Mjölnir: Breaking the Shield of Perturbation-Protected Gradients via Adaptive Diffusion

Xuan Liu[1], Siqi Cai[2], Qihua Zhou[3], Song Guo[4*], Ruibin Li[1], Kaiwei Lin[2]

[1]The Hong Kong Polytechnic University, Hong Kong
[2]Hubei Key Laboratory of Transportation Internet of Things, Wuhan University of Technology, Wuhan, China
[3]College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China
[4]Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong
xuan18.liu@connect.polyu.hk, csiqi@whut.edu.cn, qihuazhou@szu.edu.cn, songguo@cse.ust.hk
ruibin.li@connect.polyu.hk, 297662@whut.edu.cn

Abstract

Perturbation-based mechanisms, such as differential privacy, mitigate gradient leakage attacks by introducing noise into the gradients, thereby preventing attackers from reconstructing clients' private data from the leaked gradients. However, can gradient perturbation protection mechanisms truly defend against all gradient leakage attacks? In this paper, we present the first attempt to break the shield of gradient perturbation protection in Federated Learning for the extraction of private information. We focus on common noise distributions, specifically Gaussian and Laplace, and apply our approach to DNN and CNN models. We introduce Mjölnir, a perturbation-resilient gradient leakage attack that is capable of removing perturbations from gradients without requiring additional access to the original model structure or external data. Specifically, we leverage the inherent diffusion properties of gradient perturbation protection to develop a novel diffusion-based gradient denoising model for Mjölnir. By constructing a surrogate client model that captures the structure of perturbed gradients, we obtain crucial gradient data for training the diffusion model. We further utilize the insight that monitoring disturbance levels during the reverse diffusion process can enhance gradient denoising capabilities, allowing Mjölnir to generate gradients that closely approximate the original, unperturbed versions through adaptive sampling steps. Extensive experiments demonstrate that Mjölnir effectively recovers the protected gradients and exposes the Federated Learning process to the threat of gradient leakage, achieving superior performance in gradient denoising and private data recovery.

Introduction

Federated Learning (FL) is a distributed machine learning paradigm that facilitates collaborative model training without directly transmitting raw training data. Instead, it aggregates gradients or parameters shared among clients to build a global model (McMahan et al. 2017; Gong et al. 2024; Wu et al. 2023; Zhang et al. 2024). This approach preserves the privacy of raw data by keeping it within its originating domain, addressing concerns associated with traditional centralized data processing. However, FL is vulnerable to gradient inversion attacks (Geng et al. 2023; Liu et al. 2023; Zhu, Liu, and Han 2019), in which adversaries can potentially reconstruct sensitive user data from the shared gradients. This vulnerability has spurred significant research into gradient protection techniques (Tan et al. 2024a; Rodríguez-Barroso et al. 2023). *Gradient Perturbation*, such as differential privacy (DP), injecting noise into gradients to enhance privacy, has been proven to be an effective strategy for safeguarding data in FL scenarios (Wei et al. 2020; Ouadrhiri and Abdelhadi 2022; Wang, Hugh, and Li 2024).

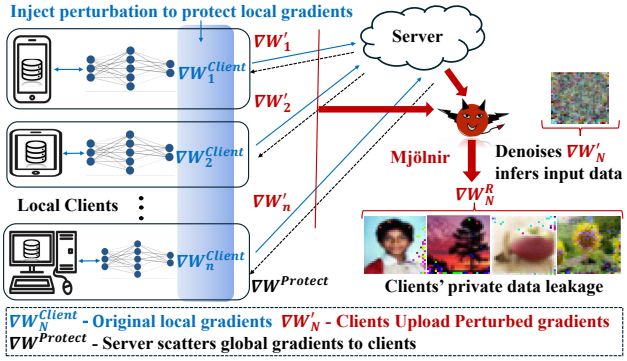


Figure 1: Threat model. The FL training process is threatened by gradient leakage attacks, where the attacker can intercept the exchanged gradients ∇W to recover the private training data. Previous work often protects the gradients by injecting perturbation into the gradients to form $\nabla W'_N$ and $\nabla W^{Protect}$. Our Mjölnir removes the perturbation injected in the protected gradients via the adaptive diffusion process.

Perturbation-based gradient protection achieves its goal by adding noise to the gradients. If a method is developed to eliminate this noise effectively, the protective mechanism of this approach becomes ineffective. *Diffusion Model* (Ho, Jain, and Abbeel 2020; Gong 2023) has the natural applicability to denoising the perturbation and is a potential approach to attack perturbation-based gradient protection. We hold the above idea based on two points: 1) Considering that gradient is the result of linear transformations applied to the training data, such as image data, we posit that if the original data are stable and predictable, then the corresponding gradient data will also be stable and predictable. 2) The essence of

*Corresponding author

gradient perturbation protection is akin to the diffusion process applied to inherently structurally stable gradient data.

In this paper, we present the *first* attempt to break the shield of gradient perturbation protection in FL based on the diffusion model. We reveal the natural diffusion properties of gradient perturbation protection and propose **Mjölnir**[1], a perturbation-resilient gradient leakage attack method and the first general gradient diffusion attack (schematic diagram shown in Fig. 2) that is capable of rendering various kinds of perturbations on gradients (e.g. Differential Privacy (Truex et al. 2020), certain layer representation perturbation (Sun et al. 2021), dynamic perturbation (Wei et al. 2021)) nearly invalid. As the first attempt, we focus on common noise distributions, specifically Gaussian and Laplace, and apply our approach to DNNs and CNNs. Our method involves constructing a surrogate model for the target attack model to obtain protected shared gradients. In the absence of the original model structure and third-party datasets, we use the surrogate gradient data supply model to generate training data for our gradient diffusion model. This trained gradient diffusion model allows us to approximate the original shared gradients from the perturbed, privacy-preserving gradients obtained by attackers. Mjölnir integrates the perturbation protection mechanism into the reverse diffusion process, adjusting the privacy-preserving gradient denoising nodes and regulating the diffusion's forward and reverse time steps with the perturbation level M as an adaptable parameter.

In summary, our key contributions include:

- We disclose the natural diffusion process in general gradient perturbation mechanisms and introduce a perturbation adaptive parameter M to adaptively adjust the diffusion step size according to the degree of perturbation.
- We propose Mjölnir, the first practical gradient diffusion attack strategy to recover the perturbed gradients, without additional access to the original model structure and third-party data, which breaks the bottleneck that existing gradient leakage attacks cannot effectively leak privacy under gradient perturbation protection.
- We demonstrate the vulnerability of gradient perturbation protection under the Mjölnir adaptive diffusion denoising process. Experimental results under the general perturbation protection FL system show that Mjölnir achieves the best gradient denoising quality and privacy leakage ability on commonly used image datasets.

Background and Related Work

FL with Perturbation Protection (FL-PP)

FL-PP includes a variety of alternative gradient perturbation techniques, such as adding random global noise to gradients, noise to specific layers, and dynamic noise addition (Zhang et al. 2022). Among these, FL with Differential Privacy (FL-DP) is the most widely used method (Shi et al. 2022; Chen et al. 2023; Tan et al. 2024b).

Generally, DP is defined based on the concept of adjacent database and has been applied in various practical areas in Artificial Intelligence to protect privacy information through adding specific perturbation, e.g., Google's RAPPORT (Erlingsson, Pihur, and Korolova 2014) and large-scale graph data publishing (Ding et al. 2021). FL-DP protects the shared model parameters or gradients between clients and the server during the FL training process by applying Local Differential Privacy (LDP (Zhao et al. 2021; Truex et al. 2020)) and (or) Differential Privacy Stochastic Gradient Descent algorithm (DPSGD (Abadi et al. 2016; Zhou et al. 2023)). In this paper we discuss both $\varepsilon - DP$ and $(\varepsilon, \delta) - DP$ (Dwork et al. 2006a,b; Abadi et al. 2016) as our sample attack background (Shi et al. 2022; Chen et al. 2023; Tan et al. 2024b):

Assumption 1. A randomized mechanism M: $D \rightarrow R$ with domain D and range R satisfies ε - differential privacy if for any two adjacent inputs *d*, *d'* $\in D$ and for any subset of outputs $S \subseteq R$, and it holds that:

$$Pr[M(d) \in S] \leq e^{\varepsilon} Pr[M(d') \in S] \tag{1}$$

Assumption 2. A randomized mechanism M: $D \rightarrow R$ with domain D and range R satisfies (ε, δ) - differential privacy if for any two adjacent inputs *d*, *d'* $\in D$ and for any subset of outputs $S \subseteq R$, and it holds that:

$$Pr[M(d) \in S] \leq e^{\varepsilon} Pr[M(d') \in S] + \delta \tag{2}$$

δ-approximation is preferably smaller than $1/|d|$. We usually apply Laplace perturbation for the definition in Eq. (1). However, as to the definition in Eq. (2), the noise perturbation mechanism needs to be Gaussian Mechanisms, which can adapt both ε and δ. Refer to Theorem A.1 in Gaussian Mechanisms (Dwork and Roth 2014), with $\nabla s = max_{D,D'}||S(D) - S(D')||$ (s is the real-value function) and c_{dp} denotes the hyperparameter used to define the DP boundary, to ensure Gaussian noise distribution $N(0, \sigma_{dp}^2)$ well preserves $(\varepsilon, \delta) - DP$, the noise scale should satisfy:

$$\sigma_{dp} \geq c_{dp} \nabla s / \varepsilon, \quad \varepsilon \in (0, 1) \tag{3}$$

$$c_{dp} \geq \sqrt{2ln(1.25/\delta)} \tag{4}$$

Taking the local participates site for example, the main actions for FL-DP can be divided into four steps: setting DP noise mechanism, local gradient clipping, local gradient perturbation, and uploading the protected parameters to the server (Wei et al. 2020). FL involves multiple participants, making composition theorem in DP necessary when applying noise. Among the composition DP methods available, including Simple Composition, Advanced Composition (Dwork and Roth 2014), and Moments Accountant (Abadi et al. 2016), we utilize Simple Composition for the following discussion. In the case where M_i satisfies $(\varepsilon, \delta) - DP$, the composition $(M_1, M_2, \cdots, M_k)$ satisfies $(\sum_{i=1}^{k} \varepsilon_i, \sum_{i=1}^{k} \delta_i)$. Aligning with the state-of-art FL-DP framework (Wei et al. 2020; Dwork et al. 2006a,b; Abadi et al. 2016; Shi et al. 2022; Chen et al. 2023; Tan et al. 2024b), We applied the following assumptions for noise calculation in local client gradient perturbation, considering both ε-DP (using the Laplace mechanism) and (ε, δ)-DP (using the Gaussian mechanism):

$$\textit{Laplace} : \sigma_{dpc} = \nabla s_c \times 1/\varepsilon \tag{5}$$

$$\textit{Gaussian} : \sigma_{dpc} = \nabla s_c \times \sqrt{2ln(1.25/\delta)}/\varepsilon \tag{6}$$

[1]Mjölnir is the hammer wielded by Thor and is renowned for its incredible power to break any shield.

where ∇s_c is the sensitivity and can be formulated as $\nabla s_c = \frac{2C}{m}$. C is the clipping threshold for bounding $||\nabla W_i|| \leq C$, where ∇W_i denotes the unperturbed gradient from the $i-th$ client and m denotes the minimum size of local datasets.

Gradient Inversion Attack (GradInv)

GradInv is a prevalent method in gradient leakage attacks and aims to steal the client's privacy information in the FL system. The primary idea for the majority of *GradInv* to recover original information is minimizing the distance between the dummy gradient ∇W_ξ and the original gradient ∇W while updating the random data x_ξ and label y_ξ until the optimized results ∇W_ξ^* and (x_ξ^*, y_ξ^*) are close enough to the original ones. The key formulation can be described as:

$$min||\nabla W_\xi - \nabla W|| : (x_\xi, y_\xi) \rightarrow (x_\xi^*, y_\xi^*) \quad (7)$$

Previous works utilize Peak Signal-to-Noise Ratio (PSNR) of the recovered images to evaluate the performance of *GradInv*, with the threshold for the success of such attacks typically hinging on the degree of detail discernible to the human eye in the reconstructed private images (Geiping et al. 2020; Zhu, Liu, and Han 2019; Liu et al. 2023).

Methodology

Mjölnir is the first Diffusion Attack Method focusing on the gradient data structure that can be applied to multiple kinds of gradient perturbation protection in FL through adaptive parameters setting on both forward and reverse process of the Gradient Diffusion Model employed in Mjölnir. The threat models targeted by Mjölnir encompass various forms of gradient perturbation. Meanwhile, the extremely similar noise mechanism on gradient perturbation protection and diffusion Markovian process provides a mathematical necessity for Mjölnir to realize an efficient privacy attack on gradient perturbation protection. Overall, our Mjölnir method (shown in Fig. 2) can be summarized in four steps:

Step (1) Get Protected Gradients. Honest-but-curious malicious attackers steal the Shared Perturbed Gradients $\nabla W'$ during the FL training process by hiding on the server side or waiting on the way of parameter sharing (Fig.1).

Algorithm 1: Gradients Extracted for Mjölnir Training

1: Stolen Protected Gradients: $F_i \rightarrow \nabla W'$;
2: Construct surrogate Model: $\nabla W' \rightarrow F^s$;
3: $j = 0$; $\epsilon \sim N(\mathbf{0}, \mathbf{I})$;
4: **for** $iteration = 1$ to $Length_{randomdataset}$ **do**
5: $\quad \nabla W_j = \partial l(F^s(x_j; W^s), y_j)/\partial W^s$;
6: $\quad$ **Save** ∇W_j;
7: $\quad j = j + 1$;

Step (2) Construct Surrogate Model. Construct Surrogate Gradients Data Supply Model F^s from the data structure of Shared Perturbed Gradients $\nabla W'$ stolen by the attacker that can output the same gradient data structure as the target attacked client's local model (Algorithm 1, Fig. 2).

Before delving into Step (3) and Step (4), two configurations that recur in the subsequent steps are defined:

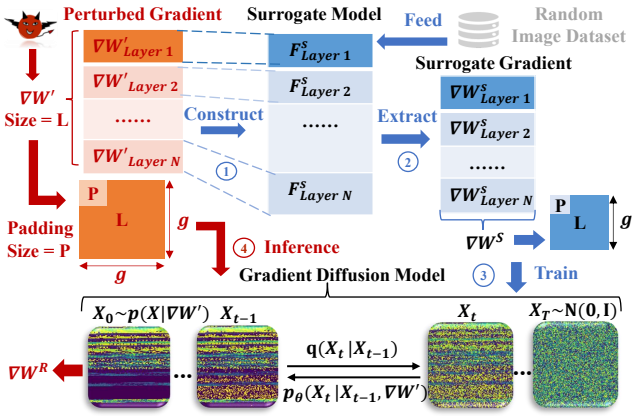


Figure 2: Mjölnir Overview. After intercepting the exchange gradients protected by unknown perturbation ($\nabla W'$) from clients, the attacker will (1) leverage the invariance of gradient data structure before and after perturbation to construct the surrogate model; (2) feed random image dataset into the surrogate model to extract surrogate clean gradients; (3) flatten the surrogate gradients and pad them to the appropriate size $g^2 = P + L$ (g is the minimum integer satisfies $g^2 > L$) to create a surrogate gradient (∇W^s) dataset for training Gradient Diffusion Model (if additional conditions are chosen to guide the training process, joint (∇W^s,$\nabla W^s_{perturbed}$) or (∇W^s,$\nabla W'$) as the training dataset, where $\nabla W^s_{perturbed}$ denote surrogate gradients applied with known perturbation; if no conditions are needed, directly use ∇W^s); (4) use the trained Gradient Diffusion Model to denoise $\nabla W'$ to generate the recovered gradient ∇W^R. From ∇W^R, the attacker can recover clients' privacy information.

[Gradient Adjustment]: Gradients each with total size L ($L = L_{\nabla W_{Layer1}} + L_{\nabla W_{Layer2}} + ... + L_{\nabla W_{LayerN}}$) are adjusted into $1 \times g \times g$ ($g^2 = L + P$: g = minimum integer satisfies $g^2 > L$; P = 0-padding size) before feeding into Gradient Diffusion Model for training or inference to adapt gradient diffusion process. The adjustment is only related to Diffusion procedures, gradients are restored to their original structure and size before further Gradient-Based Attacks and Evaluations.

[M-Adaptive Process]: Adaptive parameter M is inserted into $\nabla W'$ before inference to ensure the starting time step is appropriately positioned to maximize the denoising capability of Gradient Diffusion Model (Eq. (14) $\sim$ Eq. (16)).

Step (3) Train Gradients Diffusion Model. Construct

Algorithm 2: Gradient Diffusion Model Training

1: **if** With Condition $\nabla W'$ **then** $X_0 = (\nabla W', \nabla W_j)$;
2: **else** $X_0 = \nabla W_j$;
3: **Repeat**:
4: $\quad X_0 \sim p(X_0)$; $t \sim Uniform(1 \rightarrow T)$; $\epsilon \sim N(\mathbf{0}, \mathbf{I})$;
5: $\quad$ Take a gradient descent step on: $\nabla_\theta ||\epsilon - f_\theta(\sqrt{\gamma_t} X_0 + \sqrt{1 - \gamma_t}\epsilon, t)||^2$;
6: **until** converged;

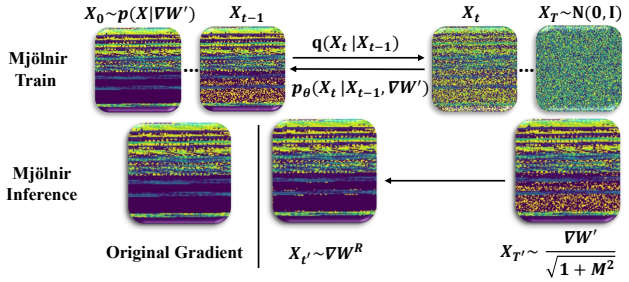


Figure 3: Visualization of Markovian gradient diffusion process. M is the noise scale of perturbation, which is set as the adaptive parameter in *[M-Adaptive Process]*. Mjölnir Train and Mjölnir Inference correspond to Algorithm 2 and Algorithm 3 respectively.

our Gradient Diffusion Model that takes into account the level of knowledge about the attacked model (e.g., whether the level of perturbation noise or the type of noise is known). Conduct *[Gradient Adjustment]* to Clean Gradients ∇W^s extracted from Surrogate Model F^s to build a training gradient dataset to train our Gradient Diffusion Model (Fig. 2, Fig.3). Algorithm 2 demonstrates a detailed training process.

Step (4) Recover Original Gradients. The stolen Shared Perturbed Gradients $\nabla W'$ are put into our trained Mjölnir Gradient Diffusion Model after *[Gradient Adjustment]* and *[M-Adaptive Process]* to generate the Recovered Gradients ∇W^R for further Gradient-Based Attack to get certain privacy information based on various demands (Algorithm 3, Fig. 2 and Fig. 3).

Algorithm 3: Generate Original Gradient

1: **if** Known Noise Scale M **then** $\mathbf{c} = \frac{1}{\sqrt{1+M^2}}$;
2: **else** $\mathbf{c} \in (0, 1)$;
3: $X_{T'} \sim \mathbf{c}\nabla W'$;
4: **for** $t = T', ..., 1$ **do**
5: **if** $t > 1$ **then** $\mathbf{z} \sim N(\mathbf{0}, \mathbf{I})$;
6: **else** $\mathbf{z} = \mathbf{0}$;
7: $X_{t-1} = \frac{1}{\sqrt{\alpha_t}}(X_t - \frac{1-\alpha_t}{\sqrt{1-\gamma_t}} f_\theta(X_t, t)) + \sqrt{1-\alpha_t}\mathbf{z}$;
8: **return** X_0; $\nabla W^R \leftarrow X_0$: ▷ For Further GradInv;

Mjölnir's Gradient Diffusion Model is inspired by DDPM(Ho, Jain, and Abbeel 2020), to be specific:

In **Step (3) Train Gradients Diffusion Model**, refer to Algorithm 2 lines 4 to 6, we set $t \in (1, T)$ as the time steps of Gaussian noise addition. T is the total time step of the forward Markovian diffusion process. $\alpha_t (0 < \alpha_t < 1)$ denote the adaptive variables at each iteration and $\gamma_t = \prod_{t=0}^{t} \alpha_t$. With the above settings, the forward process (q: with no learnable parameters) of Mjölnir's Gradients Diffusion Model can be modeled as:

$$q(X_1|X_0) = N(X_1; \sqrt{\alpha_1}X_0, (1-\alpha_1)I) \quad (8)$$

$$q(X_t|X_0) = N(X_t; \sqrt{\gamma_t}X_0, (1-\gamma_t)I) \quad (9)$$

Given (X_0, X_t), X_{t-1} can be modeled as:

$$q(X_{t-1}|X_0, X_t) = N(X_t; \mu_t, \sigma_t I) \quad (10)$$

If insert the stolen Shared Perturbed Gradients $\nabla W'$ or constructed Surrogate Perturbed Gradients $\nabla W^s_{perturbed}$ as training condition then input $X_0 = (condition, \nabla W_j)$.If not to train with the condition, then input $X_0 = \nabla W_j$.Through algebraic calculation, μ_t and σ_t in Eq. (10) can be simplified for further usage in the reverse training process(Saharia et al. 2023) as:

$$\mu_t = \frac{\sqrt{\gamma_{t-1}}(1-\alpha_t)}{1-\gamma_t} X_0 + \frac{\alpha_t(1-\gamma_{t-1})}{1-\gamma_t} X_t \quad (11)$$

$$\sigma_t^2 = \frac{1-\gamma_{t-1}}{1-\gamma_t}(1-\alpha_t) \quad (12)$$

In the reverse process (p) of Mjölnir's Gradient Diffusion training, the objective function f_θ which is trained to predict noise vector ϵ is modeled as:

$$E_{X_0,\epsilon}[\frac{(1-\alpha_t)^2}{2\sigma_t^2\alpha_t(1-\gamma_t)}||f_\theta(\sqrt{\gamma_t}X_0 + \sqrt{1-\gamma_t}\epsilon, t) - \epsilon||^2] \quad (13)$$

In **Step (4) Recover Original Gradients**, refer to Algorithm 3, two different original gradient generation processes are chosen depending on whether or not attackers know the Noise Scale of Perturbation of the Threat Model. Take FL with Gaussian Differential Privacy (Eq. (6)) as an example, consider an experienced and clever attacker who may know the DP Privacy Budget ε and the probability of information leakage δ (δ is likely to be set as 10^{-5} from usual practice in DP), combining with the sensitivity ∇S (which can be estimated if the attacker has some previous information with the target model training dataset), the noise scale can be calculated or estimated to an approximate value M (M defined as the adaptive parameter in *[M-Adaptive Process]*). So, $\nabla W'$ can be modeled as:

$$\nabla W' = \nabla W + MN(0, I) \quad (14)$$

Considering the forward Markovian gradient diffusion process in Step 3 Eq. (8) & Eq. (9), the relation between stolen perturbed gradients $\nabla W'$ and original gradients ∇W can be remodeled as:

$$\frac{1}{\sqrt{1+M^2}}\nabla W' = \frac{1}{\sqrt{1+M^2}}\nabla W + \frac{M}{\sqrt{1+M^2}}N(0, I) \quad (15)$$

Recall that our target is to generate ∇W^R, so $\frac{1}{\sqrt{1+M^2}}\nabla W'$ should be considered as X_t in the inverse process of Gradient Diffusion Model, while ∇W stands for X_0.

Correspondingly, during the construction of the recovered gradient ∇W^R, the inverse time steps can be set as T'. The relationship between T' and M is:

$$\frac{1}{1+M^2} = \prod_{t=0}^{T'} \alpha_t \quad (16)$$

Since $\gamma_t = \prod_{t=0}^{t} \alpha_t$ have been predefined and calculated during the forward Markovian diffusion process, T' can be fixed in an approximate range $T' \in (T'_-, T'_+)$, where

Datasets	Model	ε=1		ε=5		ε=10	
		$PSNR_i$	LRA	$PSNR_i$	LRA	$PSNR_i$	LRA
MINIST	GRNN(Ren, Deng, and Xie 2022)	14.70	1.000	16.54	1.000	20.19	1.000
	IG(Geiping et al. 2020)	3.721	-	5.164	-	6.067	-
	DLG(Zhu, Liu, and Han 2019)	0.222	0.676	8.025	0.752	17.08	0.909
	Mjölnir	17.87	0.851	24.17	0.895	33.09	0.927
	Mjölnir(Non-Adaptive)	17.39	0.851	24.14	0.889	32.98	0.925
	Mjölnir(Conditional)	20.87	0.887	28.39	0.924	35.14	0.957
CIFAR100	GRNN(Ren, Deng, and Xie 2022)	18.24	1.000	20.61	1.000	21.15	1.000
	IG(Geiping et al. 2020)	3.684	-	5.073	-	5.971	-
	DLG(Zhu, Liu, and Han 2019)	0.112	0.769	7.673	0.833	18.57	0.896
	Mjölnir	18.25	0.883	19.04	0.902	21.32	0.902
	Mjölnir(Non-Adaptive)	18.15	0.871	17.98	0.884	21.06	0.891
	Mjölnir(Conditional)	18.69	0.901	21.02	0.902	23.49	0.908
STL10	GRNN(Ren, Deng, and Xie 2022)	12.24	1.000	16.24	1.000	20.98	1.000
	IG(Geiping et al. 2020)	4.323	-	6.082	-	7.851	-
	DLG(Zhu, Liu, and Han 2019)	0.038	0.783	5.673	0.815	17.25	0.851
	Mjölnir	12.36	0.806	16.30	0.857	19.77	0.889
	Mjölnir(Non-Adaptive)	12.28	0.801	15.19	0.849	19.76	0.887
	Mjölnir(Conditional)	12.98	0.816	20.30	0.872	22.77	0.906

Table 1: Privacy leakage capability of Mjölnir variant models and traditional Gradients Leakage Attacks in FL-DP ($\delta = 10^{-5}$, ε = 1, 5, 10). Gray marker: Mjölnir outperforms the highest result of traditional ones.

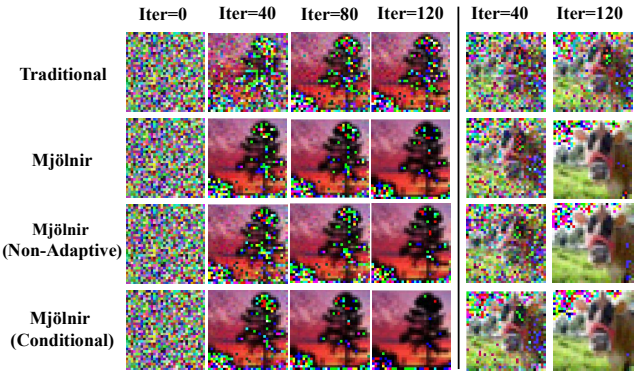


Figure 4: Comparison of the private image recovery procedures to the iterations between Mjölnir variant models and traditional Gradient Leakage Attack methods (DLG (Zhu, Liu, and Han 2019)).

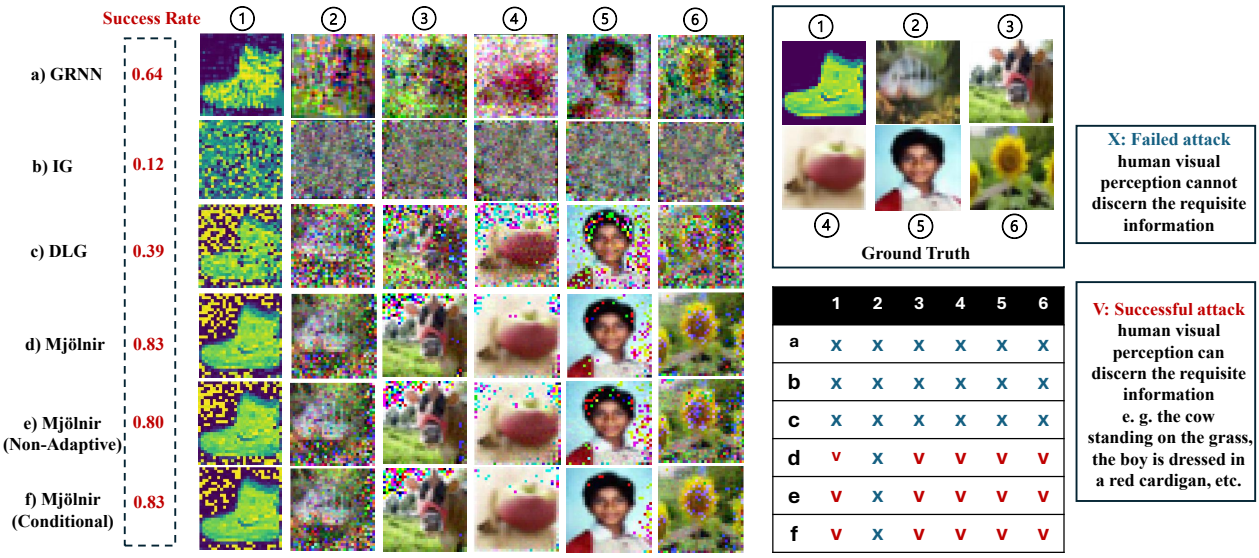
$\prod_{t=0}^{T'_-} \alpha_t < \frac{1}{1+M^2} < \prod_{t=0}^{T'_+} \alpha_t$ and T'_{-or+} are positive integers less than the total forward noise addition time step T. On the other hand, if M can not be estimated, which means the attacker knows nothing about the noise scale. Since $\frac{1}{1+M^2} \in (0, 1)$, the input value of the inference process can simply be set as $c\nabla W'$ where $c \in (0, 1)$ to adapt the Markovian forward process.

Also, if conditions allow, c can be modeled and predicted by a separate Machine Learning model according to the specific requirement of attackers.

Experiments

Experimental Setups

(A) Mjölnir Variant Attack Models. Mjölnir (trained with only unperturbed surrogate gradients), Conditional Mjölnir (trained with both perturbed gradients and unperturbed surrogate gradients), and Non-Adaptive Mjölnir (without *[M-Adaptive Process]*: no perturbation scale M as an adaptive parameter during gradient diffusion process) are presented.
(B) Benchmarks and Datasets. We employ MNIST, CIFAR100, and STL10 as client privacy datasets, which also serve as the ground truth for privacy leakage evaluation. We extract the unperturbed original gradients (∇W) of the aforementioned three datasets from the local training model of the target client as the reference benchmark of gradient denoising under the FL-PP paradigm. The Mjölnir gradient diffusion model is trained with gradients extracted from a separate dataset, FashionMNIST.
(C) Evaluation and Boundary. To evaluate the Privacy Leakage Capability, we utilize the Image Average Peak Signal-to-Noise Ratio $PSNR_i$ and the Label Recovered Accuracy LRA. These metrics are employed to assess the fidelity of the recovered images and the accuracy of the recovered labels, respectively, to the original images and ground truth labels. The boundary of Privacy Leakage Attack is aligned with previous works: the attack is considered successful if human visual perception can discern the requisite information from the recovered images. In the context of evaluating Gradient Denoising, we employ the cosine similarity $CosSimilar_g$ and the Average Peak Signal-to-Noise Ratio $PSNR_g$ as metrics to assess the quality of the Recovered Gradients ∇W^R compared to the Original Gradients ∇W. Higher values of the two metrics indicate better accuracy in the original gradient recovery.



	1	2	3	4	5	6
a	x	x	x	x	x	x
b	x	x	x	x	x	x
c	x	x	x	x	x	x
d	v	x	v	v	v	v
e	v	x	v	v	v	v
f	v	x	v	v	v	v

Figure 5: Comparisons on ground truth clients' private images and corresponding recovered privacy images from Mjölnir variant models and commonly used traditional gradient leakage attacks. ($\delta = 10^{-5}$; $\varepsilon = 10$; Success Rate: overall attack success rate)

Privacy Leakage Capability

The comparison of overall privacy leakage capability from perturbed gradients of Mjölnir and traditional Gradient Leakage Attacks (GRNN (Ren, Deng, and Xie 2022), IG (Geiping et al. 2020), and DLG (Zhu, Liu, and Han 2019)) under FL-DP, compares Image Average Peak Signal Noise Ratio $PSNR_i$ and Label Recovered Accuracy LRA of the recovered local clients' privacy information. Local clients' private training datasets are MNIST, CIFAR100, and STL10.

According to the numerical experimental results shown in Table 1 and the visualization results shown in Fig. 5, Mjölnir variant models exhibit a substantial superiority over traditional gradient leakage attacks in terms of private image leakage. On average, there is an approximately 209% increase in the recovered image $PSNR_i$ when using Mjölnir. For traditional methods, GRNN consistently achieves an LRA metric of 1.000 by using a robust generative network for label recovery, while other methods update images and labels jointly, leading to less accurate results. Moreover, upon examining the private image recovery procedures of Mjölnir variant models and traditional gradient leakage attacks illustrated in Fig. 4, it becomes evident that the attacks incorporating Mjölnir not only achieve considerably improved accuracy in the ultimate reconstruction of private images compared to conventional approaches but also showcase notable advantages in terms of attack iteration rounds and speed (Mjölnir (Conditional) $>$ Mjölnir $>$ Mjölnir (Non-Adaptive)).

Gradients Denoising under FL-DP

Among FL-DP, we choose the NbAFL framework (Wei et al. 2020) (Noising before model aggregation FL) as the threat model in this experiment due to its widespread adoption. To ensure a comprehensive evaluation of the effectiveness of the Mjölnir, we train the three Mjölnir variant models, as well as non-diffusion denoising models such as NBNet (Cheng et al. 2021), SS-BSN (Han and Yu 2023), and AP-BSN (Lee, Son, and Lee 2022), using gradients extracted from the surrogate downstream task model trained on the FashionMnist. Following that, we utilize the trained models to denoise the shared perturbed gradients intercepted by the attacker from the target clients. These target clients have locally trained downstream task models using privacy datasets (MNIST, CIFAR100, and STL10). The gradients are protected using the NbAFL(Wei et al. 2020) before being shared with the server. Based on the experimental results presented in Table 2, it is evident that Mjölnir showcases superior denoising ability for perturbed gradients under FL-DP, with an average Cosine Similarity exceeding 0.992 and a PSNR of 37.68. This represents a significant improvement of over 27% compared to non-diffusion methods. Further analysis reveals that among the three Mjölnir variant models, the performance can be ranked as follows: Mjölnir(Conditional)$>$ Mjölnir $>$ Mjölnir (Non-Adaptive). This ranking demonstrates both *M-Adaptive Process* and Conditional training enhance the denoising and generation performance of Mjölnir's Gradient Diffusion Model.

Gradients Denoising under FL-PP

To evaluate the generalization capability of Mjölnir on gradient denoising, we constructed two different types of noise (Laplace and Gaussian) randomly applied to each layer of the original gradients within the FL-PP framework. The training and inference processes of Mjölnir variant models, and non-diffusion denoising models (NBNet (Cheng et al. 2021), SS-BSN (Han and Yu 2023), and AP-BSN (Lee, Son, and Lee 2022)) are the same as the above experiments on gradients denoising under FL-DP. In contrast to attacks specifically tailored for the FL-DP framework, the gradient denoising experiments on FL-PP showcase Mjölnir's abil-

Model	MNIST		CIRAF100		STL10	
	$CosSimilar_g$	$PSNR_g$	$CosSimilar_g$	$PSNR_g$	$CosSimilar_g$	$PSNR_g$
NBNet(Cheng et al. 2021)	0.992	35.74	0.979	27.51	0.979	27.23
SS-BSN(Han and Yu 2023)	0.995	32.35	0.845	22.85	0.845	22.84
AP-BSN(Lee, Son, and Lee 2022)	0.968	29.61	0.892	24.01	0.893	24.12
Mjölnir	0.996	38.76	0.990	30.42	0.990	30.01
Mjölnir(Non-Adaptive)	0.995	38.59	0.990	30.39	0.990	29.87
Mjölnir(Conditional)	0.996	38.78	0.990	30.53	0.993	30.22

Table 2: Overall results on gradients denoising in FL-DP. Threat model setting: DP-(ε, δ) = (2,10^{-5}) with Gaussian perturbation.

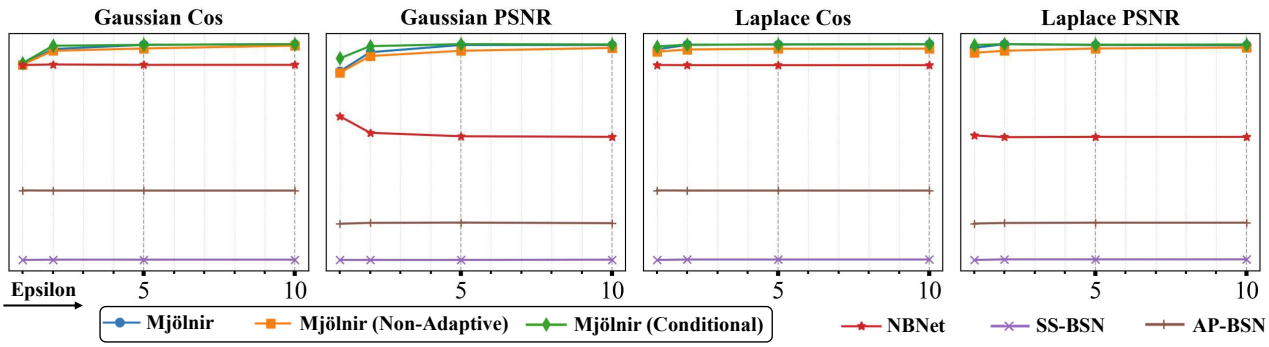


Figure 6: Gradients denoising under FL-PP: Gaussian and Laplace perturbed gradients denoising performance of $PSNR_g$ and $CosSimilar_g$ via different perturbation magnitudes. Perturbation decreases, ε increases. (Private dataset: STL10, $\delta = 10^{-5}$)

Model	Inference time (s)
NBNet(Cheng et al. 2021)	2.653
SS-BSN(Han and Yu 2023)	26.15
AP-BSN(Lee, Son, and Lee 2022)	7.138
Mjölnir	6.834
Mjölnir(Non-Adaptive)	6.834
Mjölnir(Conditional)	6.917

Table 3: Gradient denoising average inference time of Mjölnir variant models and non-diffusion denoising models under FL-PP. (Device: NVIDIA GeForce RTX 2060 GPU; Intel(R) Core(TM) i7-10870H CPU at 2.20GHz)

ity to effectively handle various types of perturbations and adapt to different magnitudes of perturbation. Referring to the experimental results illustrated in Fig. 6, it is observed that when subjected to Laplace perturbation, Mjölnir variant models exhibit an average improvement of 21.6% in PSNR and 9.2% in Cosine Similarity. Similarly, under Gaussian perturbation, Mjölnir achieves an average enhancement of 21.3% in PSNR and 9.1% in Cosine Similarity. These findings provide compelling evidence for the robustness and stability of the Mjölnir variant models. We further compared the average inference time of different denoising models under FL-PP. Results presented in Table 3 indicate that Mjölnir variant models exhibit a relative advantage in terms of gradient denoising speed, surpassing non-diffusion methods by an average improvement of 32.7% in inference time.

Limitations: (1) Mjölnir is not effective in attacking perturbations that are not based on gradient diffusion (noise perturbation) such as representation perturbation. (2) The overall effectiveness of Mjölnir in reconstructing privacy datasets is also bounded by the selected subsequent Gradient Leakage Attacks. (3) Mjölnir is used to attack CNN and (or) DNN-based models. Experimental results suggest that it struggles with transformer-based models due to their complex structure and larger number of parameters, making it difficult to accurately denoise gradients and recover data.

Defense Strategies: Possible defense strategies against Mjölnir can be approached by preserving the privacy of original data, preserving the target attack models, and shared gradients protection. Mjölnir effectively leaks privacy data by attacking the perturbed shared gradients, which suggests that the possible defense approaches against Mjölnir should either be non-perturbation gradient protection methods or non-gradient privacy-preserving methods.

Conclusion

This paper makes the first attempt to investigate the diffusion property of the widely used perturbation-based gradient protection. To reveal potential vulnerabilities, we propose a novel Perturbation-Resilient Gradient Leakage Attack via an adaptive diffusion process. This effective attack paradigm deactivates perturbation protection by leveraging the denoising capability of diffusion models without access to clients' models and external data. Based on Mjölnir, we wish to enhance public consciousness of the privacy leakage issues of existing perturbation-based defenses on gradients.

Acknowledgements

This research was supported by fundings from the Hong Kong RGC General Research Fund (152244/21E, 152169/22E, 152228/23E, 162161/24E), Research Impact Fund (No. R5011-23F, No. R5060-19), Collaborative Research Fund (No. C1042-23GF), NSFC/RGC Collaborative Research Scheme (No. CRS_HKUST602/24), Theme-based Research Scheme (No. T43-518/24-N), Areas of Excellence Scheme (No. AoE/E-601/22-R), and the InnoHK (HKGAI). We would also like to thank Dr. Chuang Hu of Wuhan University for his valuable suggestions on the naming of the attack methodology and for insightful discussions.

References

Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H. B.; Mironov, I.; Talwar, K.; and Zhang, L. 2016. Deep Learning with Differential Privacy. CCS '16, 308–318. New York, NY, USA: Association for Computing Machinery. ISBN 9781450341394.

Chen, J.; Zhao, Y.; Li, Q.; Feng, X.; and Xu, K. 2023. FedDef: Defense Against Gradient Leakage in Federated Learning-Based Network Intrusion Detection Systems. *Trans. Info. For. Sec.*, 18: 4561–4576.

Cheng, S.; Wang, Y.; Huang, H.; Liu, D.; Fan, H.; and Liu, S. 2021. NBNet: Noise Basis Learning for Image Denoising With Subspace Projection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4896–4906.

Ding, X.; Wang, C.; Choo, K.-K. R.; and Jin, H. 2021. A Novel Privacy Preserving Framework for Large Scale Graph Data Publishing. *IEEE Transactions on Knowledge and Data Engineering*, 33(2): 331–343.

Dwork, C.; Kenthapadi, K.; McSherry, F.; Mironov, I.; and Naor, M. 2006a. Our Data, Ourselves: Privacy via Distributed Noise Generation. In *Proceedings of the 24th Annual International Conference on The Theory and Applications of Cryptographic Techniques*, EUROCRYPT'06, 486–503. Berlin, Heidelberg: Springer-Verlag. ISBN 3540345469.

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. 2006b. Calibrating Noise to Sensitivity in Private Data Analysis. In Halevi, S.; and Rabin, T., eds., *Theory of Cryptography*, 265–284. Berlin, Heidelberg: Springer Berlin Heidelberg.

Dwork, C.; and Roth, A. 2014. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4): 211–407.

Erlingsson, U.; Pihur, V.; and Korolova, A. 2014. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. CCS '14, 1054–1067. New York, NY, USA: Association for Computing Machinery. ISBN 9781450329576.

Geiping, J.; Bauermeister, H.; Dröge, H.; and Moeller, M. 2020. Inverting gradients-how easy is it to break privacy in federated learning? In *NeurIPS*, 16937–16947.

Geng, J.; Mou, Y.; Li, Q.; Li, F.; Beyan, O.; Decker, S.; and Rong, C. 2023. Improved Gradient Inversion Attacks and Defenses in Federated Learning. *IEEE Transactions on Big Data*, 1–13.

Gong, X.; Li, S.; Bao, Y.; Yao, B.; Huang, Y.; Wu, Z.; Zhang, B.; Zheng, Y.; and Doermann, D. 2024. Federated Learning via Input-Output Collaborative Distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 22058–22066.

Gong, Y. 2023. Gradient Domain Diffusion Models for Image Synthesis. *arXiv preprint arXiv:2309.01875*.

Han, Y.-J.; and Yu, H.-J. 2023. SS-BSN: Attentive Blind-Spot Network for Self-Supervised Denoising with Nonlocal Self-Similarity. Procceeding of IJCAI 2023:2305.09890.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising Diffusion Probabilistic Models. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 6840–6851. Curran Associates, Inc.

Lee, W.; Son, S.; and Lee, K. 2022. AP-BSN: Self-Supervised Denoising for Real-World Images via Asymmetric PD and Blind-Spot Network. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 17704–17713. Los Alamitos, CA, USA: IEEE Computer Society.

Liu, X.; Cai, S.; Li, L.; Zhang, R.; and Guo, S. 2023. MGIA: Mutual Gradient Inversion Attack in Multi-Modal Federated Learning (Student Abstract). *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(13): 16270–16271.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and Arcas, B. A. y. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Singh, A.; and Zhu, J., eds., *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, 1273–1282. PMLR.

Ouadrhiri, A. E.; and Abdelhadi, A. 2022. Differential Privacy for Deep and Federated Learning: A Survey. *IEEE Access*, 10: 22359–22380.

Ren, H.; Deng, J.; and Xie, X. 2022. GRNN: Generative Regression Neural Network—A Data Leakage Attack for Federated Learning. *ACM Transactions on Intelligent Systems and Technology*, 13(4): 1–24.

Rodríguez-Barroso, N.; Jiménez-López, D.; Luzón, M. V.; Herrera, F.; and Martínez-Cámara, E. 2023. Survey on federated learning threats: Concepts, taxonomy on attacks and defences, experimental study and challenges. *Information Fusion*, 90: 148–173.

Saharia, C.; Ho, J.; Chan, W.; Salimans, T.; Fleet, D. J.; and Norouzi, M. 2023. Image Super-Resolution via Iterative Refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4): 4713–4726.

Shi, S.; Hu, C.; Wang, D.; Zhu, Y.; and Han, Z. 2022. Distributionally Robust Federated Learning for Differentially Private Data. In *2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS)*, 842–852.

Sun, J.; Li, A.; Wang, B.; Yang, H.; Li, H.; and Chen, Y. 2021. Soteria: Provable Defense Against Privacy Leakage in Federated Learning From Representation Perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9311–9319.

Tan, Q.; Li, Q.; Zhao, Y.; Liu, Z.; Guo, X.; and Xu, K. 2024a. Defending Against Data Reconstruction Attacks in Federated Learning: An Information Theory Approach. *arXiv preprint arXiv:2403.01268*.

Tan, Q.; Li, Q.; Zhao, Y.; Liu, Z.; Guo, X.; and Xu, K. 2024b. Defending Against Data Reconstruction Attacks in Federated Learning: An Information Theory Approach. arXiv:2403.01268.

Truex, S.; Liu, L.; Chow, K.-H.; Gursoy, M. E.; and Wei, W. 2020. LDP-Fed: Federated Learning with Local Differential Privacy. In *Proceedings of the Third ACM International Workshop on Edge Systems, Analytics and Networking*, EdgeSys '20, 61–66. New York, NY, USA: Association for Computing Machinery. ISBN 9781450371322.

Wang, F.; Hugh, E.; and Li, B. 2024. More than enough is too much: Adaptive defenses against gradient leakage in production federated learning. *IEEE/ACM Transactions on Networking*.

Wei, K.; Li, J.; Ding, M.; Ma, C.; Yang, H. H.; Farokhi, F.; Jin, S.; Quek, T. Q. S.; and Vincent Poor, H. 2020. Federated Learning With Differential Privacy: Algorithms and Performance Analysis. *IEEE Transactions on Information Forensics and Security*, 15: 3454–3469.

Wei, W.; Liu, L.; Wut, Y.; Su, G.; and Iyengar, A. 2021. Gradient-Leakage Resilient Federated Learning. In *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*, 797–807.

Wu, C.; Zhu, Y.; Zhang, R.; Chen, Y.; Wang, F.; and Cui, S. 2023. FedAB: Truthful Federated Learning With Auction-Based Combinatorial Multi-Armed Bandit. *IEEE Internet of Things Journal*, 10(17): 15159–15170.

Zhang, R.; Chen, Y.; Wu, C.; Wang, F.; and Li, B. 2024. Multi-level Personalized Federated Learning on Heterogeneous and Long-Tailed Data. arXiv:2405.06413.

Zhang, R.; Guo, S.; Wang, J.; Xie, X.; and Tao, D. 2022. A Survey on Gradient Inversion: Attacks, Defenses and Future Directions. In *IJCAI*, 5678–5685.

Zhao, Y.; Zhao, J.; Yang, M.; Wang, T.; Wang, N.; Lyu, L.; Niyato, D.; and Lam, K.-Y. 2021. Local Differential Privacy-Based Federated Learning for Internet of Things. *IEEE Internet of Things Journal*, 8(11): 8836–8853.

Zhou, Y.; Liu, X.; Fu, Y.; Wu, D.; Wang, J. H.; and Yu, S. 2023. Optimizing the Numbers of Queries and Replies in Convex Federated Learning with Differential Privacy. *IEEE Transactions on Dependable and Secure Computing*, 1–15.

Zhu, L.; Liu, Z.; and Han, S. 2019. Deep Leakage from Gradients. In *NeurIPS*, 14747–14756.