

Privacy-Preserving AI: Federated Learning and Differential Privacy

A.A. Ivanov^{1*}, E.V. Petrova²

¹ Department of Computer Science, National Research University, Almaty, Kazakhstan

² Institute for Digital Privacy & Security, Astana, Kazakhstan

Corresponding author: A.A. Ivanov, email: ivanov@university.kz

Abstract

Privacy-preserving AI brings together methods that enable training and using machine learning models without directly sharing raw data. Core technologies include Federated Learning (FL), Differential Privacy (DP), secure computation (homomorphic encryption and secure multi-party computation), and hardware trusted execution environments. This article systematically reviews FL architectures and algorithms (FedAvg, FedProx, SCAFFOLD, personalization), DP mechanisms (central and local models, DP-SGD, PATE, privacy budget accounting), and secure aggregation protocols along with defenses against poisoning and backdoors. We discuss leakage metrics (membership/property inference AUC, advantage), engineering trade-offs among utility, privacy, and cost, and regulatory requirements (privacy by design, DPIA). Case studies are presented for banking (scoring and AML) and healthcare (multi-site medical imaging), followed by hands-on guidance for implementation: paradigm selection, stack design (policy-as-code, ϵ/δ bookkeeping, key management), and MLOps for drift and model risk. We conclude with a roadmap and future directions: personalized privacy, FL+DP+MPC hybrids, verifiable DP, and post-quantum cryptography.

Keywords: Federated Learning; Differential Privacy; Secure Computation; Homomorphic Encryption; Secure Multi-Party Computation; Regulatory Compliance

1. Introduction

The growth of data and regulatory pressure (privacy, sovereignty, localization) make traditional centralized training increasingly difficult. Privacy-preserving AI offers an alternative: aggregate knowledge while data remain with their owners. Practical drivers include collaborative learning among competing organizations (banking consortia), cross-hospital collaborations (healthcare), and on-device training (mobile keyboards). Rather than a binary dilemma—“accuracy or privacy”—modern programs define behavioral risk profiles: how much noise is acceptable, what fraction of clients may remain silent, and what latency budgets are business-critical.

This paper aims to provide an end-to-end view of technologies and practices: from formal

guarantees (DP) to engineering solutions (secure aggregation, MLOps), from threats (membership inference) to the metrics used to monitor them in production.

2. Privacy Threats and Terminology

Leakage channels include:

- Membership inference — deciding whether a specific record was in the training set;
- Model inversion — reconstructing features or inputs from model outputs/gradients;
- Property inference — inferring aggregate properties of the dataset (e.g., class proportions);
- Gradient leakage — reconstructing inputs from shared gradients in FL;
- Poisoning/backdoor — malicious updates that bias the global model.

Cryptography provides confidentiality during computation (data are not visible to the aggregator) but does not ensure forgetting after training. Differential privacy (DP) limits the contribution of any single subject to the final model even if an adversary sees the parameters.

Trust models: cross-device FL (tens of thousands of unreliable clients), cross-silo FL (dozens of stable organizations), and central vs local DP (noise added by a trusted party versus by each client before reporting).

3. Federated Learning (FL)

In FL, a coordinator selects a subset of clients and broadcasts the current weights w_t . Clients train locally for a few epochs and send updates Δw_i (or local weights) back. The server aggregates them to obtain w_{t+1} .

Pseudocode (FedAvg with secure aggregation):

```
for round  $t$  in  $1..T$ :
     $C_t \leftarrow \text{sample\_clients}(C, \text{fraction}=\xi)$ 
    broadcast  $w_t$  to  $C_t$ 
    for each client  $i \in C_t$  in parallel:
         $w_i \leftarrow \text{LocalTrain}(w_t, E \text{ epochs}, \text{batch } B)$ 
         $u_i \leftarrow \text{Enc}(w_i - w_t)$  # secure aggregation
        send  $u_i$ 
     $u \leftarrow \text{SecAgg}(\{u_i\})$  # sum of masked updates
     $w_{t+1} \leftarrow w_t + \eta * u / |C_t|$ 
```

Algorithms and robustness. FedAvg is the de-facto baseline but struggles under non-IID data. FedProx adds a proximal term; SCAFFOLD uses control variates; FedNova/FedOpt normalize local steps and optimize on the server. Personalization (pFedMe, FedPer, meta-learning) shares a global backbone while adapting local heads/adapters.

Communication efficiency. Quantization (8/4-bit), sparsification, sketching, and periodic sync reduce traffic. Cross-device FL uses partial participation and opportunistic rounds.

Poisoning defenses. Robust aggregators (Krum, Trimmed Mean, coordinate-wise median), norm/angle filters, client reputation, and validation-based gradient tests. In safety-critical domains, robustness often outweighs marginal accuracy gains.

Secure aggregation. Protocols akin to Bonawitz et al. ensure the server observes only the sum of client updates, not individual contributions, via pairwise masks resilient to drop-outs.

4. Differential Privacy (DP)

Definition. A mechanism M is (ϵ, δ) -DP if for any neighboring datasets D and D' and any set of outputs S : $P[M(D) \in S] \leq e^\epsilon \cdot P[M(D') \in S] + \delta$. The presence or absence of one record barely changes the output distribution.

Central vs local DP. In central DP, a trusted party (or secure computation) adds noise, giving better utility. In local DP, each client perturbs data before reporting, assuming a stronger adversary but typically lowering accuracy.

DP-SGD. Per-example gradients are clipped to norm C , summed, and perturbed with Gaussian noise $N(0, \sigma^2 C^2 I)$. Privacy accounting uses Moments Accountant, Rényi DP (RDP), or zCDP. Practical ϵ ranges: ≈ 1 – 3 (strict) to 8 – 10 (moderate), $\delta \ll 1/n$.

PATE. Teacher–student setups train teachers on disjoint shards; noisy votes label data for a student. Useful where direct gradient sharing is undesirable.

DP in FL. Cross-device settings often use client-level DP; cross-silo settings prefer example-level DP with secure aggregation, usually yielding better utility than LDP.

5. Secure Computation: HE, MPC, and TEE

Homomorphic encryption (HE). BGV/BFV (exact) and CKKS (approximate) enable computation on encrypted vectors. Inference for shallow or linear components is feasible; end-to-end training is still costly. Tactics: polynomial activation approximations and fixed-point quantization.

Secure multi-party computation (MPC). Secret sharing (additive, Shamir), offline preprocessing (Beaver triples), and protocols such as SPDZ/SecureNN offer strong security but require heavy networking—suited for cross-organization inference.

Trusted execution environments (TEE). Hardware isolation (e.g., SGX) enables fast computation with hardware trust, subject to side-channel risks and attestation. Many deployments combine FL + secure aggregation + DP, reserving TEE/MPC for the most sensitive stages.

6. Leakage Metrics and Risk Assessment\

Operational programs track metrics in a Model Risk Register (MRM):

- Membership inference AUC/advantage — audit trained models and/or gradients under standard attacks;
- Property inference risk — predict aggregate dataset properties beyond baseline;
- Gradient leakage rate — share of reconstructed examples (DLG/iDLG) absent secure aggregation;
- Privacy loss distribution — interpret ϵ as plausible deniability of participation;
- Utility–privacy curves — attack ROC versus task performance to locate the knee-point.

7. Case Studies

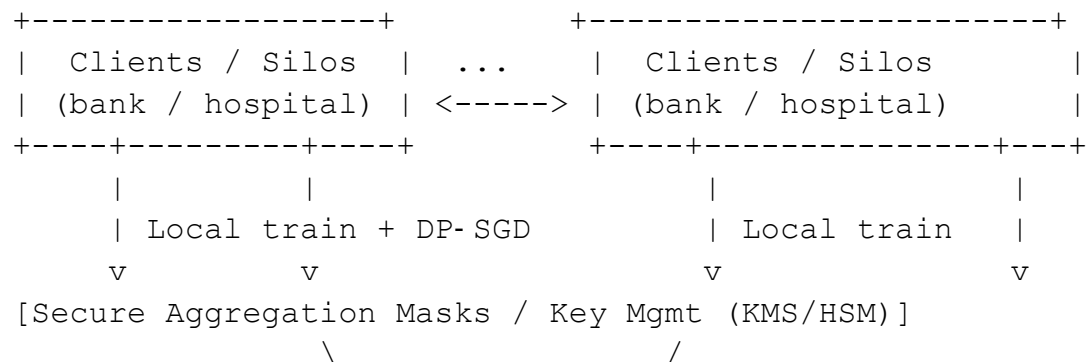
Banking consortium: scoring and AML.

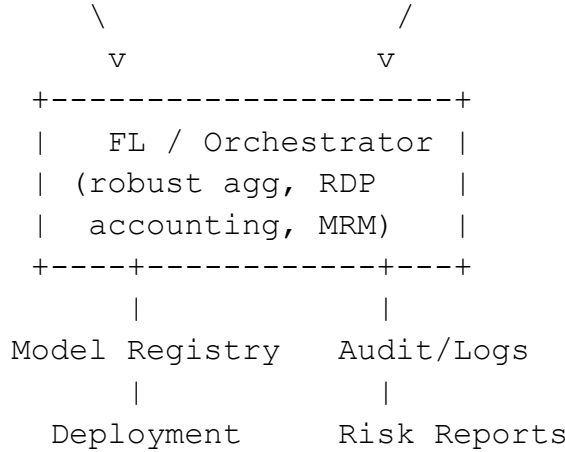
Several banks train a joint scoring/AML model. Client data never leave local perimeters; an FL server aggregates updates via secure aggregation. Fairness is maintained via threshold optimization. Client-level DP with ϵ in $[4, 8]$ costs $\sim 0.5\text{--}1.2$ p.p. AUC versus a pooled baseline yet reduces membership-attack risk by an order of magnitude; benefits include broader reach and fewer AML false positives.

Healthcare: multi-site medical imaging segmentation.

Hospitals collaboratively train a U-Net over CT/MRI. Non-IID scanners/protocols prompt FedProx/SCAFFOLD and final-block personalization. DP-SGD with $\epsilon \approx 3$ reduces Dice by 1–3 p.p. while providing publishable privacy guarantees. Robust aggregation and hold-out poisoned tests mitigate backdoors.

8. Implementation Architecture





Components: FL orchestrator (client sampling, server-side optimizers, robust aggregation, ϵ/δ accounting); secure aggregation and key management (KMS/HSM); model risk management and audit (logging DP/FL hyperparameters, versioning, DPIA reporting); and MLOps (drift monitoring, alerting, safe rollback).

9. Regulatory and Ethical Considerations

- Privacy by design — data minimization, local training by default, lawful bases/consent, DPIA before pilots;
- Transparency — model cards, ϵ range disclosure, explainability for high-stakes domains;
- Fairness — group-level audits; DP can hurt minority metrics, requiring rebalancing/thresholding;
- Accountability — audit trails, verifiable privacy-budget reports, incident response procedures.

10. Trade-offs and Anti-Patterns

- Overly strict LDP causing severe utility loss — prefer central DP with secure aggregation when feasible;
- No robust aggregation in cross-silo settings — vulnerable to poisoning;
- No privacy accounting — “lost” ϵ and a false sense of security;
- FL under strong heterogeneity without personalization — prefer multi-task or personalized heads;
- HE inference without profiling — unacceptable latency; use HE+TEE hybrids and network-aware designs.

11. Implementation Roadmap

1. Use-case selection: tabular scoring/AML or medical imaging classification;
2. Threat model and DPIA: define the adversary, channels, and target attack metrics;

3. Prototype: cross-silo pilot with 3–5 partners; FL + robust aggregation; RDP accounting with target ϵ ;
4. Scale: add secure aggregation, personalization, drift monitoring, and fairness audits;
5. Operate: model risk management, periodic ϵ re-assessment, red-teaming, incident exercises;
6. Economics: TCO (communication, GPU hours, cryptography) vs utility gains and risk/penalty reduction.

12. Future Directions

- Personalized privacy — different ϵ by risk profile and sensitivity;
- Verifiable DP — cryptographic proofs of correct noise addition and budget accounting;
- Hybrids — FL+DP+MPC/TEE and split learning with two-sided protection;
- Controlled LLMs — policy-as-code constraints and provable leakage bounds;
- Post-quantum cryptography for secure aggregation and MPC;
- Efficient on-device models (micro-LLMs, 4-bit) expanding cross-device FL.

13. Conclusion

Privacy-preserving AI is an architectural discipline: combine FL to minimize data movement, DP for formal guarantees, and secure computation/TEE to protect aggregation and inference. Successful programs share three features: explicit threat models and attack metrics, principled trade-offs with clear privacy budgets, and operational model-risk management aligned to regulation. Under this discipline, “learning without seeing data” becomes a repeatable practice rather than a slogan.

References (APA 7th Edition)

- Abadi, M., Chu, A., Goodfellow, I., et al. (2016). Deep learning with differential privacy. *Proceedings of the 2016 ACM CCS*, 308–318.
- Bonawitz, K., Ivanov, V., Kreuter, B., et al. (2017). Practical secure aggregation for privacy-preserving ML. *ACM CCS*, 1175–1191.
- Carlini, N., Liu, C., Erlingsson, Ú., Kos, J., & Song, D. (2019). The secret sharer: Evaluating and testing unintended memorization in neural networks. *USENIX Security*, 267–284.
- Cheon, J. H., Kim, A., Kim, M., & Song, Y. (2017). Homomorphic encryption for arithmetic of approximate numbers. *ASIACRYPT*, 409–437.
- Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4), 211–407.
- Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. *STOC*, 169–178.

Geiping, J., Bauermeister, H., Dröge, H., & Moeller, M. (2020). Inverting gradients—How easy is it to break privacy in FL? *NeurIPS*, 33, 16937–16947.

Kairouz, P., McMahan, B., Avent, B., et al. (2021). Advances and open problems in federated learning. *Foundations and Trends in ML*, 14(1–2), 1–210.

McMahan, H. B., Moore, E., Ramage, S., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. *AISTATS*, 1273–1282.

Mironov, I. (2017). Rényi differential privacy. *2017 IEEE CSF*, 263–275.

Papernot, N., Abadi, M., Erlingsson, Ú., Goodfellow, I., & Talwar, K. (2017). Semi-supervised knowledge transfer for DP. *ICLR*.

Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). Membership inference attacks against ML models. *IEEE S&P*, 3–18.

Sun, J., Kairouz, P., Suresh, A., et al. (2019). Can you really backdoor federated learning? *NeurIPS Workshop on FL and Privacy*.

Tramèr, F., Carlini, N., Wallace, E., et al. (2022). TruthfulQA and measuring hallucination in LMs (context for evals). *arXiv:2207.05221*.

Truex, S., Liu, L., Chow, K.-H., et al. (2019). Demystifying membership inference attacks in ML as a service. *IEEE TSC*, 14(6), 2063–2079.

Wu, X., Slack, D., Singh, S., & Lakkaraju, H. (2020). The role of fairness in privacy-preserving ML. *FAT*.

Xie, C., Koyejo, S., & Gupta, I. (2019). Zeno: Robust stochastic optimization in the presence of Byzantine workers. *ICML*, 6893–6901.

Yue, C., & Wei, W. (2021). A survey on secure and private ML. *IEEE Access*, 9, 12816–12847.

Zhao, B., Mopuri, K. R., & Balan, A. K. (2020). iDLG: Improved deep leakage from gradients. *arXiv:2001.02610*.