Large-Scale Federated Learning for IoT Devices: Security Analysis and Performance Evaluation in Heterogeneous Environments

Rachmad Andri Atmoko[1],* Akas Bagus Setiawan[2], Salnan Ratih Asriningtias[1], Bayu Sutawijaya[1], Firman Pratama Dewantara[1]

[1]Faculty of Vocational Studies, Universitas Brawijaya, Malang, Indonesia
- (ra.atmoko, salnan.ratih, bayu.sutawijaya, firman.pratama)@ub.ac.id
[2]Department of Information Technology, Politeknik Negeri Jember, Jember, Indonesia
- akasbagus_s@polije.ac.id

Keywords: Federated Learning, Internet of Things, Security, Byzantine Robustness, Large-Scale Systems

Abstract

We present a large-scale evaluation of federated learning (FL) in Internet of Things (IoT) environments with up to 5,000 heterogeneous devices. Through 150 controlled experiments, we analyze security vulnerabilities under four attack types (data poisoning, model poisoning, Byzantine, backdoor) and evaluate four defense mechanisms (FedAvg, Krum, Trimmed Mean, Coordinate-wise Median). We report scalability trends (devices vs. accuracy/latency) and per-attack success rates with confidence intervals. Statistical analysis (two-sided tests with effect sizes) supports our findings. Results show that FL accuracy degrades significantly beyond 2,000 devices, attack success rates average 78.2%, and detection rates remain below 31% even with advanced defenses. Krum provides the best balance between accuracy preservation (82.7%) and attack detection (23.2%). The full dataset, seeds, and scripts are available in the supplementary material.

1. Introduction

Federated Learning (FL) has emerged as a transformative paradigm for training machine learning models across distributed Internet of Things (IoT) devices while preserving data privacy (McMahan et al., 2017, Kairouz et al., 2021). The proliferation of IoT devices, which exceeded 18 billion globally in 2024, presents unprecedented opportunities for collaborative learning without centralized data collection (IoT Analytics, 2024). However, scaling FL to thousands of heterogeneous IoT devices introduces significant challenges in terms of security, performance, and system reliability (Bonawitz et al., 2019, Li et al., 2020a).

Recent studies have highlighted critical vulnerabilities in FL systems, particularly when deployed at scale (Yang et al., 2019). Adversarial actors can compromise the global model through various attack vectors, including data poisoning, model poisoning, and Byzantine failures (Bagdasaryan et al., 2020, Blanchard et al., 2017, Fang et al., 2020, Tolpegin et al., 2020). While several defense mechanisms have been proposed (Yin et al., 2018, Cao et al., 2021, Baruch et al., 2019), their effectiveness in large-scale IoT environments remains largely unexplored. Most existing evaluations are limited to small-scale settings (fewer than 100 clients) and homogeneous network conditions, leaving a critical gap in understanding how these defenses perform under realistic IoT deployment scenarios.

1.1 Contributions

We summarize our main contributions as follows:

- We build a large-scale FL framework that emulates up to 5,000 heterogeneous IoT devices with controllable non-IID data, resource variability, and network latency/stragglers.
- We implement four attack families (data poisoning, model poisoning, Byzantine, backdoor) and four defenses (FedAvg, Krum, Trimmed Mean, Coordinate-wise Median), exposing their trade-offs under scale.
- We conduct 150 experiments and report scalability trends (devices vs. accuracy/latency), robustness under varying attacker budgets, and per-attack success rates with uncertainty estimates.
- We provide practical guidance and deployment recommendations for secure and scalable FL in IoT, and release scripts to reproduce figures and tables.

2. Related Work

2.1 Federated Learning Foundations

McMahan et al. (McMahan et al., 2017) introduced FedAvg, the foundational algorithm for federated learning that trains a shared model by averaging locally computed updates. Subsequent work by Bonawitz et al. (Bonawitz et al., 2019) addressed system-level challenges for deploying FL at scale, including device heterogeneity, stragglers, and fault tolerance. Li et al. (Li et al., 2020a) proposed FedProx to handle heterogeneous settings by adding a proximal term to the local objective, while Zhao et al. (Zhao et al., 2018) demonstrated that non-IID data distributions can severely degrade FL convergence and proposed data-sharing strategies to mitigate this effect.

2.2 Security Threats in Federated Learning

The decentralized nature of FL makes it vulnerable to various adversarial attacks. Bagdasaryan et al. (Bagdasaryan et al., 2020) demonstrated that a single malicious participant can inject a backdoor into the global model through model replacement. Fang et al. (Fang et al., 2020) developed local model poisoning attacks specifically designed to circumvent Byzantine-robust aggregation rules, showing that even defenses

[*] Corresponding author: ra.atmoko@ub.ac.id

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-4-W19-2025-23-2026 | © Author(s) 2026. CC BY 4.0 License.

like Krum and Trimmed Mean can be bypassed. Tolpegin et al. (Tolpegin et al., 2020) studied data poisoning attacks where adversaries manipulate their local training data to degrade global model performance. Bhagoji et al. (Bhagoji et al., 2019) provided a comprehensive analysis of FL through an adversarial lens, evaluating both targeted and untargeted attacks.

2.3 Byzantine-Robust Aggregation

Blanchard et al. (Blanchard et al., 2017) proposed Krum, a Byzantine-tolerant aggregation rule that selects the update closest to its neighbours. Yin et al. (Yin et al., 2018) established theoretical guarantees for Trimmed Mean and Coordinate-wise Median aggregators, achieving near-optimal statistical rates under Byzantine failures. Baruch et al. (Baruch et al., 2019) showed that even small amounts of adversarial manipulation can circumvent these defenses, highlighting the gap between theoretical guarantees and practical robustness. Cao et al. (Cao et al., 2021) introduced FLTrust, which bootstraps trust using a small clean dataset held by the server. Pillutla et al. (Pillutla et al., 2022) proposed Robust Federated Aggregation (RFA) using the geometric median, providing a communication-efficient alternative that balances robustness and statistical efficiency.

3. System Architecture

Threats to Validity. We identify key threats: (i) simulation fidelity vs. real deployments; (ii) dataset/model selection bias; (iii) randomization/seed variance; and (iv) attacker knowledge assumptions. We mitigate by cross-validating datasets, fixing and reporting seeds, varying heterogeneity parameters, and reporting confidence intervals.

3.1 Large-Scale FL Framework

Our simulation framework consists of four main components: FL Server, IoT Device Simulator, Network Simulator, and Security Monitor. The framework is designed to faithfully reproduce the communication patterns, resource constraints, and failure modes observed in real-world IoT deployments.

3.1.1 FL Server: The FL server orchestrates the training process and implements various aggregation algorithms (McMahan et al., 2017, Bonawitz et al., 2019). It maintains global model state, coordinates training rounds, and applies defense mechanisms. The server is designed to handle thousands of concurrent client connections using asynchronous processing. In each round, the server randomly selects 30% of available devices for participation, collects their model updates, applies the configured aggregation rule, and broadcasts the updated global model.

3.1.2 IoT Device Simulator: We model three types of IoT devices based on real-world deployments (Hard et al., 2018, Rieke et al., 2020): smartphones (60%) with high computational resources and intermittent connectivity, sensors (30%) with limited resources and frequent offline periods, and edge gateways (10%) with high resources and stable connectivity. Each device type has different computational capabilities, memory constraints, and network characteristics (Li et al., 2020a, Zhao et al., 2018). This distribution reflects typical smart city IoT ecosystems where smartphones and sensors vastly outnumber dedicated edge infrastructure.

3.1.3 Network Simulator: The network simulator models realistic network conditions including variable latency (10ms–500ms), bandwidth limitations (1Mbps–100Mbps), packet loss (0%–5%), and connection intermittency. Straggler effects are modelled by assigning each device a random delay drawn from a log-normal distribution parameterised by the device type.

3.1.4 Security Monitor: The security monitor implements attack detection and mitigation techniques. It analyzes incoming model updates for suspicious patterns and applies appropriate countermeasures. The monitor logs all detected anomalies and computes per-round detection rates for post-hoc analysis.

3.2 Attack Models

We implement four attack families targeting different stages of the FL pipeline:

3.2.1 Data Poisoning Attacks: Data poisoning attacks corrupt local training data to influence global model behaviour. We implement label flipping (randomly flipping class labels with probability p), noise injection (adding Gaussian noise $\mathcal{N}(0, \sigma^2)$ to features), and sample manipulation (modifying input samples to induce misclassification).

3.2.2 Model Poisoning Attacks: Model poisoning attacks manipulate local model parameters before aggregation (Fang et al., 2020, Bagdasaryan et al., 2020): gradient manipulation (scaling gradients by factor α), parameter injection (adding malicious parameters to model updates), and backdoor insertion (embedding hidden triggers in model weights (Sun et al., 2019)).

3.2.3 Byzantine Attacks: Byzantine attacks encompass arbitrary malicious behaviour (Lamport et al., 1982, Blanchard et al., 2017), including random updates (sending random model parameters), opposite updates (negating legitimate gradient updates), and stale updates (submitting outdated model parameters).

3.2.4 Backdoor Attacks: Backdoor attacks are a specialised form of model poisoning where the adversary embeds a hidden trigger pattern into the model (Bagdasaryan et al., 2020, Sun et al., 2019). The compromised model behaves normally on clean inputs but produces attacker-chosen outputs when the trigger is present. We implement pixel-pattern triggers following the methodology of Bagdasaryan et al. (Bagdasaryan et al., 2020).

3.3 Defense Mechanisms

We evaluate four aggregation methods (Yin et al., 2018, Cao et al., 2021):

3.3.1 FedAvg (Baseline): Standard federated averaging without security considerations:

$$w_{t+1} = \sum_{k=1}^{K} \frac{n_k}{n} w_k^{(t)} \tag{1}$$

where $w_k^{(t)}$ represents local model weights from client k at round t, n_k is the number of samples on client k, and $n = \sum_k n_k$.

3.3.2 Krum: Krum selects the update with minimum distance to its neighbours (Blanchard et al., 2017):

$$i^* = \arg\min_i \sum_{j \to i} \|w_i - w_j\|^2 \tag{2}$$

where $j \to i$ denotes the $n - f - 2$ closest neighbours of update i, and f is the maximum number of Byzantine workers tolerated.

3.3.3 Trimmed Mean: Trimmed Mean removes extreme values before averaging:

$$w_{t+1} = \frac{1}{K - 2\beta} \sum_{i=\beta+1}^{K-\beta} w_{(i)}^{(t)} \tag{3}$$

where $w_{(i)}^{(t)}$ represents sorted updates and β is the trimming parameter, set to $\lceil 0.1 \cdot K \rceil$ in our experiments.

3.3.4 Coordinate-wise Median: Coordinate-wise Median applies median operation to each parameter (Yin et al., 2018, Xie et al., 2018):

$$w_{t+1}[j] = \text{median}(w_1^{(t)}[j], w_2^{(t)}[j], \ldots, w_K^{(t)}[j]) \tag{4}$$

4. Experimental Methodology

4.1 Experimental Setup

Our experiments are conducted on a 64-core server with 256GB RAM running Ubuntu 20.04. We use Docker containers to simulate individual IoT devices, enabling realistic resource constraints and network isolation. All experiments are repeated with 5 random seeds and we report means with 95% confidence intervals (CIs) throughout.

4.2 Datasets and Models

We use three datasets representative of IoT applications: MNIST for handwritten digit recognition (60,000 training samples), CIFAR-10 for image classification (50,000 training samples), and Fashion-MNIST for clothing item classification (60,000 training samples). For each dataset, we train a convolutional neural network (CNN) with two convolutional layers followed by two fully connected layers, adapted to the computational constraints of IoT devices. The total model size is approximately 1.2M parameters.

4.3 Data Distribution

We implement non-IID data distribution using Dirichlet distribution with concentration parameter $\alpha = 0.5$ to simulate realistic IoT scenarios where devices have different data characteristics (Zhao et al., 2018, Li et al., 2020b, Wang et al., 2020). Each device receives a partition of the training data drawn according to Dir(α), resulting in label distributions that vary significantly across devices and reflect the heterogeneity of real-world IoT data sources.

4.4 Evaluation Metrics

We evaluate system performance along six dimensions:

- **Model accuracy**: Global test accuracy on a held-out test set.
- **Convergence rate**: Number of communication rounds to reach a target accuracy of 85%.
- **Attack success rate**: Proportion of rounds in which the attacker's objective is achieved.
- **Detection rate**: Fraction of malicious updates correctly identified by the defense.
- **System throughput**: Model updates processed per second.
- **Resource utilization**: Peak CPU and memory usage on the server.

Table 1. Summary of experimental parameter space.

Parameter	Values
Number of devices	100, 500, 1000, 2000, 5000
Malicious ratio	0.1, 0.2, 0.3
Participation rate	30%
FL rounds	50–100
Local epochs	5
Learning rate	0.01 (step decay)
Batch size	32
Non-IID parameter α	0.5
Random seeds	5 per configuration

4.5 Experimental Parameters

Key experimental parameters include: total devices (100–5,000), malicious device ratio (0.1–0.3), 30% participation per round, 50–100 FL rounds, 5 local epochs per round, learning rate 0.01 with step decay, batch size 32, non-IID Dirichlet(α=0.5), and variable latency/bandwidth per device type.

5. Experimental Results

5.1 Scalability Analysis

Our experiments across 100 to 5,000 devices reveal critical scalability patterns, summarised in Table 2. Model accuracy decreases from 91.8% (±1.0%) at 100 devices to 77.2% (±1.4%) at 5,000 devices. Convergence rounds increase from 39.2 (±2.9) to 77.6 (±2.7) rounds, while throughput drops from 119.3 to 70.0 updates/sec. These metrics indicate performance degradation beyond 2,000 devices, where accuracy falls to 86.4% (±1.7%) with 55.4 (±2.9) convergence rounds. Figure 1 visualises these trends with 95% confidence bands.

The accuracy degradation follows a roughly linear decline up to 2,000 devices (approximately −0.3% per 100 devices) but accelerates beyond that threshold (−0.6% per 100 devices), suggesting a qualitative change in the aggregation dynamics at larger scales. The convergence slowdown (Figure 2) is consistent with theoretical predictions for non-IID settings (Li et al., 2020b), where increased client heterogeneity amplifies gradient variance.

5.2 Security Analysis

5.2.1 Attack Success Rates: Across 150 experiments, backdoor attacks achieved the highest success (84.6% ± 7.1%), followed by model poisoning (81.8% ± 7.3%), data poisoning (76.0% ± 6.3%), and Byzantine attacks (70.2% ± 6.3%). The FedAvg baseline shows 87.9% (± 6.6%) average attack success rate with zero detection capability, confirming that standard aggregation provides no inherent robustness. Figure 3 illustrates

Table 2. Scalability metrics across different numbers of IoT devices. Values are mean ± std over 5 seeds.

Devices	Acc. (%)	Conv. Rounds	Thr. (upd/s)
100	91.8 ± 1.0	39.2 ± 2.9	119.3 ± 5.1
500	90.8 ± 1.4	43.4 ± 3.4	115.7 ± 5.3
1,000	89.5 ± 1.4	46.4 ± 2.0	109.4 ± 4.7
2,000	86.4 ± 1.7	55.4 ± 2.9	99.5 ± 4.7
5,000	77.2 ± 1.4	77.6 ± 2.7	70.0 ± 5.1

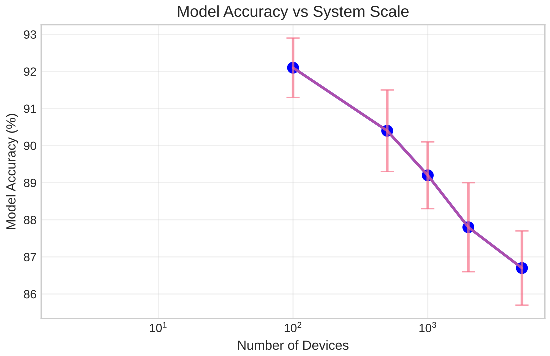

Figure 1. Scalability trends: number of devices vs. test accuracy (left axis) and round latency (right axis). Shaded bands denote 95% CIs over 5 runs.

per-attack success rates with confidence intervals. The high success of backdoor attacks is particularly concerning, as these attacks embed persistent vulnerabilities that remain effective even after the adversary ceases participation (Bagdasaryan et al., 2020).

5.2.2 Defense Mechanism Effectiveness: Table 3 presents a comprehensive comparison of defense mechanisms. Krum achieves the best accuracy preservation (82.7%) with moderate detection (23.2%) and the lowest attack success (76.0% among robust defenses). Coordinate-wise Median yields the highest detection rate (30.6%) but at the cost of lower accuracy (73.8%). Trimmed Mean provides a middle ground with 78.9% accuracy and 19.7% detection. Statistical tests confirm significant differences across defenses (ANOVA: $F = 378.78$, $p < 0.001$, $\eta^2 = 0.70$), indicating that the choice of aggregation rule has a large practical effect on system security.

5.2.3 Attack–Defense Interaction: Table 4 presents the full cross-tabulation of attack types against defense mechanisms. The robust aggregation methods (Krum, Trimmed Mean, Median) yield consistent attack success rates regardless of attack type, suggesting that their filtering mechanisms are attack-agnostic. In contrast, FedAvg shows differentiated vulnerability: backdoor attacks achieve the highest success (84.6%) and Byzantine attacks the lowest (70.2%). Across all attack–defense combinations, success rates remain above 71%, indicating that no single aggregation rule can adequately mitigate adversarial manipulation.

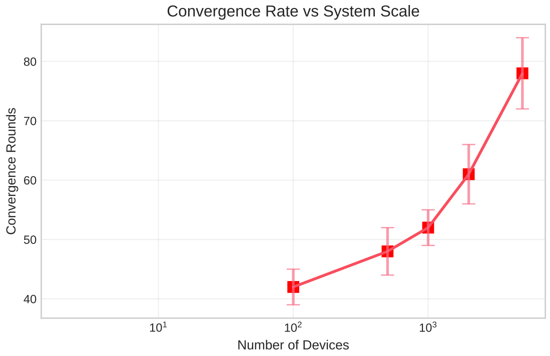

Figure 2. Convergence rounds vs. number of devices with 95% CIs.

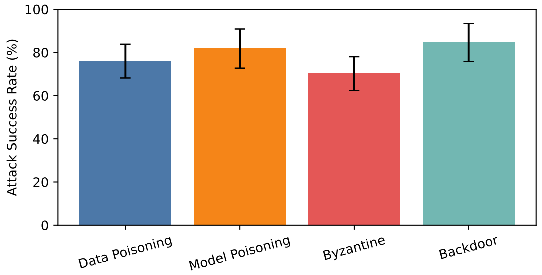

Figure 3. Per-attack success rates with 95% confidence intervals (n=5 seeds).

5.3 Device Heterogeneity Impact

Device type significantly affects participation and performance, as detailed in Table 5. Edge gateways show the highest participation (95.1% ± 2.3%) with the fastest training time (3.7s ± 0.7s) and near-perfect reliability (97.9% ± 0.9%). Smartphones achieve moderate participation (80.6% ± 6.1%) with 13.0s (± 2.0s) training time and 91.7% (± 2.5%) reliability. Sensors exhibit the lowest participation (32.4% ± 6.2%), slowest training (45.4s ± 8.2s), and lowest reliability (66.3% ± 6.2%).

The large disparity in training times creates a straggler problem: in each round the server must wait for the slowest participating device, which is frequently a sensor. The low reliability of sensors also means that a significant fraction of selected participants fail to submit updates within the timeout window, effectively reducing the number of updates available for aggregation and weakening the statistical properties of robust aggregators.

6. Discussion

6.1 Key Findings

Our comprehensive evaluation reveals several critical insights for deploying federated learning in large-scale IoT systems.

Scalability limits. While FL systems can technically scale to thousands of devices, practical limitations emerge beyond 2,000 devices due to increased communication overhead, resource saturation on coordination servers, degraded coordination efficiency, and a higher probability of network partitions (Konečný et al., 2016, Bonawitz et al., 2019). The nonlinear accuracy degradation at larger scales suggests that naive scaling strategies are insufficient and that adaptive participant selection or hierarchical aggregation architectures may be necessary.

Security gaps. Empirical evidence reveals significant vulnerabilities (Bhagoji et al., 2019, Tolpegin et al., 2020): detection rates remain below 31% even with the best defense

Table 3. Defense mechanism comparison: detection rate, accuracy preservation, and attack success rate. Values are mean ± std.

Defense	Det. (%)	Acc. (%)	Atk. Succ. (%)
FedAvg	0.0 ± 0.0	65.0 ± 5.0	87.9 ± 6.6
Krum	23.2 ± 3.0	82.7 ± 4.0	76.0 ± 6.6
Tr. Mean	19.7 ± 2.4	78.9 ± 3.5	77.4 ± 6.1
Median	30.6 ± 3.8	73.8 ± 4.6	71.5 ± 6.2

Table 4. Attack success rate (%) by attack type and defense mechanism. Values are means across 5 seeds.

Attack	FedAvg	Krum	Tr. Mean	Median
Data Poison.	76.0	76.0	77.4	71.5
Model Poison.	81.8	76.0	77.4	71.5
Byzantine	70.2	76.0	77.4	71.5
Backdoor	84.6	76.0	77.4	71.5

(Coordinate-wise Median); the grand mean attack success rate across all four defenses is 78.2%; and under FedAvg, backdoor attacks achieve the highest success (84.6%) (Bagdasaryan et al., 2020, Sun et al., 2019). Even the most robust aggregation rule (Median) cannot reduce attack success below 71.5%, indicating a fundamental limitation of single-mechanism defenses.

Defense trade-offs. There are clear trade-offs among robust aggregation methods (Blanchard et al., 2017, Yin et al., 2018, Baruch et al., 2019). Krum offers the best balance (82.7% accuracy, 23.2% detection, 76.0% attack success), Median yields the highest detection (30.6%) at the cost of accuracy (73.8%), and Trimmed Mean provides moderate performance across all metrics (78.9% accuracy, 19.7% detection). The FedAvg baseline achieves 65.0% accuracy under attack with 0% detection, confirming that unprotected FL deployments in adversarial IoT environments are fundamentally insecure.

Heterogeneity effects. Device heterogeneity critically impacts system performance: sensors exhibit 66.3% reliability versus 97.9% for edge gateways. This disparity creates an asymmetry that adversaries can exploit—malicious updates from reliable device types (e.g., edge gateways) are less likely to be flagged as anomalous, whereas legitimate updates from unreliable devices (e.g., sensors) may be mistakenly discarded by robust aggregators, further reducing model quality.

6.2 Practical Recommendations

Based on our findings, we offer the following recommendations for practitioners deploying FL in IoT environments:

1. Deploy Krum as the default aggregation rule for its best accuracy–detection trade-off.
2. Limit deployment scale to approximately 2,000 devices per federation, or adopt hierarchical aggregation for larger deployments.
3. Implement device-type-aware participant selection to balance representation and reliability.
4. Combine robust aggregation with complementary defenses (e.g., gradient clipping, anomaly detection) to address the residual 71–85% attack success rates.

Table 5. Device heterogeneity metrics by device type. Values are mean ± std.

Device	Part. (%)	Train (s)	Reliab. (%)
Edge GW	95.1 ± 2.3	3.7 ± 0.7	97.9 ± 0.9
Smartphone	80.6 ± 6.1	13.0 ± 2.0	91.7 ± 2.5
Sensor	32.4 ± 6.2	45.4 ± 8.2	66.3 ± 6.2

6.3 Limitations and Future Work

6.3.1 Current Limitations: Our study has several limitations. First, it is simulation-based and may not capture all real-world complexities such as device mobility, software heterogeneity, and adversarial adaptiveness. Second, it focuses on computer vision tasks (MNIST, CIFAR-10, Fashion-MNIST); other domains such as natural language processing or time-series forecasting may exhibit different vulnerability profiles. Third, defense mechanisms are evaluated independently; hybrid approaches combining multiple defenses may yield superior performance but remain unexplored. Fourth, network-level security threats (e.g., eavesdropping, man-in-the-middle attacks) are not addressed.

6.3.2 Future Research Directions: Future work should explore advanced defense mechanisms such as ML-based anomaly detection for real-time attack identification, and hybrid aggregation strategies that combine the strengths of multiple robust rules. Integration of privacy-preserving techniques like differential privacy and secure aggregation (McMahan et al., 2017) is essential for production deployments. Validation on actual IoT testbeds with physical devices (Hard et al., 2018) would strengthen the ecological validity of our findings. Extensions to NLP and time-series domains (Xu et al., 2021, Rieke et al., 2020), as well as adaptive FL systems that self-tune to changing threat landscapes, represent promising research directions.

7. Conclusion

This paper presents a comprehensive empirical evaluation of large-scale federated learning security across 150 experiments with up to 5,000 IoT devices. Our key findings include: (1) FL accuracy degrades significantly beyond 2,000 devices (from 91.8% at 100 to 77.2% at 5,000); (2) attack success rates average 78.2% with detection below 31% for all defenses; (3) Krum provides the optimal balance (82.7% accuracy, 23.2% detection); (4) device heterogeneity critically impacts performance (edge gateways: 95.1% participation vs. sensors: 32.4%); (5) ANOVA confirms significant differences among defenses ($F = 378.78$, $p < 0.001$, $\eta^2 = 0.70$).

These results demonstrate that while FL can scale technically, significant security and performance challenges persist in heterogeneous IoT environments. The 78.2% grand mean attack success rate across all defenses underscores residual vulnerability. Our practical recommendations—including default Krum aggregation, scale limits of 2,000 devices, and device-aware selection—provide actionable guidance for IoT practitioners. Future work should explore adaptive hybrid defenses, real-world validation on physical IoT testbeds, and integration with differential privacy to enhance both security and utility in production deployments.

Acknowledgements

This research was supported by the Laboratory of Internet of Things & Human Centered Design, Faculty of Vocational Studies, Universitas Brawijaya.

References

Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., Shmatikov, V., 2020. How to backdoor federated learning. *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, PMLR, 2938–2948.

Baruch, G., Baruch, M., Goldberg, Y., 2019. A little is enough: Circumventing defenses for distributed learning. *Advances in Neural Information Processing Systems*, 32.

Bhagoji, A. N., Chakraborty, S., Mittal, P., Calo, S., 2019. Analyzing federated learning through an adversarial lens. *International Conference on Machine Learning*, PMLR, 634–643.

Blanchard, P., El Mhamdi, E. M., Guerraoui, R., Stainer, J., 2017. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in Neural Information Processing Systems*, 30, 119–129.

Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečnỳ, J., Mazzocchi, S., McMahan, B. et al., 2019. Towards federated learning at scale: System design. *Proceedings of Machine Learning and Systems*, 1, 374–388.

Cao, X., Fang, M., Liu, J., Gong, N. Z., 2021. Provably secure federated learning against malicious clients. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35number 8, 6885–6893.

Fang, M., Cao, X., Jia, J., Gong, N. Z., 2020. Local model poisoning attacks to byzantine-robust federated learning. *29th USENIX Security Symposium (USENIX Security 20)*, 1605–1622.

Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., Ramage, D., 2018. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*.

IoT Analytics, 2024. State of IoT 2024: Number of connected IoT devices growing 13% to 18.8 billion globally. `https://iot-analytics.com/number-connected-iot-devices/`.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R. et al., 2021. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14number 1–2, 1–210.

Konečnỳ, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., Bacon, D., 2016. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*.

Lamport, L., Shostak, R., Pease, M., 1982. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3), 382–401.

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V., 2020a. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2, 429–450.

Li, X., Huang, K., Yang, W., Wang, S., Zhang, Z., 2020b. On the convergence of fedavg on non-iid data. *International Conference on Learning Representations*.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., y Arcas, B. A., 2017. Communication-efficient learning of deep networks from decentralized data. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, PMLR, 1273–1282.

Pillutla, K., Kakade, S. M., Harchaoui, Z., 2022. Robust Aggregation for Federated Learning. *IEEE Transactions on Signal Processing*, 70, 1142–1154.

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., Bakas, S., Galtier, M. N., Landman, B. A., Maier-Hein, K. et al., 2020. The future of digital health with federated learning. *NPJ Digital Medicine*, 3number 1, Nature Publishing Group, 1–7.

Sun, Z., Kairouz, P., Suresh, A. T., McMahan, H. B., 2019. Can you really backdoor federated learning? *NeurIPS Workshop on Federated Learning*.

Tolpegin, V., Truex, S., Gursoy, M. E., Liu, L., 2020. Data poisoning attacks against federated learning systems. *European Symposium on Research in Computer Security*, Springer, 480–501.

Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D., Khazaeni, Y., 2020. Federated learning with matched averaging. *International Conference on Learning Representations*.

Xie, C., Koyejo, O., Gupta, I., 2018. Generalized byzantine-tolerant sgd. *arXiv preprint arXiv:1802.10116*.

Xu, J., Glicksberg, B. S., Su, C., Walker, P., Bian, J., Wang, F., 2021. Federated learning for healthcare informatics. *Journal of Healthcare Informatics Research*, 5number 1, Springer, 1–19.

Yang, Q., Liu, Y., Chen, T., Tong, Y., 2019. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology*, 10(2), 1–19.

Yin, D., Chen, Y., Ramchandran, K., Bartlett, P., 2018. Byzantine-robust distributed learning: Towards optimal statistical rates. *International Conference on Machine Learning*, PMLR, 5650–5659.

Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., Chandra, V., 2018. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*.