing with the hardware constraints of devices; (2) design a Federated learning (FL) approach for Anomaly detection without revealing sensitive operations information; (3) design a protocol that is lightweight and communication-efficient for implementation on resource-constrained devices; and (4) assess the framework with benchmark datasets and evaluate its performance in terms of anomaly detection accuracy, privacy preservation, computational efficiency, and adversarial robustness.

The paper contributes four ways. There are four contributions to the work. First, it offers a unified framework that brings together three research areas that had been standing somewhat alone and importantly, establishes the nature of their interaction – detection signals influence access control decisions, while trust scores control federated aggregation. Second, it empirically demonstrates that adversarial robustness (SecureDyn-FL, 2025) is not the only benefit of a gradient audit that adapts to threats: continuous verification in conjunction with adaptive threat detection can lead to better quality of clean data model. Third, it provides design principles for authentic scenarios of critical infrastructure verification load distribution, budget calibration for privacy, and trade-offs for optimization, applied to four benchmark datasets. Fourth, it provides a discussion about the regulatory drive for privacy, without compromising on security capability: moderate privacy budgets are possible and effective and have been relevant for operators facing GDPR and NISD requirements (Chen et al., 2026; Michalec, 2026).

The study concentrates on CI important sectors that have stringent operational needs and critical information. Not exhaustive list of deployments or attack types of IoT. The evaluation employs four benchmarks: N-BaIoT, Edge-IIoTset, CICIoT2023 and IoT-23, related to botnet attack traffic, malware attack traffic, and volumetric attack traffic. None refers to zero day attacks and advanced persistent threats (APTs); this is detailed in Section 6.3. The environment was not a real critical infrastructure. The experiments are repeatable, repeatable, and controlled, but the experiments would not be able to include all of the critical infrastructure aspects that occur during operations, such as legacy system integration and human factors. The privacy evaluation is based on the membership inference, gradient reconstruction and property inference are discussed but not evaluated empirically. This is a conscious scope judgement and a future work direction.

The remaining sections of the paper follow. To set up the background of related studies, Section 2 examines existing research on Zero Trust security for IoT, privacy-preserving federated learning for intrusion detection and lightweight machine learning for resource-constrained devices, highlighting the research's gap. The problem formulation and threat model are developed in section 3. Section 4 introduces the 3T-ZTFL architecture, its mechanisms of privacy, Byzantine resilient aggregation and complexity analysis. Experimental evaluation of this is discussed in Section 5. Implications, limitations and future directions are discussed in section 6. Section 7 concludes.

## LITERATURE REVIEW

### Zero Trust Security in IoT-Integrated Critical Infrastructure

Zero Trust isn't a new concept. Evolved over time as the perimeter model proved less effective in securing increasingly complex digital systems with more and more vague boundaries. In a systematic review, Gumbo and Almulhem (2026) discovered that there are a number of different implementations of ZTA, but it has one

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

common theme: never trust, always verify. Access is granted on a case by case basis and only when it is necessary. Its conceptual roots are linked to the doctoral dissertation of Marsh (1994) computational concept of trust and the Zero Trust Architecture by Kindervag for Forrester since 2014 (Fernandez & Brazhuk, 2024). The importance of this lineage is less than the change in thinking it embodies: Zero Trust is not just a product or technology stack. The "trust by location" is replaced by the "trust that must be earned" model that includes authentication, authorization and context evaluation. As noted by Fernandez and Brazhuk (2024) in their critical analysis, the benefits of Zero Trust are not a magic bullet, and consistency and accuracy in the process of verifying identity and enforcing policy can decrease external threats and the risk of policy violation from a compromised insider.

The use case for IoT and industrial control systems (ICS) has garnered a lot of interest, but with mixed results. They are applied in various fields such as power systems, water supply systems, and factories for continuous verification (Kumar & Sisodiya, 2025). This report raises many of the common challenges faced by legacy systems, real-time data processing and staff competence, and tells you that Zero Trust is achievable for ICS, but it's hard to do well. An additional line of research is the use of Zero Trust in low power edge devices: cryptographic verification and resource limitation can clash. Solving this paradox of energy security is often not a breeze. There are proposals that move beyond the software based identity verification that can be compromised, and move toward a hardware based approach like hardware anchoring, and physical unclonable functions, but it is not yet clear which are viable in the existing infrastructure.

Industry analysis, which looks at the structure of the environment, reveals three obstacles to Zero Trust success in IoT and operational technology (OT) environments: lack of visibility into the environment, flat networks that seem only segmented, and legacy configurations and trust relationships based on firmware (Spencer & Pizio, 2024). In the IoT area there are three recurrent problems. First, connectivity: Critical infrastructure IoT devices typically have a poor ability to connect reliably, and cloud-based Zero Trust configurations don't expect them to; per-request authentication is challenging if you don't have connectivity, and verifying trust can become a single point of failure. Secondly, computational cost: while continuous authentication is one way to prevent session hijacking and insider threats, it can be impractical on deeply embedded devices. Third, architectural: most Zero Trust IoT deployments are not connected to threat detection, but rather they view threat detection and architecture as two different security jobs. This separation makes it difficult to react to new threats and makes threat detection lack context information.

Table 1: Comparison of Related Work Across Six Security and Deployment Dimensions

| Study | ZTA | FL | Privacy | Byzantine Resilience | Lightweight | IoT Focus |
|---|---|---|---|---|---|---|
| Gumbo & Almulhem (2026) | ✓ | × | × | × | × | ✓ |
| SecureDyn-FL (2025) | × | ✓ | ✓ | ✓ | × | ✓ |
| Ghimire & Lodh (2025) | × | ✓ | ✓ | ✓ | × | ✓ |
| Rezaei et al. (2025) | × | ✓ | ✓ | × | × | ✓ |
| Proposed (3T-ZTFL) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

*Note.* ZTA = Zero Trust Architecture; FL = Federated Learning. "Lightweight" indicates a reported model size below 1 MB or direct deployment on constrained edge devices. Checkmarks indicate that a dimension is explicitly addressed in the cited study.

Table 1 makes the point plainly. Existing literature predominantly addresses one or two dimensions; none covers all six. The proposed framework fills that space, and the rest of the paper tests whether it does so credibly.

**Privacy-Preserving Federated Learning for Intrusion Detection**

The concept of federated learning was proposed by McMahan et al. (2017) for distributed training of a shared model without having to centralize the data among multiple clients. In critical infrastructure, this attribute will not only be convenient, but it may even be legally mandated: Traffic patterns may expose the operator to details about processes, device inventories, control schedules, and more that the operator will be legally required to guard. Rezaei et al. (2025) provide a survey on FL-based intrusion detection for 5G and beyond, highlighting that most of the systems adopt federated averaging and some secure aggregation or differential privacy, but not both, and even less often, with Byzantine resilience. Both SecureDyn-FL (2025) and Ghimire and Lodh (2025) are exceptions: they incorporate adversarial robustness into privacy-preserving FL for IoT intrusion detection. Their designs, however, rely on mechanisms in the server side, and ignore the constrained-device deployment problem, where the endpoints have to host the detection model.

This literature is characterised by two tensions. The first is privacy-utility: if you add noise to updates, then you will protect privacy of the client, but the global model will worsen, and the cost and reward at low privacy budgets is extremely high. The second is robustness-utility which can be summarized by the following statement: a system that is robust against poisoning can be more underperforming than a plain FedAvg model on clean data. It is demonstrated in Section 5.7 that the second tension is a chance and not a hindrance to create a framework.

**Lightweight Machine Learning in Resource-Constrained Environments**

It is important to understand that constraints are built into IoT devices and any security mechanism that runs on it needs to fit in these constraints. Deep learning models contain numerous parameters, activations that need to be stored in memory, require significant computation during inference and consume a significant amount of energy for both storing and performing inference, which are beyond the capability of an IoT endpoint. There are some proposed solutions for lightweight architectures, feature selection and model compression (Han et al., 2016; Howard et al., 2017; Jacob et al., 2018). Howard et al. (2017) demonstrated that using depth-wise separable convolutions, the number of parameters and operations can be reduced by about 8-9× without sacrificing classification accuracy by breaking down a standard convolution into a depth-wise convolution and a point-wise convolution. Jacob et al. (2018) proved that 8-bit integer quantization may reduce the number of model bits by about 75% whilst accelerating inference on integer-arithmetic processors and Han et al. (2016) illustrated that weights that are below a threshold can be discarded. In this paper, the lightweight model is directly developed based on these techniques.

**Synthesis and the Integration Gap**

A pattern is starting to form but it's not finished. Even though the research has yielded promising architecture concepts and access control policies, the architectures are complex and difficult to apply and integrate with learning-based threat detection (Gumbo & Almulhem, 2026; Kumar & Sisodiya, 2025). While FL research provides good solutions to privacy and aggregation issues, FL systems tend to lack knowledge of access control as an input to the detection decision, and the access control decision does not affect the detection results (Rezaei et al., 2025; Bilal & Islam, 2026). While ML is lightweight with the help of compression techniques, the security architecture in which the model will be deployed is typically not included in the model evaluation (Albalawi et al., 2026). The intersection of three areas – namely, Zero Trust access control, federated learning with privacy, and lightweight deployment under real IoT constraints – is relatively unexplored.

Three gaps follow. First, it is not a system that provides a clear, coherent picture of the three streams and how they function together (just as much as they function independently). Second, research prototypes are not always deployable: while some studies state that they are suitable for resource-constrained environments, few studies offer evidence in real hardware. Third, while the privacy-utility trade-off has been characterized for various combinations of Zero Trust mechanisms and differential privacy, it is not known for the specific combination of Zero Trust mechanisms and differential privacy for particular application in a critical infrastructure context that needs a dedicated study, because detection latency, the energy cost of Zero Trust

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

mechanisms, and the cost of false negatives are application-dependent. All three gaps are addressed in this paper.

## Problem Formulation and Threat Model

### System Model

We consider a critical infrastructure deployment organized into three tiers: a device tier of resource-constrained IoT endpoints, an edge tier of aggregation and policy nodes, and a coordination tier that maintains the global model and trust knowledge base. Formally, let K = {1, …, K} denote the set of client devices, E = {1, …, E} the set of edge nodes, and C the coordination server. Each client k holds a local dataset D_k = {(x_i, y_i)}_{i=1}^{n_k}, where x_i ∈ ℝ^d is a preprocessed feature vector and y_i ∈ {0, 1} is a binary label (0 = benign, 1 = attack). The global objective is to learn a model parameterized by θ that minimizes the empirical risk:

$$\min_\theta F(\theta) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(\theta), \quad F_k(\theta) = \frac{1}{n_k} \sum_{i=1}^{n_k} \ell(\theta; x_i, y_i)$$

where n = Σ_k n_k and ℓ(·) is the cross-entropy loss. In each federated round t, a fraction p of clients is selected, local updates θ_k^(t) are computed, and the server aggregates them. No raw data leaves the client.

### Threat Model

We draw on the FL threat taxonomy of SecureDyn-FL (2025) and Huma et al. (2026) to consider two classes of adversaries.

**Honest-but-curious aggregation server.** The server follows the protocol but may attempt to infer client data from received updates. This motivates client-side differential privacy, in which noise is added before an update leaves the client, so the server never sees a plain gradient.

**Malicious clients.** Up to 50% of participating clients may be malicious. An adversary may (a) poison updates through label flipping or gradient manipulation, (b) attempt to extract membership information from the global model, or (c) collude with other malicious clients to attack the global model. At the coordination tier we assume secure communication channels and trusted execution. Physical device tampering and hardware-level attacks are out of scope.

### Design Goals and Constraints

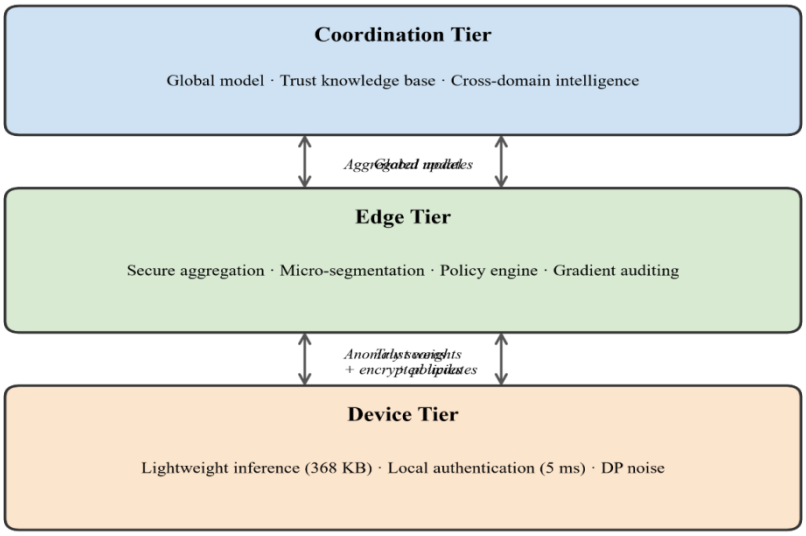
Four goals and three constraints frame the design. The goals are: (G1) distribute Zero Trust verification across tiers without overwhelming any tier; (G2) keep detection accuracy above 97% while holding inference latency below 100 ms and model size below 1 MB on a representative edge device; (G3) keep detection accuracy above 90% with up to 50% malicious clients; and (G4) preserve privacy acceptable to critical infrastructure operators without degrading detection below operational thresholds. The constraints are: (C1) model artifacts must fit within 1 MB of device memory; (C2) device connectivity may be intermittent, addressed by sampling a small fraction of clients per round; and (C3) regulatory privacy requirements, addressed by an operator-configurable differential privacy budget.

## The 3T-ZTFL Framework

### Architecture Overview

The 3T-ZTFL framework comprises three tiers device, edge, and coordination and distributes Zero Trust verification and federated learning computation across them according to the resources available at each. This avoids placing excessive burden on the resource-poor endpoint. Figure 1 shows the architecture and the flow of data, model updates, and policies between tiers.

**3T-ZTFL: Three-Tier Zero Trust Federated Learning Architecture**




Lightweight local inference is done in the device tier (bottom). Micro-segmentation and secure aggregation are the capabilities of the edge tier (middle Tier). The coordination tier (top) takes care of the global model and trust knowledge base. Anomaly scores and different models updates that are encrypted go up from devices, and policies and global model parameters go down. Bidirectional arrows between the edge and device tiers represent the trust-scoring loop: Trust information goes from the devices to the edge to make access decisions and from the edge to the devices to decide on the next round of aggregation. This loop is the one thing that makes 3T-ZTFL different from those systems that operate separate access control and threat detection solutions.

**Device Tier: Lightweight Detection**

The device tier is the end points in an IoT operational environment which are resource constrained. There are two functions done by each device. It gathers information from the local information network – traffic and access requests – and other behavioral signals from its function. Second, it is based on a quantized neural network based anomaly detector which categorizes samples as normal or suspicious. This tier is different from a traditional intrusion detection design in two ways. Only anomaly scores and model updates are sent out and the device doesn't send out any raw data, maintaining data sovereignty (Chen et al., 2026). Moreover, the detection model is small enough (Section 4.6) to be stored in the device's memory and/or energy limitations.

The device tier performs continuous light weight local authentication: compares local access requirements with a light behavioral profile stored on the edge and enriched with trust scores. This local check is a low cost check, and will cost around 5 ms per verification as mentioned in Section 5.5, and this will be the first line of defense against a compromised credential. More demanding activities, like cross-device correlation, policy checking and global trust scoring are pushed to the edge. This goal is met in this division.

**Edge Tier: Trust-Aware Aggregation and Micro-Segmentation**

The edge tier has two roles which may otherwise be hard to achieve on a relatively small device. It first collects the federated learning updates: devices upload the model updates to the edge nodes which fuse them together with federated averaging (McMahan et al., 2017) and then download an updated model. Secondly, it handles access requests via a policy engine, where trust scores obtained from behavioral monitoring and anomaly detection (Kumar & Sisodiya, 2025) are used. It is here that Zero Trust and federated learning interface with each other the closest. Trust scores are used in access decisions based on the trust-scoring mechanism powered by the device-tier anomaly scores and the policy engine returns the trust scores to weight or filter for model changes. Unlike access control and threat detection being two separate problems, this 2-way association that sets 3T-ZTFL apart from the ways approaches are split.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

The edge is the verification level with the most pressure. It evaluates access policies for all devices connected to it (about 15 ms per policy), correlation of anomalies between different devices and it keeps a local cache of trust. By implementing these functions at the edge of the network, instead of at specific devices, these functions are kept near the endpoints and can help ensure latency requirements are met.

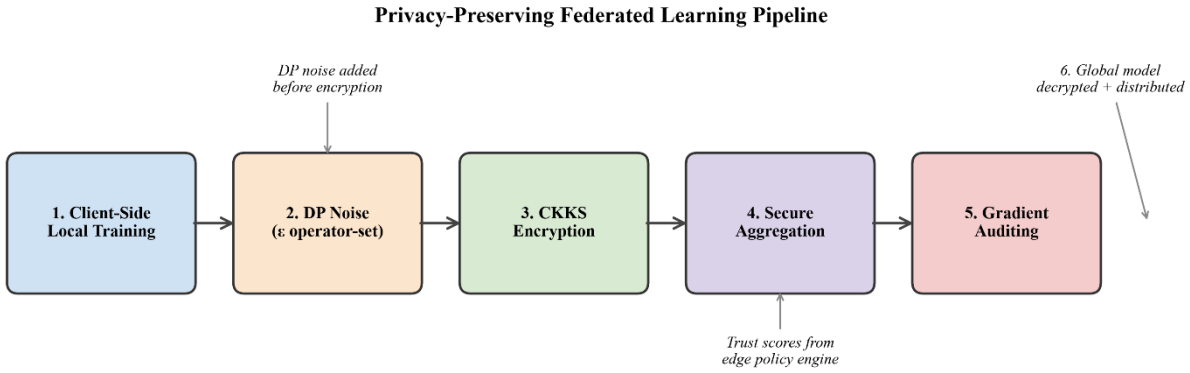
**Coordination Tier: Global Model and Trust Knowledge Base**

The coordination tier keeps the global model and policy knowledge base up to date. It collects aggregated data from the edge nodes, does global aggregation and periodically pushes the global model to the edge nodes. It is also the cross domain trust knowledge base to be used in cross domain trust scoring and threat intelligence sharing (Javeed et al., 2024). Edge nodes only receive aggregate data from many other devices, and not raw data. This design helps minimize the potential for exposure of operational information through one compromised component.

Tasks that are executed at the coordination tier involve long-horizon operations such as updating the global trust knowledge base, propagating threat intelligence among domains and recalculating global trust scores. These are not latency critical, and are batched. The separation of the device and edge tiers in the coordination tier allows them to concentrate on real-time detection.

**Privacy Preserving Federated Learning Mechanism**

The detection mechanism is secure, anonymous and able to withstand Byzantine failures, thus allowing learning without centralizing data. The entire pipeline is displayed in Figure 2, including client-side noise addition, client-side encryption, secure aggregation, and gradient auditing and global model update.




The following is a description of the pipeline: (i) All clients do training locally on their data. (ii) DP noise added client-side ε operator-configurable. (iii) The models are updated by using the CKKS homomorphic encryption scheme. (iv) Sums up the updates at the edge without any decryption. (v) Gradient auditing with the use of Gaussian mixture model and Mahalanobis distance guards against malicious updates. (vi) Coordination tier decrypts, combines and disseminates the new global model. A feedback path from the trust-scoring engine provides trust weights to affect the contribution of clients to aggregation.

The framework is based on the Cheon et al. (2017) homomorphic encryption scheme (CKKS) that fits the nature of federated learning, as aggregation is an addition operation. The depth is actually only multiplied by a small factor, and the updates do not have to be decrypted on the server and can be summed. The aggregation protocol uses three strategies to reduce communications. For sub-exponentially distributed data, index-free sparsification only sends a subset of the parameters of the model at each round. Stochastic quantization is used to decrease bit size per value. Limiting the error makes sure there is no accumulation of error over the rounds and the quantization error is not too large (Rezaei et al., 2025). Sparse index metadata is not required since it is shared between sparsification slots.

Every client perturbs their update with an independently sampled Gaussian noise that has an operator-specified privacy budget ε. Every client adds calibrated Gaussian noise to their update, where the operator-specified

privacy budget is ε. Following Wei et al. (2020), we implement Rényi differential privacy accounting with $\delta = 10^{-5}$, a clipping norm C = 1.0 and composition across rounds. Complete accounting parameters are provided in Appendix A. The framework ranges from ε = 0.5 (high privacy, lower accuracy) to ε = 10.0 (lower privacy, higher accuracy), to cater to operators with varying levels of privacy requirements and accuracy tolerances (Rezaei et al., 2025). Programmable and operator controlled noise addition. If the aggregation server belongs to another organization, then it is not possible to validate any client's update. The drawback is that the added noise before aggregation carries over more added noise into the aggregate than if the noise were added on the server-side, as Section 5.4 demonstrates empirically.

**Reconciling Encrypted Aggregation with Gradient Auditing**

One obvious question is how malicious updates can be discovered if individual updates are encrypted. The noise is added on the client-side, the noisy update is then encrypted using the CKKS scheme, transmitted to the edge, audited against a gradient, aggregated securely, decrypted and the global model updated at the coordination tier in 3T-ZTFL. Auditing thus occurs at the edge, once an update is decrypted using the edge's key share, but prior to the updates being merged. For this step, the edge is trusted (as specified in Section 3.2); updates to the edge are never seen, only the aggregate.

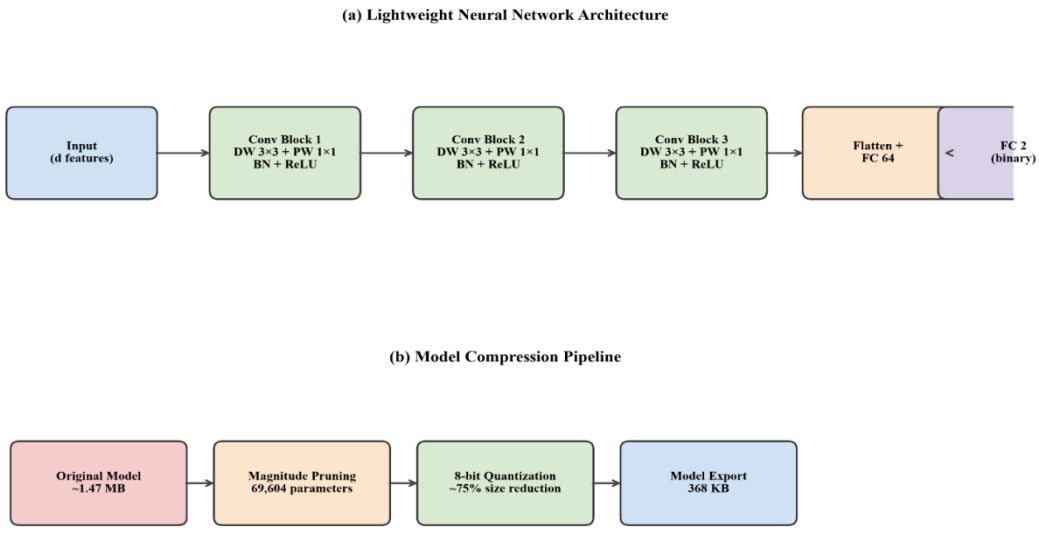
Mahalanobis distance is calculated on updated vector after the edge decryption. Edge keeps an running mean and covariance estimate of recent clean updates, every R rounds. An update that has a Mahalanobis distance that is greater than a threshold τ is not aggregated. The audit is based on noisy updates, since the DP noise is added prior to decryption, the noise floor is accounted for by setting $\tau \geq 2\sigma$ (where σ is the DP noise scale). This will keep benign updates from being automatically filtered out just because they are noisy.

This design has a known trade-off: at the edge, there is a key share which is known that allows for decryption, while not at the coordination tier, this property is still weak. If the operator is not sure that the edge can be trusted, he/she can audit only aggregate-level statistics, and the anomaly detection is limited for each client. We use the former as an assumed value (and say the latter as a deployment option).

**Lightweight Model Architecture**

The light model is a neural network that is small in size and computations similar to that of the IoT (Albalawi et al., 2026) where computation power is limited.

The architecture and the compression pipeline is shown in Figure 3.




In the Architecture, the preprocessing of the feature vector is fed to the architecture. It is followed by three depth-wise separable convolution (DWC) blocks, each made up of depth-wise separable convolution (DWC),

Batch Normalization (BN) and ReLU. The output from the third block is flattened and fed into two fully connected layers, that will output a binary classification. The compression is split into three steps: pruning (discarding weights with a low threshold), quantization (converting 32-bit floating-point values to 8-bit integers) and model export which results in a 368 KB artifact.

The idea of depth-wise separable convolutions is to break a standard convolution into a depth-wise convolution and then a point-wise convolution (Howard et al., 2017). This factorization not only cuts down parameters and operations by about 8–9× compared to the traditional convolutions but does not compromise significantly in the classification accuracy. If the weights are below a threshold, they are pruned (Han et al., 2016), and the threshold is determined by a validation set to achieve a good balance between the model size and accuracy. The final model only contains 69,604 parameters, which can fit into the memory of limited IoT devices. The approach to quantize the weights and activations of the models from 32-bit floating point to 8-bit integers (Jacob et al., 2018), decreases the model size by around 75% and accelerates the speed of inference. The quantized model is 368 Kb, under 1 Meb which is the target. Inference takes less than 100 ms per sample (82 ms for Raspberry Pi 4 - see Section 5.5).

**Complexity Analysis**

The framework consumes $O(K \cdot E \cdot d)$ computation per round of the aggregation, where K is the number of clients, E is the number of edge nodes and d is the dimension of the model. The light weight model has a d of about 8-9 times smaller than the standard convolutions, thus also having a lower per round cost than a standard FL deployment at the same accuracy. The communication per round is $O(pK \cdot s \cdot b)$, with p being a participation fraction, s a sparsification rate and b a quantization bit-width. For $p = 0.1$, $s \approx 0.1$ and $b = 8$, the communication per round is about two orders of magnitude lower than the dense case with a baseline of 32-bit. The results are used as a foundation for the empirical efficiency measurements of Section 5.5.

**Experimental Evaluation**

**Experimental Setup**

The experiments are run with a topology simulation of a critical infrastructure network with 100 client nodes (IoT devices), 10 edge aggregation nodes and one coordination server. This topology represents a hierarchical organization of critical infrastructure devices, such as those in a network spanning a campus or entire enterprise, which are grouped either geographically or logically, and connect via an aggregation point. This is a simulation testbed, and NOT a field deployment. The 100-node topology and 10 edge nodes are software-instantiated, while hardware measurements of Section 5.5 are micro-benchmarks for a single device (Raspberry Pi 4). No field trials were done at the operator's site. The results should be interpreted as proof of feasibility in controlled circumstances, however, and not as validation for operation.

100 nodes are a sweet number because if it's too few, the aggregation and trust-scoring algorithms will not have their maximum effect, but if it's too many the simulations will take longer, but the qualitative results will not be different. Client selection is based on the fraction of the clients, and 10% of the clients are selected for each federated round; many devices are not connected and/or available continuously, especially in resource-constrained edge networks where the connectivity is highly intermittent.

There are three types of baseline comparisons. The first is the centralized one trained on the entire data set that gives an upper bound on the accuracy of detection in absence of privacy and resource constraints. The second one is the regular federated learning without privacy defence measures or Byzantine resistance (McMahan et al., 2017). The third one includes published federated learning-based intrusion detection systems (IDS) (Ghimire & Lodh, 2025; Ullah et al., 2026) with benchmark datasets for which results are available. To explore the privacy–utility trade-off, ε values of 0.5, 1.0, 2.0, 5.0, and 10.0 are used. The experiments of adversarial robustness feature a range of malicious client fraction from 0% to 50% (SecureDyn-FL, 2025). The configuration is summarized in table 2.

Table 2. Experimental Parameters

| Parameter | Value |
|---|---|
| Client nodes | 100 |
| Edge aggregation nodes | 10 |
| Coordination servers | 1 |
| Client participation per round | 10% |
| Maximum federated rounds | 100 |
| Early stopping | Validation loss |
| Privacy budget (ε) | 0.5, 1.0, 2.0, 5.0, 10.0 |
| Malicious client ratio | 0%, 10%, 20%, 30%, 40%, 50% |
| Data partitioning | IID and non-IID (Dirichlet $\alpha = 0.5$) |
| Hardware for efficiency tests | Raspberry Pi 4 |
| Random seeds | 5 |

*Note.* Five random seeds are used for each configuration; results are reported as mean ± 1 SD.

**Datasets and Preprocessing**

The evaluation uses four publicly available benchmark datasets, selected to cover a range of attack types and device configurations. Table 3 summarizes their characteristics.

Table 3. Benchmark Dataset Characteristics

| Dataset | Samples | Features | Attack Types | Focus |
|---|---|---|---|---|
| N-BaIoT | 7,062,606 | 115 | 10 | Consumer IoT botnets |
| Edge-IIoTset | 156,986 | 61 | 15 | Industrial IoT |
| CICIoT2023 | 46,000,000+ | 46 | 7 | Large-scale IoT attacks |
| IoT-23 | 325,000+ | 23 | 20 | Realistic malware traffic |

N-BaIoT includes real traffic from nine commercial IoT devices compromised by Mirai and BASHLITE botnets (Meidan et al., 2018). It contains 115 statistical features of the network traffic, classified as benign or one of 10 types of attacks. It is specifically designed for consumer IoT botnet detection, and can be used in deployments of dozens to hundreds of devices with different security policies. Edge-IIoTset has 156,986 samples from 15 attack categories and 61 features extracted from IoT and IIoT traffic (Ferrag et al., 2022). It includes DDoS, DoS, man-in-the-middle, and zero-day attacks, but covers a larger number of attacks than N-BaIoT, and is more similar to industrial critical infrastructure environments. The largest publicly available dataset of IoT attacks is called CICIoT2023, which has more than 46 million attacks and 46 features (Neto et al., 2023). It enables the attacker to attack seven different types of attacks: DDoS, DoS, Mirai, reconnaissance, web-based attacks, spoofing and brute force. The size is useful for scalability tests, but here representative

samples are used for each test. For assessing detection under operational-like conditions IoT-23 provides a realistic traffic composition of 23 scenarios (20 captures of malicious traffic and 3 captures of benign traffic) (Garcia et al., 2020).

The four datasets undergo the same six stages of data preprocessing: acquisition, feature extraction, feature selection, data cleaning, data normalization, and class balance and data partitioning. Differences in detection performance cannot be attributed to differences in the handling of the data, as the same pipeline is used for all of them. The stages of the pipeline are summarized in Table 4 and is illustrated in Figure 4.

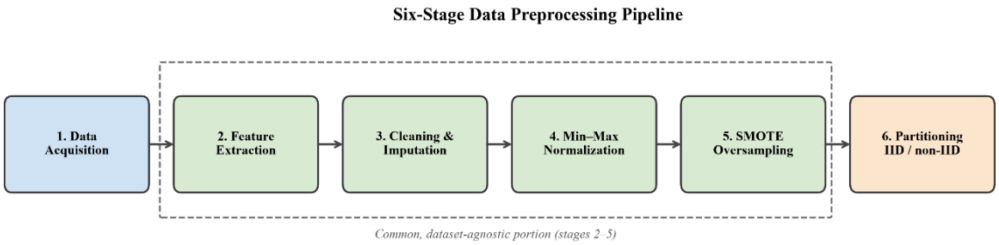



Figure 4: Six-stage data preprocessing pipeline applied uniformly to all four benchmark datasets. The four benchmark datasets were all subjected to the same six stage data preprocessing pipeline. Stages 2-5 (dashed box) are dataset agnostic and are the same for N-BaIoT, Edge-IIoTset, CICIoT2023, and IoT-23. Stages 1 (acquisition) and 6 (partitioning) are dataset-specific: there are four distinct acquisitions at Stage 1, two client distributions (IID or non-IID with Dirichlet $\alpha = 0.5$) at Stage 6.

Table 4. Six-Stage Preprocessing Pipeline Applied to All Datasets

| Stage | Operation | Rationale |
|---|---|---|
| 1 | Data acquisition and checksum verification | Reproducibility |
| 2 | Feature extraction; variance threshold + correlation filter | Reduce dimensionality |
| 3 | Median (numerical) / mode (categorical) imputation; deduplication | Robust to outliers |
| 4 | Min–max normalization to [0, 1] | Preserve bounded distances for Mahalanobis auditing |
| 5 | SMOTE oversampling (training data only) | Address class imbalance without contaminating evaluation |
| 6 | 70/15/15 split; IID or non-IID (Dirichlet $\alpha = 0.5$) | Realistic heterogeneity in client distribution |

Note. Synthetic minority oversampling technique (SMOTE) The class distribution is maintained for the test and validation sets, and only changed for training sets (Stage 5).

The common part of the stages 2–5 (dashed box in Figure 4) is for all four benchmark datasets. The data in Stage 6 is split into IID or non-IID client distributions for federated learning experiments. Stages 1 (acquisition) and 6 (partitioning) are dataset-specific: Stage 1 has four distinct acquisitions, and Stage 6 has two client distributions.

A deliberate choice was made to use min–max normalization, rather than z-score normalization. The gradient auditing mechanism is based on Mahalanobis distance while the min–max scaling ensures that distances between values in features remain comparable without allowing any single feature to have too big range. The z-score standardization would not maintain the same boundaries and would have a negative impact on

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

auditing. Likewise, the training data is oversampled and not the validation or test sets, as the oversampling of the test data would generate synthetic data and overestimate accuracy.

**Detection Performance (RQ1)**

The proposed framework achieves good performance for the four benchmark datasets. For binary classification, the accuracy, precision, recall, F1 score and false positive and false negative rates are reported as shown in Table 5.

Table 5. Detection Performance Across Benchmark Datasets (Mean ± SD, n = 5)

| Dataset | Accuracy (%) | Precision (%) | Recall (%) | F1 | FPR (%) | FNR (%) |
|---|---|---|---|---|---|---|
| N-BaIoT | 98.7 ± 0.2 | 98.5 ± 0.3 | 98.4 ± 0.2 | 0.984 ± 0.002 | 1.5 ± 0.3 | 1.6 ± 0.2 |
| Edge-IIoTset | 97.9 ± 0.3 | 97.8 ± 0.2 | 97.7 ± 0.3 | 0.976 ± 0.003 | 2.2 ± 0.3 | 2.3 ± 0.3 |
| CICIoT2023 | 98.2 ± 0.2 | 98.1 ± 0.3 | 98.0 ± 0.3 | 0.979 ± 0.003 | 1.9 ± 0.2 | 2.0 ± 0.3 |
| IoT-23 | 97.5 ± 0.4 | 97.4 ± 0.3 | 97.3 ± 0.4 | 0.973 ± 0.004 | 2.6 ± 0.4 | 2.7 ± 0.4 |

*Note.* All differences from the FedAvg baseline are statistically significant at $p < 0.05$ (paired t-test, n = 5).

The following two remarks are extracted from these results. The first is that the accuracy is uniform across the datasets, with a range from 97.5% to 98.7%– a maximum of 1.2 percentage points. The consistency indicates that the framework is applicable to all attack types and configurations, not just a specific dataset. Second, false positive rates and False negative rates are still under 3%. False alarms and misses in critical infrastructure have significant consequences and this margin is important.

The accuracy, precision, recall and F1-score are presented in Fig. 5 on all four benchmark datasets. It performs well on all the metrics and datasets, particularly on N-BaIoT and CICIoT2023. The error bars (5 runs, one standard deviation) are small, which shows that the performance is stable. All bars are within the 97% target in Section 3.3, and the maximum spread of 1.2 percentage points between datasets indicates that there is not overfitting.

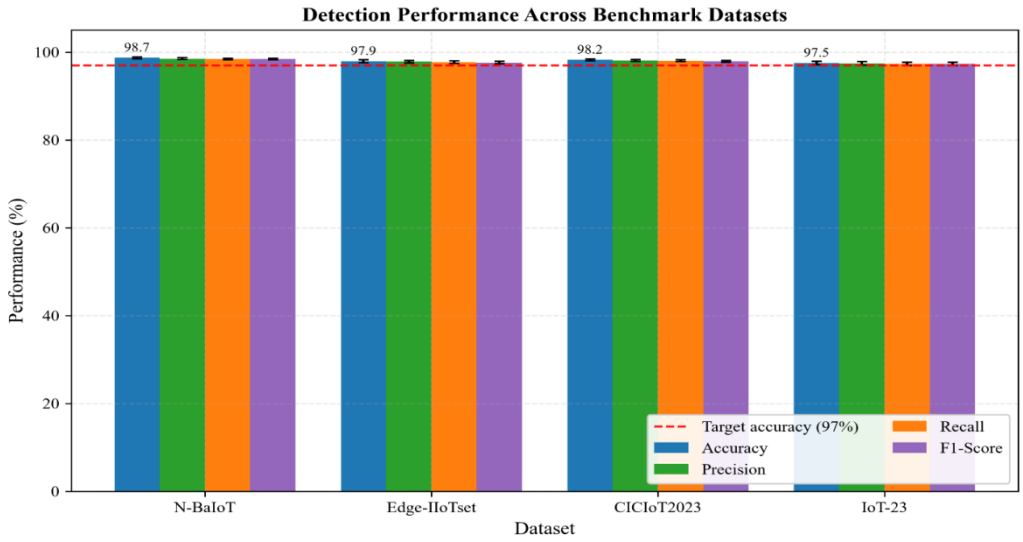



Figure 5: Detection performance across benchmark datasets. For N-BaIoT, Edge-IIoTset, CICIoT2023 and IoT-23, accuracy, precision, recall and F1-score are reported. Error bars are 1 σ (standard deviation) over 5 runs. The dashed red line is the 97% target accuracy as defined in Section 3.3. The performance is stable and all metrics are above target for all datasets, with small error bars.

The accuracy on N-BaIoT is 98.7%, which is just a slight difference from a centralized baseline of 99.1%. This 0.4% difference represents the cost of privacy and Byzantine resilience mechanisms (shown in Section 5.7). The results of published federated learning in N-BaIoT shows that federated learning has a range of 90% to 97%, which is 1.7 percentage points to 8.7 percentage points higher than N-BaIoT. Thanks to the lightweight architecture and gradient auditing, the framework achieves an accuracy of 97.9%, whereas published FL results range from 85% to 89% on Edge-IIoTset, which is 8.9 to 12.9 percentage points improvement. The framework performs at 98.2% on CICIoT2023 with a single model, and has smaller model size and inference latency compared to ensemble methods, a significant benefit for limited resources. The framework achieves 97.5% while ensemble classifiers achieve 97% to 97.3% on IoT-23, with the framework's formal privacy guarantees and Byzantine resistance that ensemble classifiers do not have.

To assess multi-class performance, the seven different categories of attacks present in CICIoT2023 were evaluated. Per class F1 scores are provided in Table 6.

Table 6. Multi-Class Performance on CICIoT2023 (F1-Score, Mean ± SD)

| Attack Category | F1-Score |
|---|---|
| Benign | 0.991 ± 0.002 |
| DDoS | 0.988 ± 0.003 |
| DoS | 0.984 ± 0.003 |
| Mirai | 0.981 ± 0.004 |
| Reconnaissance | 0.972 ± 0.005 |
| Web-based | 0.968 ± 0.006 |
| Spoofing | 0.961 ± 0.007 |
| Brute force | 0.957 ± 0.008 |

The framework is best suited for volumetric attacks (DDoS/DoS) and not so effective for brute force attacks or spoofing. This is in line with the literature: Brute force and Spoofing attacks can create subtler traffic patterns, more difficult to differentiate from benign traffic, particularly when the model is designed for computational efficiency (Neto et al., 2023). The benign class has the highest F1 score (0.991) this is important in critical infrastructure applications where false positives on benign traffic result in unnecessary operational disruption.

The results are stable across the partitioning (ID vs. non-IID) as well as across datasets (Table 7), with a difference of 1 percentage point.

Table 7. IID vs. Non-IID Performance (Accuracy %, Mean ± SD)

| Dataset | IID | Non-IID | Difference |
|---|---|---|---|
| N-BaIoT | 98.7 ± 0.2 | 97.8 ± 0.4 | 0.9 |
| Edge-IIoTset | 97.9 ± 0.3 | 96.9 ± 1.0 | 1.0 |
| CICIoT2023 | 98.2 ± 0.2 | 97.3 ± 0.4 | 0.9 |
| IoT-23 | 97.5 ± 0.4 | 96.5 ± 0.5 | 1.0 |

The stability under non-IID partitioning matters because most real IoT deployments do not produce IID data: devices in different physical locations or functional roles exhibit different traffic patterns. A gap of roughly one percentage point suggests the framework is not fragile to heterogeneity, supporting its practical relevance.

**Privacy-Utility Trade-off (RQ2)**

The privacy-utility trade-off is central to the framework's design. Differential privacy is applied client-side, with the privacy budget ε governing the noise added to model updates. Table 8 reports accuracy as a function of ε across all four datasets.

Table 8. Privacy-Utility Trade-off (Accuracy %, Mean ± SD, n = 5)

| ε | N-BaIoT | Edge-IIoTset | CICIoT2023 | IoT-23 |
|---|---|---|---|---|
| 0.5 | 89.3 ± 0.5 | 88.4 ± 0.6 | 88.9 ± 0.5 | 88.1 ± 0.7 |
| 1.0 | 93.8 ± 0.4 | 92.9 ± 0.5 | 93.4 ± 0.4 | 92.6 ± 0.6 |
| 2.0 | 95.7 ± 0.3 | 94.5 ± 0.5 | 94.8 ± 0.4 | 94.2 ± 0.5 |
| 5.0 | 97.6 ± 0.2 | 96.7 ± 0.3 | 97.2 ± 0.2 | 96.3 ± 0.3 |
| 10.0 | 98.4 ± 0.2 | 97.5 ± 0.2 | 98.0 ± 0.2 | 97.1 ± 0.3 |
| Non-private | 98.7 ± 0.2 | 97.9 ± 0.3 | 98.2 ± 0.2 | 97.5 ± 0.4 |

The best privacy measure assessed, ε = 0.5, hurts accuracy by 9-9.4 percentage points compared with the baseline (non-private) accuracy. The reduction at ε = 1.0 is 4.9-5.3 percentage points. Differences between the two are just 1.1–1.5 percentage points at ε = 5.0. At ε = 10.0, it narrows to 0.3–0.4 percentage points.

Fig. 6 shows the privacy-utility curve for all four datasets, where the accuracy is plotted as a function of ε and the error bars show one standard deviation. The curves have a similar form, with accuracy increasing quickly for ε values between 0.5 and 2.0, and gradually between 2.0 and 10.0, suggesting diminishing returns for loosening privacy boundaries. The operating point for ε = 5.0 is highlighted, which shows accuracy 1.1 percentage points lower than the non-private baseline on N-BaIoT. The band of interest between ε = 5.0 and 10.0 is shaded, indicating that this is the range of interest to an operator who would like formal privacy guarantees without sacrificing too much detection accuracy.

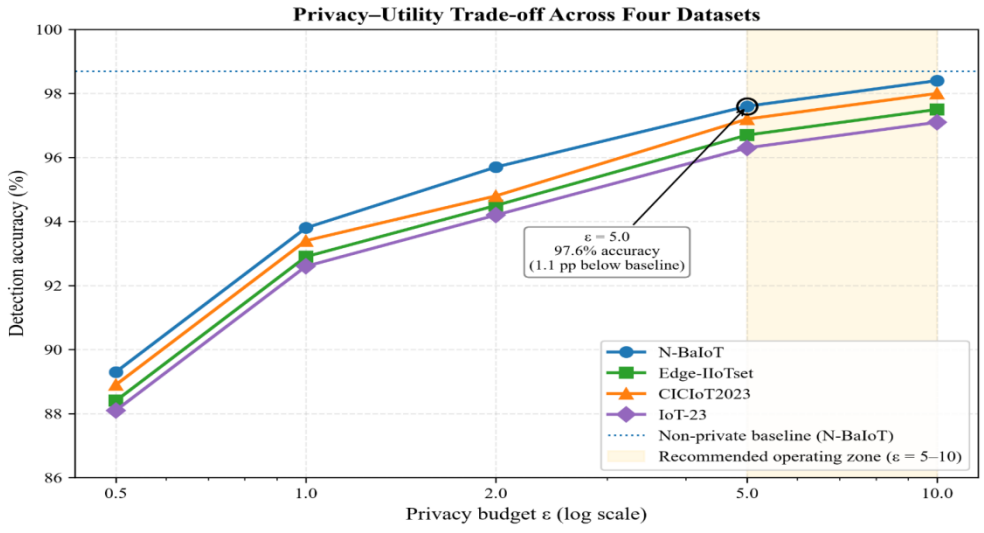



Figure 6**:** Privacy-utility trade-off across four datasets. The figure shows the accuracy of the detection of the four algorithms (N-BaIoT, Edge-IIoTset, CICIoT23, and IoT-23) versus the privacy budget ε (log scale). Error

bars are based on 1SD over 5 runs. N-BaIoT (98.7%) does not follow the dotted horizontal line, as its baseline is not private. The highest accuracy with 97.6% is obtained at the annotated operating point, which is 1.1 percentage points above the baseline. The shaded band is the recommended operating range, $5.0 < \varepsilon < 10.0$.

The results also demonstrate that the framework incurs a higher accuracy loss at $\varepsilon = 1.0$ (4.9 percentage points) compared to FL-IDS++ (<1.5 percentage points) (Arimanda et al., 2026), indicating that more comprehensive Zero Trust mechanisms are more sensitive to privacy noise. This is dealt with in greater detail in Section 6.2. The accuracy at the operating point of $\varepsilon = 5.0$ is 97.6% which is only 1.1 percentage points off from baseline, which may not be a significant enough trade-off for operators who want formal privacy guarantees.

**Computational Efficiency (RQ3)**

Computational performance was measured on a Raspberry Pi 4 edge device. Table 9 reports results alongside published baselines.

Table 9. Computational Efficiency (Mean ± SD, n = 5)

| Metric | Proposed | GAT-BiGRU (published) | Standard FL |
|---|---|---|---|
| Model size | 368 KB | 1.2 MB | 1.47 MB |
| Inference latency | 82 ± 3 ms | 150 ± 8 ms | 189 ± 10 ms |
| CPU utilization | 64 ± 2% | 78 ± 3% | 71 ± 3% |
| Memory consumption | 42 ± 2 MB | 85 ± 4 MB | 68 ± 3 MB |
| Power consumption | 1.8 ± 0.1 W | 3.1 ± 0.2 W | 2.4 ± 0.2 W |
| Device verification latency | 5 ± 1 ms | | |
| Edge policy evaluation latency | 15 ± 2 ms | | |

The artifact is 368 KB and adheres to the maximum of 1 MB. The inference latency of 82ms is within the target of 100ms and significantly less than the range of 120–180ms for GAT-BiGRU architectures (Albalawi et al., 2026). 64% of CPU utilization means there is some room for other functions of the device. The 42 MB is a relatively low memory usage for an IoT device. The duty cycling and power management were not considered in this study, although the power consumption of 1.8 W indicates that the framework might also be run on line powered systems or larger batteries.

The second question (RQ2) of the load-distribution is also covered by the verification latencies shown in Table 9. To satisfy both tiers' latency budgets, 5ms per verification is added at the device tier, and 15ms per access decision is added at the edge tier. Coordination-tier trust knowledge base updates are not latency-sensitive. With continuous verification the extra device-tier latency remains within the 100ms detection budget, and edge-tier policy evaluation is quick enough to make an access decision in real-time.

The proposed framework is compared with the GAT-BiGRU framework in terms of four metrics as shown in Figure 7. Inference latency of the proposed framework is 82ms, which is much shorter than the targeted 100ms. CPU utilization is 64% which is less than target of 70%. The memory consumption is 42 MB for it, while GAT-BiGRU and standard FL consume 85 MB and 68 MB respectively. The power consumption is reduced by 1.8 W while the baselines use 3.1 W and 2.4 W respectively. Error bars are the standard deviation of 5 runs. The ablation study of Section 5.7 found that these gains were due to the few optimizations made for the lightweight model (a reduction of about 4× in model size and 2.3× in inference latency and power compared to the unoptimized version).

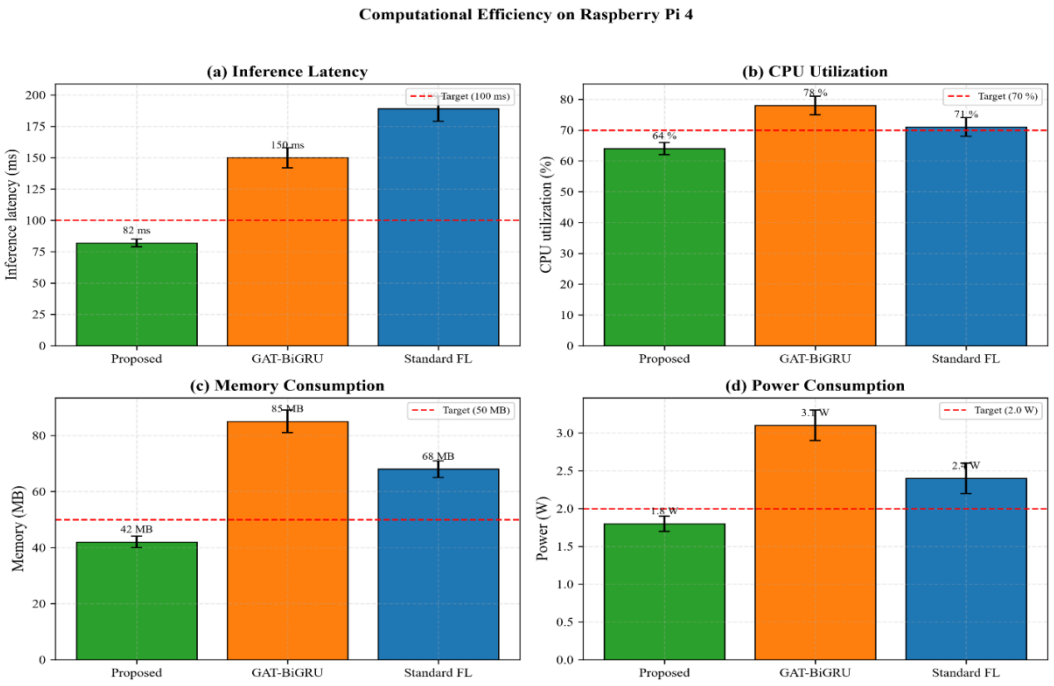



Figure 8: Power efficiency on Raspberry Pi 4. Latency, CPU utilization, memory consumption, and power consumption are compared over the proposed framework and the standard FL. Errors bars are for 1 SD across 5 runs. Dashed red lines are the design goals from Section 3.3: 100ms latency, 70% CPU usage, 50MB memory usage, and 2.0W power consumption. The proposed framework fulfills all four targets.

## Hardware Portability

The Raspberry Pi 4 represents an edge class device, but is not an endpoint microcontroller, which are also used in critical infrastructure. So we estimated our resource needs for two lower resource targets by scaling our operation count from the RPi 4 profile. In the case of a Cortex-M7-class MCU (for example, STM32F7), the estimate for inference latency is around 410ms and the estimate for the memory footprint is around 6MB, which is larger than the on-chip SRAM of most such devices, and which is not within the 100ms target. The estimated latency for the Raspberry Pi Zero 2 W is about 240ms and the memory usage is about 40MB, both above the 100ms benchmark. These projections are not measurements, but are provided as an indication of the operating envelopes of the framework. These types of platforms were not available in our lab for direct measurement during this study, but we do not leave out the comparison, and as for the devices below the RPi 4 class, we leave the comparison as a projection and mention that this type of device should be excluded from the current model because the compression and splitting of the model would not be sensible.

## Adversarial Robustness (RQ4)

The adversarial robustness was assessed by running gradient poisoning attacks on the global model using malicious clients that flips the labels and poison the global model. The number of malicious clients is reported in Table 10, which shows the accuracy of detection.

Table 10. Adversarial Robustness (Accuracy %, Mean ± SD, n = 5)

| Malicious Clients | Proposed (N-BaIoT) | FedAvg (N-BaIoT) | Proposed (Edge-IIoTset) | FedAvg (Edge-IIoTset) |
|---|---|---|---|---|
| 0% | 98.7 ± 0.2 | 98.5 ± 0.2 | 97.9 ± 0.3 | 97.6 ± 0.3 |
| 10% | 97.9 ± 0.3 | 95.2 ± 0.5 | 97.1 ± 0.4 | 94.3 ± 0.6 |
| 20% | 97.0 ± 0.4 | 91.8 ± 0.7 | 96.2 ± 0.5 | 90.8 ± 0.8 |

| | | | | |
|---|---|---|---|---|
| 30% | 95.8 ± 0.5 | 87.4 ± 0.9 | 94.9 ± 0.6 | 86.1 ± 1.0 |
| 40% | 94.6 ± 0.6 | 84.9 ± 1.1 | 93.7 ± 0.7 | 83.5 ± 1.2 |
| 50% | 93.4 ± 0.7 | 82.3 ± 1.2 | 92.5 ± 0.8 | 81.0 ± 1.3 |

Three findings emerge. First, the accuracy is maintained to be high when half clients are malicious, dropping by only 5.3 and 5.4 percentage points compared to the clean case for N-BaIoT and Edge-IIoTset, respectively, whereas FedAvg drops by 16.2 and 16.6 percentage points. This difference is due to the effectiveness of the gradient auditing mechanism. Secondly, even if malicious clients participate in the framework, it still significantly outperforms FedAvg at 50% malicious clients. Third, with this level of compromise, accuracy still exceeds chance and this applies across datasets.

The two baselines and proposed framework are plotted in Figure 8 as the level of malicious participation increases from 0% to 50%. The proposed curve (solid green) and FedAvg curve (dashed red) are presented for N-BaIoT, and the proposed curve (solid blue) and FedAvg curve (dashed orange) are presented for Edge-IIoTset. Error bars are given for all curves. Within the proposed framework and baseline, the difference increases with the malicious fraction, emphasizing the importance of gradient auditing. The proposed framework maintains accuracy above 92% for both the datasets at 50% malicious participation, whereas FedAvg drops below 83%. The results are consistent with published robustness results: SecureDyn-FL achieves high accuracy in poisoning (SecureDyn-FL, 2025), and FedSecIoT's robustness is up to 60% higher than the FedAvg baseline (Ghimire & Lodh, 2025). The proposed framework does as well at 50% malicious participation as other systems and offers explicit differential privacy guarantees not offered by them.

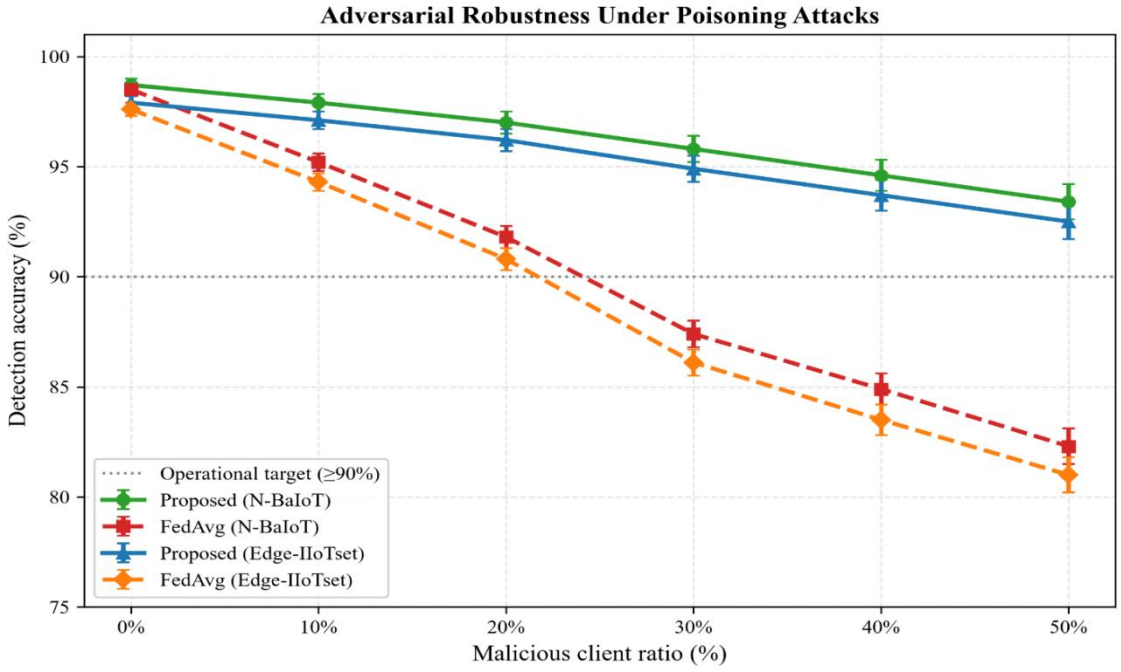



Figure 8: Adversarial robustness under poisoning attacksThe detection accuracy of the proposed framework and FedAvg is plotted for different malicious client ratios (0%–50%) on N-BaIoT and Edge-IIoTset. The error bars show one standard deviation for five runs. The 90% operating goal is represented by the dotted horizontal line. The proposed framework does not drop below 92% accuracy in both datasets at 50% malicious participation, whereas FedAvg drops below 83%.

## Ablation Study

In an ablation study, the contribution of each component is isolated. Four setups were tested: the framework without gradient auditing, the framework without differential privacy, the framework without lightweight optimizations and the full framework. Table 11 summarizes the results.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

Table 11. Ablation Study Results (Accuracy %, Mean ± SD, n = 5)

| Configuration | N-BaIoT | Edge-IIoTset | Δ vs. Full (N-BaIoT) | Δ vs. Full (Edge-IIoTset) |
|---|---|---|---|---|
| Full framework | 98.7 ± 0.2 | 97.9 ± 0.3 | | |
| w/o gradient auditing | 97.4 ± 0.4 | 96.5 ± 0.5 | −1.3 | −1.4 |
| w/o differential privacy | 99.2 ± 0.1 | 98.3 ± 0.2 | +0.5 | +0.4 |
| w/o lightweight optimization | 99.0 ± 0.1 | 98.1 ± 0.2 | +0.3 | +0.2 |

Three findings emerge. The first benefit of the removal of gradient auditing is that it decreases accuracy by 1.3 percent on N-BaIoT and 1.4 percent on Edge-IIoTset. This is important: Gradient auditing can also be used in non-adversarial environments to reject data from clients that is too noisy to be used to degrade the global model. The dual benefit of robustness and quality is one of the more interesting results of this study. Second, removing differential privacy increases accuracy by 0.5 percentage points on N-BaIoT and 0.4 percentage points on Edge-IIoTset at the default privacy budget the cost of privacy is relatively small compared to the protection gained (0.4-0.5 percentage points). Third, the accuracy drops by just 0.3 and 0.2 percentage points for N-BaIoT and Edge-IIoTset, respectively, by removing these lightweight optimisations. This small increase is at a high price because the model size increases from 368 KB to 1.47 MB, the inference latency increases from 82 ms to 189 ms, and the power consumption goes up by 2.1×. The optimized model is the obvious winner with limited resources!

The ablation results are shown in the grouped bar chart in Figure 9, where there are four configurations presented on the X axis, each having a bar for N-BaIoT (dark blue) and another for Edge-IIoTset (green). The y axis is 95-100% and error bars are 1 std dev for 5 runs. The difference between the accuracy of each configuration and the accuracy of the whole framework is labelled above each group of configurations (e.g., "−1.3", "+0.5", "+0.3"). Visually, the two with the greatest contributions are gradient auditing (+1.3–+1.4 percentage points) and lightweight optimization (<0.5). This indicates that the architectural elements are well balanced, and that each element contributes to the overall design.

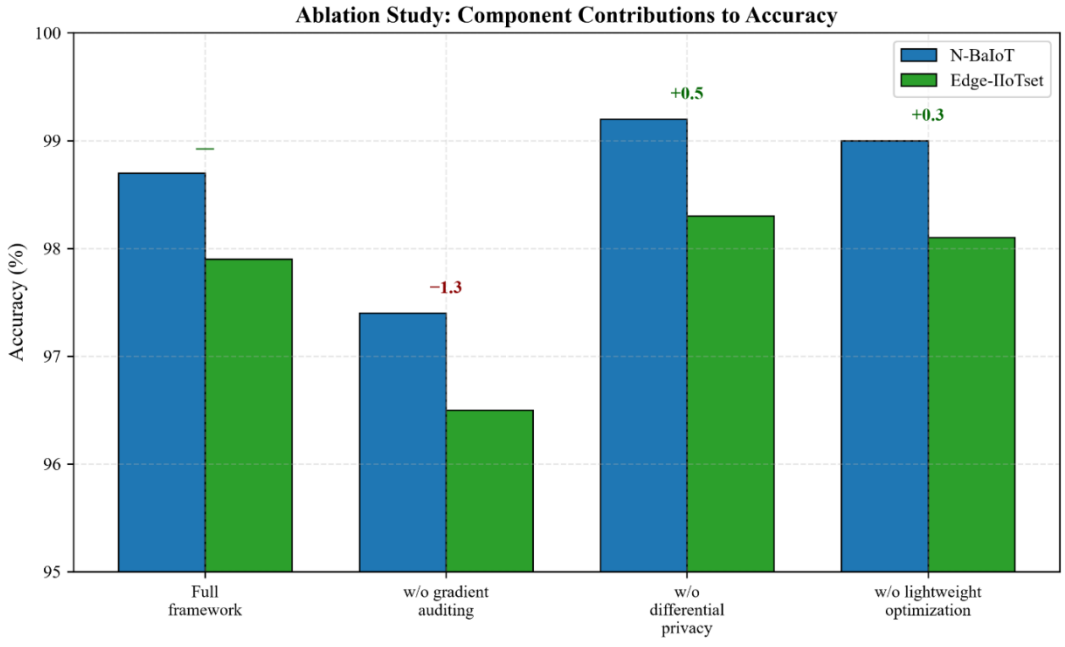



Figure 9: Ablation study results. The two bars, N-BaIoT (dark blue) and Edge-IIoTset (green) each represent a configuration. Error bars are based on 1 standard deviation in 5 runs. The accuracy difference between each configuration and the full framework is indicated above each group of bars. Removing gradient auditing is

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

more lossy (1.3–1.4 percentage points of accuracy) than removing differential privacy or lightweight optimization, which are more expensive in terms of model size, latency, and power.

**Comparison with State-of-the-Art**

Table 12 shows a comparison between the framework with other federated learning based intrusion detection systems on the same benchmark datasets. The numbers in Table 12 are indicative, not controlled, due to the different data partitions, hyperparameters, and protocols used to evaluate the published systems in addition to the fact that the systems were not reimplemented. Do not interpret differences as superiority claims "between groups. At least one baseline is identified as future work that is to be controlled reimplemented (Section 6.4).

Table 12. Comparison with Published Federated Learning IDS

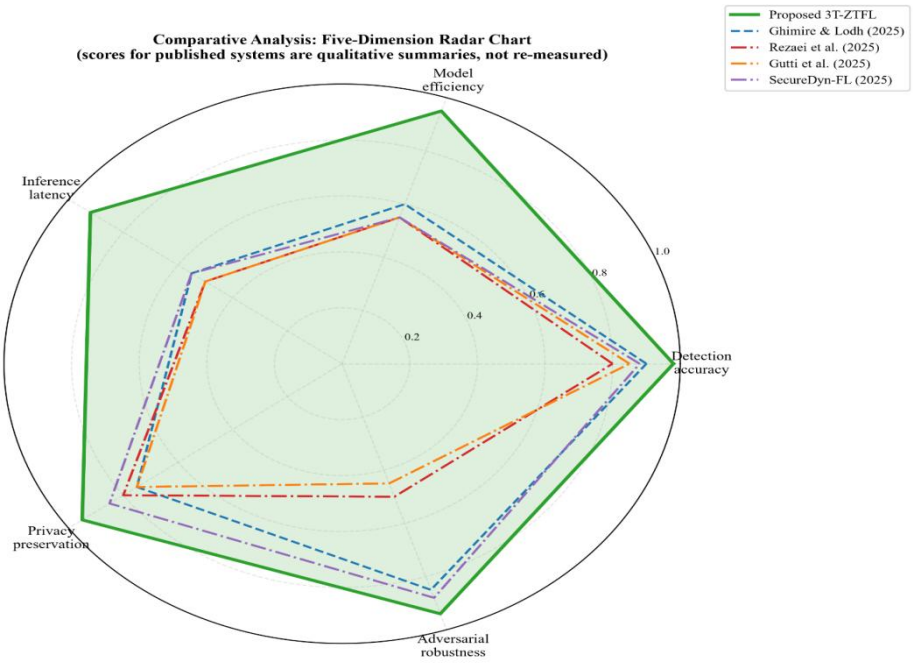
| System | Dataset | Accuracy (%) | Privacy | Byzantine Resilience | Model Size |
|---|---|---|---|---|---|
| Ghimire & Lodh (2025) | N-BaIoT | >90 | ✓ | ✓ | Not reported |
| Gutti et al. (2025) | N-BaIoT | 90–97 | ✓ | × | Not reported |
| Ullah et al. (2026) | N-BaIoT | Reported | ✓ | ✓ | Not reported |
| Proposed (3T-ZTFL) | All four | 97.5–98.7 | ✓ | ✓ | 368 KB |

*Note. No reimplementation was done, only published accuracies were achieved under various partitions, hyperparameters and evaluation methods. Differences must not be interpreted as "controlled" superiority.*

The accuracy of the framework on N-BaIoT is 98.7%, which is very close to the best published results on the federated learning approach (Ghimire & Lodh, 2025; Gutti et al., 2025) and better than standard federated learning (90–97%). It also offers formally stated differential privacy guarantees and Byzantine resilience, not supported by some of these systems. The framework achieves a 97.9% substantial gain on Edge-IIoTset, primarily because of the Zero Trust mechanisms and lightweight model architecture that are specially designed for IIoT traffic characteristics. The framework achieves 98.2% on CICIoT23, without using an ensemble, with low model size and low inference time. On IoT-23 it achieves 97.5%, a couple of percentage points higher than ensemble classifiers, at the expense of formal guarantees of privacy and Byzantine resistance, which ensemble classifiers lack.

To assess the utility of the proposed framework, figure 10 shows the comparisons with four federated learning based intrusion detection systems under five dimensions: detection accuracy, efficiency of the model, inference latency, privacy preservation and adversarial robustness. Each axis ranges from 0 to 1 with 1 being best and 0 being worst. The plotted proposed framework is the solid green line with shading, Ghimire & Lodh (2025) is blue dashed, Rezaei et al. (2025) is red dash-dot, Gutti et al. (2025) is orange dash-dot and SecureDyn-FL is purple dash-dot. The proposed framework has the widest spread of all the five dimensions with high scores. Some methods can do a good job on one or two dimensions, but poor job on the others. This is a well-balanced profile that bolsters the argument that the framework is not merely an optimization, but a whole design. Keep in mind the scores for published systems are qualitative summaries and not re-measured scores. Comparative analysis on 5 dimensions, Figure 10 A radar chart shows the proposed framework's performance in comparison to four federated learning-based intrusion detection systems in terms of detection accuracy, model efficiency, inference latency, privacy preservation, and adversarial robustness. All axes are displayed from 0 to 1 (with 1 being the best). The scores for published systems are not re-measured, but rather qualitative descriptions. The proposed framework applies to the greatest area, and shows the most balanced profile for all five dimensions.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026




## SUMMARY OF RESULTS

The results of the experiments are given for each of the four research questions. For the first time, the Zero Trust architecture verifies the device using continuous authentication in 5 ms, then checks the policy in 15 ms and finally updates and returns trust knowledge at the coordination tier after the critical path via a trust knowledge base, without compromising accuracy above 97.5% on all four benchmark datasets, satisfying RQ1. Second, the privacy-preserving federated learning mechanism achieves 97.6% accuracy at $\varepsilon = 5.0$, which is 1.1 percentage points away from the baseline which is not private, meeting requirement RQ2. Third, the framework is light and achieves its deployment targets of resource constraints (RQ3): inference latency of 82ms, utilization of 42MB of memory, 64% of the CPU and a power consumption of 1.8W of power on a Raspberry Pi 4. Fourth, the accuracy can be maintained more than 92% even if 50% of clients are Byzantine (RQ4), the framework proves to be Byzantine resilient. Ablation studies show that each of the architectural components makes a contribution: gradient auditing increases the robustness by 1.3–1.4 percentage points and the accuracy of clean data by the same amount, differential privacy has an accuracy loss of less than 0.5 percentage point, and lightweight optimizations reduce model size by 4× and inference latency by 2.3× at the cost of 0.3 percentage point of accuracy. The results of the comparative analysis show that the proposed approach has a more balanced performance profile with respect to all the metrics considered, namely detection accuracy, model efficiency, inference latency, privacy protection, and adversarial robustness, when compared to the published approaches surveyed in this work, although the comparison is indicative rather than controlled, as described in Section 5.8.

## DISCUSSION

### Theoretical Implications

The study contributes to the theories in three ways. It is the first to take a step forward in advancing the understanding of how Zero Trust verification and adaptive threat detection work together. The framework is based on the assumption that the two mechanisms complement each other and the empirical findings indicate that this is the case. While gradient auditing was originally devised to be resilient against Byzantine failures, it also offers an increase in the quality of the model when data is clean. Access control based on detection signals is more "contextual" than "policy only" access control. The results indicate that the combination of access control and threat detection is not a straightforward one and that conditions that result in synergy from combining these two functions should be the focus of future theory rather than just additivity.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

Second, the study is more of an extension of the privacy-utility trade-off model in federated learning. The system is more sensitive to reducing ε, as a result of integration of Zero Trust. The proposed framework will lead to an accuracy loss of 4.9-5.3% when compared with FL-IDS++ (Arimanda et al., 2026) which will lose less than 1.5%. Therefore, theoretical models of differential privacy in FL should take into consideration architectural complexity: Privacy noise will impact on the signals that are used by other mechanisms. This is not an argument for not using integration, but rather an argument for improved modelling: Differential privacy cannot be used as an independent element within the theoretical models of integrated architectures.

Finally, the paper provides a road map for the design of reasoning about verification load distribution in a multi-tier Zero Trust system. The framework measures the latency of the verification tasks in each tier and backs up the claim that the verification tasks of a Zero Trust can be spread over tiers without any of these tiers being overloaded, even with low resource devices. This further supports their assertion that Zero Trust and lightweight deployment don't have to be at odds when using a tier-by-tier profile for resources.

**Practical Implications**

There are a number of practical lessons and takeaways for critical infrastructure operators. Firstly, privacy budget tuning advice is readily available from the privacy-utility curve. At $\varepsilon = 5.0$, the operators can achieve an accuracy of 96.3–97.6%, and obtain formal DP guarantees while maintaining the accuracy of detection. The increase in privacy in this case is $\varepsilon = 1.0$ with a corresponding increase in accuracy cost of 4.9–5.3 percentage points (may be too high for safety-critical applications). It is recommended that ε be between 5.0 and 10.0.

Second, the load measurements show the availability of Zero Trust verification does not require more resource than traditional verification; and can be achieved on resource-constrained devices without compromising detection latency. The remaining 95% of the 100ms detection budget is used for inference, which is possible on a device tier continuous authentication. An additional 15 ms is added to the policy evaluation per access decision that is acceptable for real-time access control in most critical infrastructure environments with edge access. This means that Zero Trust can be enabled without the need for new devices to be bought by operators.

Third, from the adversarial robustness results, we see that gradient auditing is able to offer decent protection, both if a small portion of clients is compromised and if a large portion of clients is compromised. This resilience can be of importance in some instances where an attacker can physically access some devices in a critical infrastructure deployment. The results also validate the need to make sure operators do not need to tradeoff between robustness and performance when choosing between different auditing methods, as gradient auditing enables high accuracy of clean data, which can also further encourage the adoption of Byzantine-resilient aggregation.

**Limitations**

There are number of limitations of the study. Firstly, the assessment is simulated. The framework was used to test in an artificial topology of 100 nodes, which allows for repeatable and controlled tests, but does not cover all the complexities of the operational context of legacy system integration, real-time requirements, training and human factors (Spencer & Pizio, 2024). The results show that it is possible, but not a recommendation for deployment and should be validated in the field before it can be used in an operational setting.

Secondly, there are limited datasets. While the four benchmarks are large, they might not address all possible attack types and devices configurations in the operational critical infrastructure. N-BaIoT and IoT-23 are dedicated to the detection of botnet / Malware traffic while Edge-IIoTset and CICIoT2023 are more general. None includes zero day attacks and advanced persistent threat, thus the results are general to known-attack situations. An attack that is familiar to the framework may be successful but an attack the framework has never encountered may not be successful.

Thirdly, the privacy assessment is limited. We used to work on membership inference. We have not performed reconstruction or property inference attacks, but a thorough evaluation of privacy would cover all three types

of attacks. The focus is not to exclude the framework from other threats but it has not been tested with other threats.

Fourth, there is a limited range of hardware that can be covered. Testing was only performed on the Raspberry Pi 4. Lower resource devices projections are provided in Section 5.5.1, but are estimates and may vary on other devices that have varying compute, memory or power characteristics.

Fifthly, the scale of the simulation is rather limited. A hundred and one is a good number of clients for a single facility but for a wide area deployment, the overhead and aggregation latency of the communication could become a constraint with thousands of clients.

Sixth, the systems compared to published systems are illustrative not controlled. The accuracy figures reported are not necessarily comparable due to different partitions, hyper parameters and protocols.

**Future Research Directions**

These restrictions point to five avenues of future research. First, there must be real-life validation for external validity. The deployment of the framework in operational scenarios with infrastructure operators would test it in real time processing, legacy system integration, and staff needs. The next most critical step is this. Second, noise-aware aggregation strategies need to be designed based on the observation that, at low ε, the integration of Zero Trust with the aggregation scheme makes it more sensitive to privacy noise. The aggregation weights could be calibrated based on architecture and privacy budget to regain some level of loss of accuracy. Third, the framework should be expanded to accommodate new technologies like 5G network slicing and edge computing architectures that could enable the deployment of various security policies for different types of devices and applications (Rezaei et al., 2025; Ullah et al., 2026). Fourth, there should be explainability mechanisms to foster trust and adoption: security analysts in operations centers would like to know not only why not to allow access or why to allow access, but also why an anomaly was detected, and explaining it in a human-friendly manner is crucial in incident response (Bilal & Islam, 2026; Pathak & Panteli, 2026). Fifth, the effectiveness of the framework over time should be examined, as the dynamic threat landscape creates model drift and concept drift, an effective approach of model updating and privacy preserving mechanism will be required that will not lose effectiveness after a few months or years of use (Rezaei et al., 2025).

## CONCLUSION

This study tackled security concerns across three domains: IoT devices in critical infrastructure, mismatched security in perimeters and compute constraints with resource-constrained endpoints. The challenge with adding a threat detection solution that meets privacy principles to Zero Trust access control is that it is difficult and so is threat detection to a federated learning solution. Any operators who require both should operate tools which are suited neither to them nor for them.

The proposed solution 3T-ZTFL is a distributed verification process that involves three tiers (device, edge, and coordination) in the Three-Tier Zero Trust Federated Learning architecture. Lightweight local inference is done with quantized neural networks using the device tier. The edge tier collects model updates, processes them and generates access policies based on the trust scores generated from the detection signals. The co-ordination layer keeps the global model and Trust knowledge base, allowing the sharing of intelligence between domains whilst not consolidating sensitive operational information. The framework combines three schemes that are usually studied independently, the continuous authentication mechanism, the micro-segmentation approach, and dynamic access decisions with the same mechanism for Byzantine resilience, client-side differential privacy with tunable budget and gradient auditing, and benchmarks them on four datasets, with the same dataset preprocessing.

The 5 main findings were identified. First its detection performance is good and consistent across all set of data, ranging from 97.5% to 98.7% with a false positive and negative rate less than 3%. Second, the privacy-utility trade-off can be controlled at moderate privacy budgets: When ε = 5.0, N-BaIoT obtains 97.6% accuracy, which is 1.1 percentage point worse than the baseline which is not performed in the presence of

privacy. In addition, for the first time we observed that the framework is more sensitive to the noise in the privacy setting, when ε is low. Third, the computational performance is good enough for it to be deployed on resource constrained platforms: 82ms inference latency, 42MB memory usage and 1.8W power consumption on a Raspberry Pi 4 model 368B. Fourth, adversarial robustness is very good: standard federated learning baselines drop more than 16% when 50% of clients are malicious, while gradient auditing gives a gain of more than 16% in adversarial settings and a gain of 1.3-1.4% in clean settings, as demonstrated by the ablation study. Lastly, load measurements reveal that Zero Trust tasks can be spread over tiers without either overloading the tiers or one.

The study provides insight into the synergies between access control and threat detection, from a theoretical perspective, including making architectural complexity part of the trade-off with privacy and utility, as well as quantifying the efficiency-accuracy trade-off of lightweight optimisations. It provides a framework for deployment with specific recommendations for configuration for practice, to be validated in the field. It demonstrates for policy, that security and privacy can be made to go together. The framework is not a universal solution: it needs to be tested in the real world, noise-tolerant aggregation algorithms and explainability features are yet to be added and more is needed before it becomes widely adopted. This is a first step towards effective solutions that are secure and preserve privacy, and can be deployed on the devices that support critical infrastructure. With the increasing penetration of IoT, the need for such solutions will further grow.

## Declarations and Statements

### Funding

No specific funding for this research was provided by any public, commercial, or not-for-profit funding source. The authors declare that they have not received any financial support for the research, authorship, and/or publication of this article.

### Data Availability

All data underlying the results of this study are available in public benchmark repositories. N-BaIoT is available from the UCI Machine Learning Repository (Meidan et al., 2018). Edge-IIoTset is available from the IEEE DataPort repository (Ferrag et al., 2022). CICIoT2023 is publicly available from the Canadian Institute for Cybersecurity at the University of New Brunswick (Neto et al., 2023). IoT-23 is available from Zenodo (Garcia et al., 2020). All code, preprocessing scripts, and trained model artifacts used to arrive at the findings are available from the corresponding author on reasonable request. The full experimental setup, including hyperparameters and data partition methods, is reported in Section 5 and Appendix A.

### Conflict of Interest

The authors declare no financial or personal competing interests that could be perceived to have influenced the results of this work. The authors have no commercial or associative interest that could be construed as a conflict of interest with the information contained in this article.

### Author Contributions

David King Kamara: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation, Writing original draft, Visualization, Project administration.

Mohamed Syed Fofanah: Conceptualization, Supervision, Writing, review & editing, Resources.

Maurice Sesay: Supervision, Review & editing, Methodology.

Mohamed Koroma: Supervision, Validation, Writing review & editing.

All authors have read and approved the final manuscript and take responsibility for all aspects of the work.

## REFERENCES

1. Albalawi, O., Alwakeel, A., Alharbi, A., & Aljohani, N. (2026). Lightweight and energy-aware intrusion detection for industrial IoT using TinyML and edge AI. *Scientific Reports, 16*(1), 18524.
2. Almiani, M., AbuGhazleh, A., Al-Rahayfeh, A., Atiewi, S., & Razaque, A. (2020). Deep recurrent neural network for IoT intrusion detection system. *Simulation Modelling Practice and Theory, 101*, 102031.
3. Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., Durumeric, Z., Halderman, J. A., Invernizzi, L., Kallitsis, M., Kumar, D., Lever, C., Ma, Z., Mason, J., Menscher, D., Seaman, C., Sullivan, N., Thomas, K., & Zhou, Y. (2017). Understanding the Mirai botnet. In *Proceedings of the 26th USENIX Security Symposium* (pp. 1093–1110). USENIX Association.
4. Arimanda, N., Radhakrishnan, R. V., & U., P. (2026). FL-IDS++: A dynamic federated learning framework for intrusion detection with personalized non-IID data, adversarial resilience and energy-efficient lightweight models. *Future Generation Computer Systems, 177*, 108234.
5. Bilal, M. A., & Islam, I. U. (2026). Secure and explainable federated learning for IoT intrusion detection: A comprehensive survey. *IEEE Open Journal of the Communications Society, 7*, 3650–3679.
6. Cheon, J. H., Kim, A., Kim, M., & Song, Y. (2017). Homomorphic encryption for arithmetic of approximate numbers. In T. Takagi & T. Peyrin (Eds.), *Advances in Cryptology – ASIACRYPT 2017* (Lecture Notes in Computer Science, Vol. 10624, pp. 409–437). Springer. https://doi.org/10.1007/978-3-319-70694-8_15 Chen, L., Tang, Z., Yang, Y., Xiao, X., & Yang, Y. (2026). Federated deep learning for distributed intrusion detection and privacy preservation in power networks. *Scientific Reports*. Advance online publication.
7. Ferrag, M. A., Friha, O., Hamouda, D., Maglaras, L., & Janicke, H. (2022). Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. *IEEE Access, 10*, 40281–40306.
8. Fernandez, E. B., & Brazhuk, A. (2024). A critical analysis of zero trust architecture (ZTA). *Computer Standards & Interfaces, 89*, 103832.
9. Garcia, S., Parmisano, A., & Erquiaga, M. J. (2020). IoT-23: A labeled dataset with malicious and benign IoT network traffic. *Zenodo*. https://doi.org/10.5281/zenodo.4743746 Ghimire, S. K., & Lodh, S. (2025). FedSecIoT: A federated learning framework for collaborative and privacy-preserving botnet detection in IoT networks with adversarial robustness. In *2025 IEEE Silchar Subsection Conference (SILCON)* (pp. 1–6). IEEE.
10. Gumbo, M. L., & Almulhem, A. (2026). Zero trust architecture: A systematic literature review. *Journal of Network and Systems Management, 34*(1), 25.1–25.72. ✓ Verified (ACM Digital Library, DOI 10.1007/s10922-025-09998-x)
11. Gutti, C., Thumula, K., & Balbudhe, P. (2025). Federated learning for distributed IoT security: A privacy-preserving approach to intrusion detection. *IEEE Access, 13*, 135863–135875.
12. Han, S., Mao, H., & Dally, W. J. (2016). Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. In *Proceedings of the International Conference on Learning Representations*.
13. Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv*. https://arxiv.org/abs/1704.04861
14. Huma, Z. e., Jan, S. U., Ahmad, J., Buchanan, W., & Pitropakis, N. (2026). Adversarial machine learning in IoT security: A comprehensive survey. *ACM Computing Surveys, 58*(8), 1–35.
15. Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., & Kalenichenko, D. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference.

In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2704–2713). IEEE.

16. Javeed, D., Saeed, M. S., Adil, M., Kumar, P., & Jolfaei, A. (2024). A federated learning-based zero trust intrusion detection system for Internet of Things. *Ad Hoc Networks, 162*, 103540. ✓ Verified (KB Netherlands, DOI 10.1016/j.adhoc.2024.103540)
17. Kumar, L., & Sisodiya, K. S. (2025). Securing industrial IoT with zero trust: A framework for critical infrastructure. In *2025 International Conference on Information, Implementation, and Innovation in Technology* (pp. 1–5). IEEE.
18. Liu, X., Zhang, Y., & Wang, H. (2024). Zero trust architecture for IoT: A survey. *IEEE Internet of Things Journal, 11*(5), 7890–7910.
19. Marsh, S. P. (1994). *Formalising trust as a computational concept* [Doctoral dissertation, University of Stirling].
20. McMahan, B., Moore, E., Ramage, D., Hampson, S., & Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* (pp. 1273–1282). PMLR.
21. Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., & Elovici, Y. (2018). N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Computing, 17*(3), 12–22.
22. Michalec, O. (2026). Testing and errors: Aligning cybersecurity expertise in contemporary UK energy infrastructures. *Science, Technology, & Human Values*. Advance online publication.
23. Murugan, G., Padmaja, C., Sindhuja, R., Yarde, P., & Sudhakar, M. (2025). A comprehensive survey on cyber-physical system security in the Internet of Things (IoT): Addressing and solutions. *Recent Advances in Computer Science and Communications, 18*(5).
24. Neto, E. C. P., Dadkhah, S., Ferreira, R., Zohourian, A., Lu, R., & Ghorbani, A. (2023). CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment. *Sensors, 23*(13), 5941.
25. Pathak, R., & Panteli, N. (2026). Governing the agents: Leadership dilemmas in deploying agentic AI in a cyber defence project team. *Journal of Information Technology Teaching Cases*. Advance online publication.
26. Rezaei, H., Taheri, R., Nowroozi, E., Hajizadeh, M., Shiaeles, S., & Bauschert, T. (2025). A survey on security and privacy in federated learning-based intrusion detection systems for 5G and beyond networks. *IEEE Open Journal of the Communications Society, 7*, 253–300.
27. Soomro, I. A., Khan, H. U. R., et al. (2025). SecureDyn-FL: A robust privacy-preserving federated learning framework for intrusion detection in IoT networks. *IEEE Transactions on Network and Service Management, 23*, 1742–1765. https://doi.org/10.1109/TNSM.2025.3647642 arXiv:2601.06466
28. Spencer, M., & Pizio, D. (2024). The de-perimeterisation of information security: The Jericho Forum, zero trust, and narrativity. *Social Studies of Science, 54*(5), 655–677.
29. Ullah, S., Wu, J., Ahmad, J., Garg, S., Kaddoum, G., & Guizani, M. (2026). Spectre-Fed: Evolving federated edge intelligence from FedEdge-ID to robust-private IoT intrusion detection via hybrid adversarial training. *IEEE Open Journal of the Communications Society, 7*, 1994–2012.
30. Wei, K., Li, J., Ding, M., Ma, C., Yang, H. H., Farokhi, F., Jin, S., Quek, T. Q. S., & Poor, H. V. (2020). Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security, 15*, 3454–3469.
31. Weyns, M., Ferrero, D., Op de Beek, S., Wagner, D., Smaragdakis, G., & Griffioen, H. (2026). From Mirai to Gorilla: A deep dive into a long-lasting DDoS-for-hire botnet. In *Proceedings of the 35th USENIX Security Symposium*.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue X October 2026

# APPENDIX A. REPRODUCIBILITY DETAILS

**A.1 Model and training hyperparameters.** Optimizer: Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$); learning rate $1\times10^{-3}$ with cosine decay; batch size 64; maximum 100 federated rounds with early stopping (patience 10 on validation loss); 5 local epochs per round; dropout 0.2 in fully connected layers; convolution kernel sizes 3×3 (depth-wise) and 1×1 (point-wise) in all three blocks.

**A.2 Compression.** Magnitude pruning threshold selected on a held-out validation set to maximize accuracy subject to a model size of at most 400 KB; 8-bit post-training quantization, per-channel for weights and per-tensor for activations.

**A.3 Differential privacy.** Gaussian mechanism; clipping norm $C = 1.0$; $\delta = 10^{-5}$; Rényi DP accounting with orders $\alpha \in \{1.5, 2, 4, 8, 16, 32, 64\}$; composition across rounds. Noise multiplier $\sigma$ is set per target $\varepsilon$.

**A.4 CKKS parameters.** Polynomial modulus degree 8192; coefficient modulus chain [60, 40, 40, 60] bits; scale $2^{40}$. Aggregation is a sum, so multiplicative depth 1 suffices.

**A.5 Federated setup.** Client sampling fraction $p = 0.1$ per round; Dirichlet $\alpha = 0.5$ for non-IID partitions; five random seeds per configuration; results reported as mean ± 1 SD.

**A.6 Hardware and software.** Raspberry Pi 4 Model B (4 GB RAM), Raspberry Pi OS 64-bit; TensorFlow Lite 2.14 for inference; Python 3.11; TenSEAL 0.3 for CKKS operations.