

Zen Privacy: Federated Learning and Privacy-Preserving AI

Technical Report v2025.11

Antje Worring, Zach Kelling
Zen LM Research Team
research@zenlm.org

November 2025

Abstract

We present Zen Privacy, a framework for federated fine-tuning and privacy-preserving deployment of Zen models. Zen Privacy enables organizations to fine-tune models on sensitive local data without exposing that data to centralized servers. We introduce a cross-silo federated training protocol with Byzantine-robust aggregation, efficient DP-SGD with per-sample gradient clipping, and on-device inference capabilities. On privacy-utility tradeoff analysis, Zen Privacy achieves within 2.4% of centralized fine-tuning performance at $(\epsilon = 8, \delta = 10^{-5})$ differential privacy. Federated convergence across 64 siloes reaches 94.8% of centralized performance at round 200.

1 Introduction

The dominant paradigm for large language model development—centralized training on massive data aggregated from diverse sources—creates fundamental privacy tensions. Healthcare systems cannot share patient records with model providers. Financial institutions cannot expose trading data. Government agencies cannot centralize classified information. Yet all of these organizations would benefit from language models fine-tuned on their specific domain data.

Federated learning offers a resolution: train models collaboratively across multiple data-holding institutions without any institution sharing raw data. Only model updates (gradients or weight deltas) are communicated, and these can be further protected by differential privacy and secure aggregation.

Zen Privacy addresses the specific challenges of federated fine-tuning for large language models:

- **Scale:** Federated LLM fine-tuning requires transmitting billions of parameters per round.
- **Heterogeneity:** Participating siloes have different data distributions, compute capabilities, and availability.
- **Byzantine robustness:** Malicious or malfunctioning siloes must not corrupt the global model.
- **Privacy accounting:** Differential privacy budgets must be tracked precisely across training rounds.

2 Federated Fine-Tuning Protocol

2.1 Cross-Silo Setting

Zen Privacy operates in the cross-silo federated learning setting, where a fixed set of K institutional participants (siloes) collaborate to fine-tune a shared global model. Each silo k holds private dataset $\mathcal{D}_k$ with $|\mathcal{D}_k| = n_k$ examples.

The federated objective minimizes the weighted average loss:

$$\min_{\theta} \mathcal{L}(\theta) = \sum_{k=1}^K \frac{n_k}{N} \mathcal{L}_k(\theta), \quad N = \sum_k n_k \quad (1)$$

2.2 FedAvg with Parameter-Efficient Fine-Tuning

Transmitting full model gradients for a 32B parameter model is prohibitive even in cross-silo settings ($32\text{B} \times 4 \text{ bytes} \approx 128 \text{ GB}$ per round). Zen Privacy combines federated averaging with LoRA (Low-Rank Adaptation) to reduce communication by $1000\times$:

$$\theta = \theta_0 + \frac{\alpha}{r} \mathbf{A} \mathbf{B}, \quad \mathbf{A} \in \mathbb{R}^{d \times r}, \mathbf{B} \in \mathbb{R}^{r \times d'} \quad (2)$$

Only $\mathbf{A}$ and $\mathbf{B}$ (rank $r = 16$) are communicated per round. For a 32B model with 64 layers, this reduces communication to $\approx 180 \text{ MB}$ per round—achievable over enterprise WAN links.

2.3 Federated Averaging Algorithm

Server, round t :

1. Broadcast current LoRA weights $\{\mathbf{A}_t, \mathbf{B}_t\}$ to all K siloes.
2. Receive local updates $\{\Delta \mathbf{A}_k, \Delta \mathbf{B}_k\}$ from $S \leq K$ available siloes.
3. Aggregate: $\mathbf{A}_{t+1} = \mathbf{A}_t + \sum_{k \in S} \frac{n_k}{N_S} \text{Clip}(\Delta \mathbf{A}_k, C)$
4. Apply Byzantine filter (Section ??).

Silo k , local update:

1. Initialize from server weights $\{\mathbf{A}_t, \mathbf{B}_t\}$.
2. Run $E = 3$ local SGD epochs on $\mathcal{D}_k$ with DP-SGD (Section ??).
3. Compute and clip delta: $\Delta \mathbf{A}_k = \text{Clip}(\mathbf{A}_k^{\text{local}} - \mathbf{A}_t, C)$.
4. Return $\{\Delta \mathbf{A}_k, \Delta \mathbf{B}_k\}$ to server.

3 Differential Privacy

3.1 DP-SGD

Each silo applies DP-SGD during local training to bound the privacy leakage of individual training examples. For each mini-batch $\mathcal{B}$:

$$\tilde{g}_t = \frac{1}{|\mathcal{B}|} \left(\sum_{i \in \mathcal{B}} \text{Clip}(g_i, C) + \mathcal{N}(0, \sigma^2 C^2 \mathbf{I}) \right) \quad (3)$$

where C is the clipping norm and σ is the noise multiplier. Per-sample gradient clipping requires computing individual gradients, which for large models adds computational overhead. We apply *ghost clipping* to reduce this overhead to $O(1)$ memory beyond standard backpropagation.

3.2 Privacy Accounting

We use the Rényi Differential Privacy (RDP) accountant to track privacy budget across training steps. For T rounds of local training with batch sampling rate $q = |\mathcal{B}|/n_k$, noise multiplier σ , and Rényi order α :

$$\epsilon(T, \delta) = \min_{\alpha} \left(T \cdot \text{RDP}_{\alpha}(q, \sigma) + \frac{\log(1/\delta)}{\alpha - 1} \right) \quad (4)$$

Table 1: Privacy budget ϵ at $\delta = 10^{-5}$ for various configurations

Rounds	σ	q	ϵ	Downstream Acc.
100	1.0	0.01	2.4	91.2%
200	1.0	0.01	4.8	93.8%
200	0.5	0.01	8.1	95.4%
500	1.0	0.01	12.1	96.8%
No DP (baseline)	—	—	∞	97.8%

Target deployment uses ($\epsilon = 8, \delta = 10^{-5}$) at round 200, achieving 95.4% of non-private performance—within 2.4% of the centralized baseline.

4 Byzantine-Robust Aggregation

4.1 Byzantine Threat Model

We assume at most $f < K/2$ siloes are Byzantine: they may send arbitrary updates designed to corrupt the global model (model poisoning attacks). Byzantine robustness is achieved through:

- **Coordinate-wise median:** Replace weighted average with coordinate-wise median.
- **Cosine similarity filtering:** Discard updates with cosine similarity $< \tau$ to the geometric median.
- **Norm clipping:** All updates clipped to bounded norm before aggregation.

4.2 Aggregation Rule

The Zen Privacy aggregation rule (ZenAgg) combines clipping and trimmed mean:

$$\text{ZenAgg}(\{\Delta_k\}) = \frac{1}{K - 2f} \sum_{k \notin \text{top-}f, \text{bottom-}f} w_k \Delta_k \quad (5)$$

where sorting is by ℓ_2 norm of the update. ZenAgg is provably Byzantine-robust for $f < K/4$ with utility guarantee $O(f/K)$ degradation versus honest aggregation.

Table 2: Byzantine robustness: model accuracy under attack (K=64, f=16)

Attack Type	FedAvg	ZenAgg
Random noise	48.2%	94.1%
Sign flip	12.4%	93.8%
Label flip (30%)	71.4%	92.4%
FGSM poisoning	34.8%	91.8%
No attack (baseline)	96.4%	94.8%

5 Secure Aggregation

Beyond DP, Zen Privacy supports cryptographic secure aggregation where the server learns only the aggregate update, not any individual silo’s contribution:

$$\text{Server learns: } \sum_k \Delta_k, \quad \text{not: } \Delta_k \text{ for any } k \quad (6)$$

This is achieved via additive secret sharing with pairwise masking: each silo k generates random masks r_{kj} for each peer $j \neq k$, and submits $\tilde{\Delta}_k = \Delta_k + \sum_{j>k} r_{kj} - \sum_{j<k} r_{jk}$. The masks cancel in the aggregate.

Overhead: $2\times$ communication, 15% compute overhead. Enables deployment settings where even the aggregation server must not see individual updates.

6 On-Device Inference

6.1 Quantized Inference

For on-device inference where data cannot leave user devices, Zen Privacy provides 4-bit quantized model variants:

Table 3: On-device inference: model size vs. performance

Model	Precision	Size	RAM	MMLU
Zen 14B	INT8	14 GB	18 GB	76.4%
Zen 14B	INT4	7 GB	10 GB	74.8%
Zen 7B	INT4	4 GB	6 GB	71.2%
Zen 3B	INT4	2 GB	3.5 GB	65.8%

The 7B INT4 model runs on consumer GPUs (RTX 4090: 24 GB VRAM) and high-end Apple Silicon Macs (M3 Max: 48 GB unified memory).

6.2 Federated Fine-Tuning on Edge Devices

Zen Privacy extends federated learning to edge devices (cross-device setting) for applications like keyboard prediction and personalized assistants. Key adaptations:

- Micro-batch size $|\mathcal{B}| = 4$ to fit in device RAM.
- Single local epoch $E = 1$ to minimize battery consumption.
- Gradient compression: top- k sparsification retaining 1% of gradient coordinates.
- Asynchronous aggregation: devices contribute updates when charging and connected.

7 Utility-Privacy Tradeoff Analysis

Table 4: Utility-privacy tradeoff across tasks at ($\epsilon = 8, \delta = 10^{-5}$)

Task	Non-private	DP Local	DP Federated	Gap
Medical NER (F1)	0.918	0.892	0.901	0.017
Financial sentiment	92.1%	89.4%	90.8%	1.3%
Legal QA	81.4%	78.2%	79.8%	1.6%
Code completion	84.8%	81.2%	82.4%	2.4%

8 Federated Convergence Results

Table 5: Federated convergence vs. centralized training (K=64 siloes)

Round	Federated (no DP)	Federated (DP $\epsilon = 8$)	Centralized
50	81.4%	78.2%	84.8%
100	88.4%	84.8%	91.2%
150	92.4%	89.4%	94.8%
200	94.8%	91.4%	97.8%

Federated (no DP) reaches 97.0% of centralized accuracy at round 200. With DP ($\epsilon = 8$), the gap is 6.4 percentage points—acceptable for most enterprise fine-tuning applications.

9 Compliance Framework

Zen Privacy’s federated architecture satisfies key data localization requirements:

Table 6: Regulatory compliance coverage

Regulation	Requirement	Zen Privacy Support
GDPR (EU)	Data minimization, right to erasure	Local data only; model unlearning
HIPAA (US)	PHI protection	On-premise compute, DP-SGD
CCPA (California)	Consumer data rights	On-device inference option
China PIPL	Cross-border data prohibition	Local silo, no egress
PDPA (Thailand)	Data localization	Silo-local training

10 Conclusion

Zen Privacy demonstrates that federated fine-tuning with DP-SGD and Byzantine-robust aggregation can train competitive models on sensitive distributed data. At ($\epsilon = 8, \delta = 10^{-5}$) differential privacy, federated models reach within 2.4% of centralized performance across medical, financial, and legal domains. The 64-silo federated setup achieves 94.8% of centralized accuracy at round 200. These results establish federated learning as a practical path for enterprises with strict data governance requirements to benefit from custom model fine-tuning.

References

- [1] McMahan, H.B. et al. Communication-Efficient Learning of Deep Networks from Decentralized Data. *AISTATS*, 2017.
- [2] Abadi, M. et al. Deep Learning with Differential Privacy. *CCS*, 2016.
- [3] Mironov, I. Rényi Differential Privacy. *CSF*, 2017.
- [4] Blanchard, P. et al. Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent. *NeurIPS*, 2017.
- [5] Bonawitz, K. et al. Practical Secure Aggregation for Privacy-Preserving Machine Learning. *CCS*, 2017.