International Journal of Scientific Research and Engineering Development-– Volume 8 Issue 5, Sep-Oct 2025
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Review of Security and Privacy in Federated Learning

Tanvi Kansagra[1], Nisha M. Vadodariya [2]
1. (B Tech in Computer Engineering, Atmiya University, Rajkot, India
Email: tanvibenkansagra@gmail.com)
2. (Faculty of Engineering and Technology (CE), Atmiya University, Rajkot, India
Email: nisha.vadodariya@atmiyauni.ac.in)

Abstract- By enabling cooperative model training on dispersed datasets without sharing raw data, Federated Learning (FL), a decentralized paradigm, guarantees adherence to privacy laws like the GDPR. Although FL reduces the hazards of centralized data disclosure, it introduces new vulnerabilities through parameter and gradient exchanges. While backdoors, Byzantine attacks, and poisoning are significant security dangers, inference-based approaches such as Deep Leakage from Gradients (DLG) present privacy risks. Secure multi-party computation (SMC), homomorphic encryption (HE), differential privacy, and robust aggregation techniques are some of the countermeasures. Despite progress, there are still unanswered research questions for safe and scalable FL, such as data heterogeneity, privacy vs. model utility trade-offs, and protecting decentralized systems.

I. Introduction: The Two-Sided Sword of Generative AI

Large amounts of sensitive data are sent, stored, and processed using the classical Machine Learning (ML) paradigm, which is defined by centralized data gathering and computation. This poses serious confidentiality difficulties by nature. This framework has become unsustainable due to the pervasive nature of data breaches and the enforcement of legally binding constraints on data use and protection, such as the General Data Protection Regulation (GDPR) [1, 1].

Federated Learning (FL), introduced as a distributed learning framework, provides a definitive solution [1, 1 FL allows multiple clients (participants) to collaboratively train a common global model. Crucially, the sensitive raw data remain strictly local, at the edge of the network, and are never shared. Only local model parameters or gradients are exchanged among collaborative agents and a central server (aggregator). cyberattacks more sophisticated and widespread, resulting in a dynamic conflict where both sides use the same core technology.

FL's primary selling point is its ability to facilitate high-performance model generalization while essentially advancing data privacy. Nevertheless, an open attack surface is produced by the constant swapping of model parameters. Adversarial techniques can introduce malicious updates to corrupt the model's availability and integrity, or they can mathematically infer private data points from these shared parameters, jeopardizing confidentiality. Successful FL system development is therefore inextricably linked to the provision of strong and verifiable security guarantees that meet the strict requirements of contemporary regulatory compliance as well as the high technical standards of machine learning efficacy.

Review of Security and Privacy in Federated Learning

Federated Learning (FL) enables collaborate model training on distributed datasets without sharing raw data

Security Threats

- Poisoning
- Backdoor
- Byzantine

Privacy Risks

- Inference-based attacks

Challenges and Research Directions

- Data heterogeneity
- Privacy-utility trade offs
- Decentralized architectures

II. Federated Learning Fundamentals and Taxonomy

II.A. Fundamental Ideas and Enhancement

Federated Learning creates a collaborative, iterative training process with N participants.

The global objective function in FL is designed to minimize the aggregated loss g(θ) across all participating datasets Dn.

Formally this minimization is expressed as:

$g(\theta)=n\in N\sum dmnF(Pn,\theta)$,

where mn is the size of the dataset Pn for participant n, d is the total size of all aggregated data, and F(Pn ,θ) is the local loss function. The Federated Averaging (FedAvg) algorithm is the foundational algorithm for centralized FL, involving four main steps: initialization, distributed local learning, parameter update, and central aggregation.

II.B. Categorization of Federated Learning Architectures

FL deployments are categorized based on the scale of participation, data partitioning, and network topology.

1) Taxonomy by Data Partitioning (Data Composition)

Horizontal FL (HFL): Datasets share the same feature space but have minimal overlap in samples (e.g., Cross-Device FL scenarios) [1, 1]. Clients train the exact same model architecture.

Vertical FL (VFL): Datasets share a large overlap in samples (users) but differ significantly in features (e.g., collaboration between organizations) [1, 1].For collaborative model building, VFL usually requires homomorphic encryption.

Federated Transfer Learning (FTL): Utilizes knowledge transfer techniques and is used when sample and feature overlap is minimal [1, 1].

2) Taxonomy by Scale and Topology

Cross-Device FL (CDFL): Consists of a vast number of IoT devices or mobile terminals with limited resources. Communication snags and client instability are important traits. Statistical privacy techniques like Differential Privacy are preferred due to resource constraints [1, 1].

Cross-Silo FL (CSFL): Customers are data centers or organizations with a lot of resources, few in number, and stable [1, 1]. The high computational cost of robust cryptographic techniques (SMC or HE) is justified in this context.

Centralized FL (CFL): The typical topology in which all participants are coordinated by a central server. The aggregator's single point of failure is the main weakness.

Decentralized FL (DFL): Participants communicate directly with one another, eliminating the need for a central authority. This increases the security surface while removing the central bottleneck.

III. Security and Privacy Threats in FL Ecosystems

Threats in FL are classified by the actor (Aggregator or Participant) and their objective (integrity/availability or confidentiality) [1, 1]. Adversaries are generally classified as **Semi-honest** (passive eavesdropping) or **Malicious** (active deviation and tampering).

III.A. Aggregator-Initiated Attacks (Confidentiality)

A malicious or semi-honest aggregator, due to its global view, poses a high-impact confidentiality threat.

- **Deep Leakage from Gradients (DLG):** This is the most critical privacy threat, capable of reconstructing sensitive raw data [1, 1]. The adversary minimizes the distance loss between a dummy gradient and the victim's true uploaded gradient, accurately reconstructing the original training sample. This demonstrates that parameter sharing is sufficient for reconstructing private raw data.
- **Attacks Based on Generative Adversarial Networks (GANs):** These attacks train a GAN generator to function as a decoder that reverses the model updates, using advanced generative modeling to reconstruct realistic training data.

III.B. Attacks Started by Participants (Integrity and Confidentiality)

Attacks by malicious participants can target the confidentiality of other participants' data (privacy) or the integrity of the model (security)

Poisoning Attacks (Integrity/Availability): These aim to undermine the integrity of the global model, leading to prediction failures. They are major security threats.

- **Data Poisoning:** Introducing rigged or mislabeled data into the *local* training dataset to bias the global model [1, 1].
- **Model Poisoning:** Directly manipulating model parameters or gradients *before* uploading them.
- **Membership Inference Attacks (MIA):** Attempting to determine if a specific data record was part of a targeted client's training set.
- **Property Inference Attacks (PIA):** Aiming to extract valuable properties or attributes about the training data irrelevant to the main FL task.

IV. Privacy-Preserving and Security Countermeasures

The defense landscape combines mathematical, cryptographic, and architectural strategies, all navigating the trade-off between utility (accuracy), computation cost, and communication overhead.

IV.A. Cryptographic Methods (Confidentiality)

These methods provide information-theoretic confidentiality, ensuring data or model updates remain private during computation, suitable for resource-rich CSFL environments.

SMC, or secure multi-party computation, is made possible by protocols like pairwise masking and secret sharing that allow for joint computation on private inputs without disclosing individual contributions to the aggregator or any other party. The primary drawback is the high overhead of communication, which increases exponentially with the number of participants.

Calculations can be performed directly on encrypted data (ciphertext) thanks to homomorphic encryption (HE). Data breaches are avoided by using additive HE (such as Paillier) to conceal local gradient updates during aggregation.

IV.B. Techniques for Robustness (Availability and Integrity)

These methods protect model integrity from Byzantine attacks and poisoning.

Aggregation with Byzantine Resilience: Techniques that adapt the aggregation process to accept arbitrary inputs include:

Krum: Implicitly eliminates extreme outliers by choosing the local model update that is closest to its nearest neighbors.

Coordinate-wise Median/Trimmed Mean: These techniques improve robustness against arbitrary outliers by calculating the median or eliminating extreme parameter contributions prior to averaging.

IV.C. Architectural and Hardware Defenses

Blockchain Integration (BlockFL): Replaces or supplements the central aggregator with a decentralized ledger [1, 1]. This provides tamper-proof storage, verifies model updates, and manages incentives.

Trusted Execution Environments (TEEs): Rely on hardware separation (e.g., Intel SGX) to create a secure, isolated environment, guaranteeing confidentiality of loaded data and integrity of code execution. TEEs can be used for secure aggregation or local training [1, 1]. They are constrained by a small secure memory (Trusted Computing Base or TCB), limiting their use for training large DNN models [1, 1].

V. Challenges and Future Directions

The secure deployment of FL is hindered by persistent, multi-faceted challenges.

V.A. The Tripartite Trade-off: Utility, Privacy, and Cost

The fundamental conflict between model performance (utility), privacy strength, and resource consumption (cost) remains the greatest barrier [1, 1]. Implementing stronger privacy inevitably leads to diminished model utility or soaring costs.

V.B. Data Heterogeneity and Convergence Non-IID data distribution is a core characteristic of real-world FL that aggravates learning bias and degrades model performance [1, 1]. This challenges traditional robust aggregation methods.

V.C. Personalized Federated Learning (PFL), moving beyond the single-global-model paradigm to allow clients to train personalized local models or grouping clients with similar behaviors.

V.D. Security in Decentralized FL (DFL)

DFL solves the single-point-of-failure but expands the attack surface for backdoor insertion and model poisoning. The solution lies in robust .

VI. Conclusion

Federated Learning represents a vital paradigm shift, circumventing the privacy and communication constraints inherent in centralized architectures [1, 1]. However, the reliance on exchanging model parameters introduces novel and severe adversarial vulnerabilities. The most serious risks to model integrity and confidentiality are inference-based attacks such as DLG and active malicious manipulation through poisoning and Byzantine attacks [1, 1, 1].

A complex, hybrid defense posture is necessary for effective mitigation. Differential privacy offers the required statistical guarantees in resource-constrained settings, while cryptographic techniques (SMC and HE) provide the strongest information-theoretic protection for resource-rich environments [1, 1].

Byzantine-resilient aggregation (Krum) and proactive anomaly detection are used to achieve robustness. Future secure scalability requires that the persistent non-IID data problem be addressed by personalized FL models, that the intrinsic resource constraints of edge devices (TEEs) be addressed, and that transparent, auditable trust management systems be established through blockchain integration [1, 1].

VII. References

1. Gosselin, R., Vieu, L., Loukil, F., & Benoit, A. (2022). Privacy and Security in Federated Learning: A Survey. Applied Sciences, 12(19), 9901.
2. Zhang, J., Zhu, H., Wang, F., Zhao, J., Xu, Q., & Li, H. (2022). Security and Privacy Threats to Federated Learning: Issues, Methods, and Challenges. Security and Communication Networks, 2022, 2886795.https://doi.org/10.1155/2022/2886795
3. Hu, K., Gong, S., Zhang, Q., Seng, C., Xia, M., & Jiang, S. (2024). An overview of implementing security and privacy in federated learning. Artificial Intelligence Review, 57, 204. https://doi.org/10.1007/s10462-024-10846-8

4. Li, Q., Wen, Z., Wu, Z., Hu, S., Wang, N., Li, Y., Liu, X., & He, B. (2021). A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection. arXiv preprint arXiv:1907.09693.
5. Javeed, D., Saeed, M. S., Kumar, P., Jolfaei, A., Islam, S., & Islam, A. K. M. N. (n.d.). Federated Learning-based Personalized Recommendation Systems: An Overview on Security and Privacy Challenges. IEEE Transactions on Consumer Electronics.
6. McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017).Communication-efficient learning of deep networks from decentralized data. In Proceedings of the Artificial Intelligence and Statistics (pp. 1273–1282). PMLR.
7. Zhu, L., Liu, Z., & Han, S. (2019). Deep leakage from gradients. Advances in Neural Information Processing Systems, 32, 14774–14784.
8. Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., & Shmatikov, V. (2020). How to backdoor federated learning. In International Conference on Artificial Intelligence and Statistics (pp. 2938–2948). PMLR.
9. Wang, H., Yurochkin, M., Sun, Y., Papailiopoulos, D., & Khazaeni, Y. (2020). Federated learning with matched averaging. In International Conference on Learning Representations.
10. Kim, H., Park, J., Bennis, M., & Kim, S. L. (2019). Blockchained on-device federated learning. IEEE Communications Letters, 24(6), 1279–1283.
11. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., & Seth, K. (2017). Practical secure aggregation for privacy preserving machine learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 1175–1191). ACM.
12. Lyu, L., Yu, H., & Yang, Q. (2020). Threats to federated learning: A survey. arXiv preprint arXiv:2003.02133.
13. 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