Model Cloaking against Gradient Leakage

Wenqi Wei*†, Ka-Ho Chow†, Fatih Ilhan†, Yanzhao Wu‡†, Ling Liu†,
*Department of Computer and Information Sciences, Fordham University, New York City, New York, USA
†School of Computer Science, Georgia Institute of Technology, Atlanta, Georgia, USA
‡School of Computing and Information Sciences, Florida International University, Miami, Florida, USA

2023 IEEE International Conference on Data Mining (ICDM) | 979-8-3503-0788-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICDM58522.2023.00182

***Abstract*—Gradient leakage attacks are dominating privacy threats in federated learning, despite the default privacy that training data resides locally at the clients. Differential privacy has been the de facto standard for privacy protection and is deployed in federated learning to mitigate privacy risks. However, much existing literature points out that differential privacy fails to defend against gradient leakage. The paper presents ModelCloak, a principled approach based on differential privacy noise, aiming for safe-sharing client local model updates. The paper is organized into three major components. *First*, we introduce the gradient leakage robustness trade-off, in search of the best balance between accuracy and leakage prevention. The trade-off relation is developed based on the behavior of gradient leakage attacks throughout the federated training process. *Second*, we demonstrate that a proper amount of differential privacy noise can offer the best accuracy performance within the privacy requirement under a fixed differential privacy noise setting. *Third*, we propose dynamic differential privacy noise and show that the privacy-utility trade-off can be further optimized with dynamic model perturbation, ensuring privacy protection, competitive accuracy, and leakage attack prevention simultaneously.**

***Index Terms*—Federated learning, gradient leakage, privacy analysis**

I. Introduction

Federated learning is a machine learning paradigm where multiple devices or servers collaboratively train a shared model while keeping their data locally. Instead of sending raw data to a central server, as is common in traditional machine learning, each participant computes its model update (like gradients) locally on its own dataset [1]. Despite the default privacy due to data locality, recent studies [2]–[5]have shown that federated learning is vulnerable to gradient leakage attacks, where an adversary can exploit the shared gradients to infer the private data that was used to compute those gradients.

Differential privacy (DP) is a mathematical framework that provides a formal guarantee about the amount of individual information an algorithm reveals. When applied to federated learning, differential privacy can act as a defense against gradient leakage attacks by ensuring that the shared gradients do not disclose too much information about any single data point. However, existing research has demonstrated the ineffectiveness of differential privacy noise in defending gradient leakage attacks [2]–[4]. Even with the statistical differential privacy guarantee, the ability to defend against gradient leakage attacks may not be assured. These works primarily raise the concern that excessive randomized noise may hurt the accuracy utility of the trained global model. At the same time, insufficient perturbation for gradient masking may not prevent gradient leakage attacks. While the authors in [6] show the possibility of defending against gradient leakage attacks with reasonable accuracy performance under different privacy parameters settings, existing methods share one common challenge: how to determine the proper amount of perturbation to use for best balancing among three required properties: model privacy, model leakage prevention, and model accuracy.

This paper presents ModelCloak, a principled guidance for determining the adequate amount of model perturbation against gradient leakage attacks. Using differential privacy noise as the tool, ModelCloak strategically determines the appropriate amount of noise added for utility assurance, privacy protection, and leakage prevention. The paper is organized into three major components. *First*, we introduce the robustness trade-off, in search of the best balance between accuracy and leakage prevention. The trade-off is based on the intrinsic connection between the shared gradients and their training data throughout the federated training process. We show that the appropriate amount of differential privacy noise is essential to be large enough to ensure that the perturbed gradients no longer leak the client's private training data, and in the meantime, small enough to preserve the accuracy performance. *Second*, we demonstrate that finding the sweet spot of differential privacy noise injection with ModelCloak guidance could ensure leakage prevention and yet incur a minimal negative impact on the accuracy of federated learning. *At last*, we propose dynamic differential privacy noise to closely align the injected noise with the shared gradients. Our results extend ModelCloak with further accuracy improvement while offering strong gradient leakage prevention.

II. Gradient Leakage attacks

Threat Model. We consider gradient leakage attacks happening during data-in-use at server. This requires only the weak assumption that local training data is secure and there is no privacy-peeping proxy at client. The communications between a client and the federated server are encrypted. The adversary at the server can be honest-but-curious or malicious, aiming to disclose the private training data of victim clients. Given that the server will collect local model updates from all participating clients, the adversary may perform unauthorized reconstruction inference by model inversion from all clients.

Inferring Private Input using Gradients. Gradient leakage attacks infer the private training data from the stolen gradient

979-8-3503-0788-7/23/$31.00 ©2023 IEEE
DOI 10.1109/ICDM58522.2023.00182

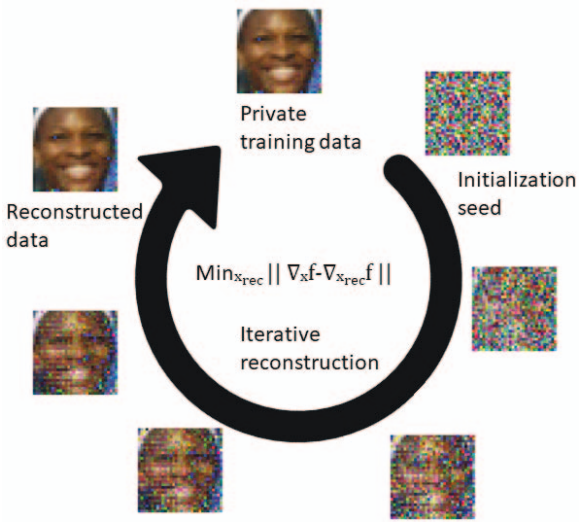


Fig. 1: Attack Reconstruction process visualization.

value $\nabla_x f$. The attacker will initiate the reconstructed data with a random seed. The reconstructed data is optimized such that the distance between the gradient of the reconstructed input $\nabla_{x_{rec}}$ and the leaked gradient value $\nabla_x f$:

$$\min ||\nabla_{x_{rec}} f - \nabla_x f||_2 \quad s.t. \quad ||x - x_{rec}||_2 \approx 0. \tag{1}$$

Figure 1 demonstrate the iterative optimization in the reconstruction attack process.

Impact of training. Due to the nature of stochastic gradient descent (SGD), gradient values would decrease as the training progresses. Thus the gradient contains less information when compared to gradient with a larger value [7]. Similar results are observed in [4], indicating that gradient leakage attack could fade away after some rounds of local training. These observations are essential for us to design model perturbations for gradient leakage prevention.

III. ModelCloak: Proper Noise Perturbation

Federated Learning with Differential Privacy. We first provide some preliminaries about differential privacy and its implementation in federated learning. Differential privacy [8] states that a randomized mechanism $\mathcal{M}$: $\mathcal{D} \rightarrow \mathcal{R}$ satisfies (ϵ, δ)-differential privacy if for any two neighboring input sets $D \subseteq \mathcal{D}$ and $D' \subseteq \mathcal{D}$, differing with only one entry: $||D - D'||_0 = 1$, $0 \leq \delta < 1$ and $\epsilon > 0$,

$$\Pr(\mathcal{M}(D) \in \mathcal{R}) \leq e^{\epsilon} \Pr(\mathcal{M}(D') \in \mathcal{R}) + \delta. \tag{2}$$

This definition implies that (ϵ, δ)-differential privacy ensures that the outcome under input D' is approximating the outcome given input D by an ϵ controlled term with at least $1-\delta$ probability. The instantiation of (ϵ, δ)-differential privacy relies on randomized response, noise injection, or both. Existing deep learning and federated learning with differential privacy [9], [10] typically deploy Gaussian noise addition by following Gaussian Mechanism [8]. Gaussian mechanism [8] states that applying Gaussian noise $\mathcal{N}(0, \varsigma^2)$ calibrated to a real-valued function: $f : \mathcal{D} \rightarrow \mathcal{R}$ with noise variance ς^2 such that $\mathcal{M}(D) = f(D) + \mathcal{N}(0, \varsigma^2)$ is (ϵ, δ)-differentially private if

$$\varsigma^2 > \frac{2\log(1.25/\delta) \cdot S^2}{\epsilon^2}, \tag{3}$$

where S represent the sensitivity in l_2 norm. Sensitivity is a key concept [11], which measures the maximum change of the function f under two neighboring datasets differing in one entry. The noise variance ς^2 is commonly replaced with $\sigma^2 S^2$ where σ is the noise scale and S is the l_2 sensitivity for better noise control [6], [9], [10], [12]–[14]:

$$\sigma^2 > \frac{2\log(1.25/\delta)}{\epsilon^2}. \tag{4}$$

In federated learning, there are two steps for adding the Gaussian noise for differential privacy protection: clipping and noise addition. The former sets up the upper bound for the gradient's l_2 norm for sensitivity. Specifically, the l_2 norm of the per-client local update is computed and compared with the pre-defined clipping bound. If the l_2 norm is larger than the clipping bound C, it will be capped at C. Otherwise, the gradient remains the same. The latter injects Gaussian noise $\mathcal{N}(0, \sigma^2 C^2 \mathbb{I})$ to the clipped local model update for noise addition. $\mathbb{I}$ denotes the size of the noise reflecting the number of gradient coordinates. Given that the Gaussian noise is added only to the per-client local model update, client-level differential privacy is ensured [6], [13].

ModelCloak Perturbation. The goal of ModelCloak is to determine the proper amount of differential privacy noise: $(\nabla_x f)^* = \overline{\nabla_x f} + \mathcal{N}(0, \varsigma^2 \mathbb{I})$ for best balancing gradient leakage resilience, accuracy, and privacy. $\overline{\nabla_x f}$ is the clipped raw gradient $\nabla_x f$. Specifically, MondelCloak selects only those differential privacy hyperparameters that result in the following amount of differential privacy noise:

$$\lambda_{\epsilon,\delta} \mathbb{I} \leq ||\nabla_x f - (\nabla_x f)^*||_2 \leq \lambda_{\max} \mathbb{I}, \tag{5}$$

where $\lambda_{\epsilon,\delta}$ is the lower bound for the injected differential privacy noise for gradient leakage prevention and $\lambda_{\max}$ is the upper bound for preserving accuracy. Given that for zero-mean Gaussian noise with a $\mathbb{I}$-coordinate vector satisfies $\mathbb{E}_{\mu \in \mathcal{N}}[||\mathcal{N}(0, \varsigma^2 \mathbb{I})||_2^2] = \varsigma^2 \mathbb{I}$, we can map the noise-induced gradient difference into the Gaussian noise variance.

For the left side, recall Equation 3, ς computed based on the given differential privacy parameters (ϵ, δ) is the lower bound noise. Therefore, we can use a minimal noise bound $\lambda_{\epsilon,\delta}$ to fulfill a given (ϵ, δ) requirement. Based on the Theorem 1 of [15] and the Lipschitz smoothness assumption, we have the following equation:

$$||x - x^*||_2 \geq \frac{||\nabla_x f - (\nabla_x f)^*||_2}{||\nabla_x f||_2}. \tag{6}$$

The numerator of Equation 6 suggests that a larger difference between the perturbed gradient and the leaked gradient can offer the larger difference between their training data counterpart. In gradient leakage attacks, if we view the perturbed gradient $(\nabla_x f)^*$ as the product of the gradient descent from a

virtual input x^*, then the reconstruction attack will force the attack optimization $(\nabla_{x_{rec}} f^*)$ to converge to the perturbed gradient $(\nabla_x f)^*$. The corresponding reconstructed data x_{rec} will lead to unrecognizable data instances x^*. In this case, the perturbed gradient no longer leaks the sensitive information about its private training data, if the difference between x^* and the original private input is large enough due to the non-linearity of deep neural networks. Furthermore, the denominator indicates that the reconstruction attack is more severe when the gradient value is larger, and the reconstruction optimization leaves a larger space for the difference between the private data and the reconstructed input from the perturbed gradient when the gradient value is smaller.

For the right side, an upper-bound noise is introduced such that the injected differential privacy noise can have a limited impact on model performance. In fact, the output stability of differential privacy states that the perturbed gradient is a $1 - e^{\epsilon}$ dominating strategy [16], slightly deviated from the main-stream gradient. Based on Equation 3, we can see that a larger noise variance can lead to a smaller ϵ when the sensitivity S and differential privacy parameter δ are fixed, implying that the perturbed gradient direction is less diverged from the original gradient, preserving the essential training information. With noise lower bound and upper bound, we are able to select an appropriate differential privacy noise that is small enough to maintain the federated model performance and yet large enough to prevent the privacy leakage of training data from the shared gradients. From Equation 5, we can derive the following proposition for the lower bound noise.

Proposition 1. ModelCloak with fixed differential privacy parameters. *Let $\mathcal{N}(0, \sigma^2 C^2 \mathbb{I})$ be the differential privacy noise $(\nabla_x f)^* = \overline{\nabla_x f} + \mathcal{N}(0, \varsigma^2 \mathbb{I})$, we have*

$$||x - x_{rec}||_2 \geq \sigma \cdot \sqrt{\mathbb{I}}. \tag{7}$$

Proof. First, $\mathbb{E}[||\mathcal{N}(0, \sigma^2 C^2 \mathbb{I})||_2^2] = C^2 \sigma^2 \mathbb{I}$. Given the definition of sensitivity and the clipping bound for sensitivity approximation, we have $C \geq S \geq ||\nabla_x f||_2$. Then by Equation 6, $\frac{||\nabla_x f - (\nabla_x f)^*||_2}{||\nabla_x f||_2} = \frac{||\overline{\nabla_x f} + \mathcal{N}(0, \varsigma^2 \mathbb{I}) - \nabla_x f||_2}{||\nabla_x f||_2} \geq \frac{||\mathcal{N}(0, \sigma^2 C^2 \mathbb{I})||_2}{||\nabla_x f||_2} \geq \frac{C \cdot \sigma \cdot \sqrt{\mathbb{I}}}{S} \geq \sigma \cdot \sqrt{\mathbb{I}}$. □

While we investigate ModelCloak with differential privacy noise, other gradient perturbation techniques, such as gradient compression [15], can also leverage ModelCloak to define the appropriate amount of perturbation with a lower bound for gradient leakage protection and an upper bound for preserving accuracy [17].

IV. Evaluating ModelCloak with Fixed Noise

Setup. We first evaluate the effectiveness of ModelCloak using fixed differential privacy noise, as studied in most federated learning with differential privacy literature [6], [10], [13]. The fixed differential privacy noise is a result of fixed clipping bound C and fixed noise scale σ according to Equation 4. We consider three benchmark datasets: MNIST, CIFAR10, and LFW[1]. **Table I** summarizes the test accuracy and hyperparameter settings for these datasets. We evaluate these datasets on a deep convolutional neural network with two convolutional layers and one fully-connected layer. We set up the federated learning system by following the simulator in [13] with a total of $N = 1000$ clients and k_t set to 10% of N per round. To demonstrate gradient leakage prevention of ModelCloak, we follow the representative attack procedure in [4]. Code is available: https://github.com/git-disl/ModelCloak.

	MNIST	CIFAR10	LFW
# training data	60000	50000	2267
# validation data	10000	10000	756
# features	28*28	32*32*3	32*32*3
# classes	10	10	62
# data/client	500	400	300
# local iteration L	100	100	100
local batch size b	5	4	3
# rounds T	100	100	60
no-private acc.	0.984	0.674	0.695

TABLE I: Benchmark datasets and parameters

Gradient Leakage Prevention with Accuracy Assurance. We consider six settings of noise scale σ, which represent six privacy budget settings according to Equation 4. We also select five clipping bound C settings, originating from existing federated learning with differential privacy research [6], [10], [13]. Consequently, we have 30 differential privacy noise settings for each dataset. To measure the attack effect under these configurations, we report the attack success rate results of reconstructing 100 images for each dataset. The reconstruction is considered failed if the mean square error (MSE) of the reconstructed data and the original data is larger than 1.4, an empirical threshold for the l_2 difference between two images with size 28 × 28 or 32 × 32. The attack optimization is iterated 300 steps maximum. In Section II, we have shown that gradient leakage attack is severer when the gradient has a larger value, which typically occurs in the early stage of training since the gradient value would decrease as training progresses for gradient descent optimization. In this set of experiments, we focus on the gradient leakage attack that happened on the first round.

Table II shows the gradient leakage prevention results for MNIST, CIFAR10 and LFW dataset, respectively. We make three observations. (1) We highlight those differential privacy noise configurations which could prevent gradient leakage in blue, and those chosen by ModelCloak in bold blue. ModelCloak would select the configuration $C = 2, \sigma = 5$ for MNIST, $C = 2, \sigma = 3$ for CFIAR10, and $C = 3, \sigma = 2$ for LFW. The reason ModelCloak determines those settings as the appropriate amount of differential privacy noise is that they deliver the best accuracy performance among those gradient leakage resilient settings. This implies that ModelCloak finds the lower bound of model perturbation, and in the meantime, maximizes the accuracy performance.

(2) Compared with ModelCloak, those highlighted gradient leakage resilient settings inject relatively larger differential

[1] https://pytorch.org/vision/stable/datasets.html

S=C		MNIST					CIFAR10					LFW				
		C=4	C=3	C=2	C=1	C=0.5	C=4	C=3	C=2	C=1	C=0.5	C=4	C=3	C=2	C=1	C=0.5
Accuracy	σ=6	0.779	0.861	0.922	0.944	0.949	0.107	0.129	0.335	0.611	0.649	0.212	0.383	0.503	0.605	0.636
	σ=5	0.835	0.863	**0.936**	0.949	0.951	0.135	0.301	0.376	0.625	0.649	0.345	0.441	0.525	0.616	0.638
	σ=4	0.867	0.923	0.937	0.951	0.954	0.249	0.335	0.559	0.636	0.651	0.404	0.505	0.571	0.635	0.64
	σ=3	0.926	0.939	0.941	0.957	0.955	0.337	0.484	**0.613**	0.648	0.666	0.506	0.567	0.607	0.642	0.64
	σ=2	0.941	0.945	0.952	0.963	0.961	0.561	**0.613**	0.644	0.672	0.674	0.569	**0.609**	0.633	0.645	0.641
	σ=1	0.952	0.959	0.963	**0.965**	0.961	0.647	0.656	0.67	**0.681**	0.673	0.637	0.644	0.647	**0.648**	0.644
Attack success rate	σ=6	0	0	0	0.10	0.60	0	0	0	0	0.31	0	0	0	0	0.25
	σ=5	0	0	**0**	0.21	0.65	0	0	0	0.06	0.42	0	0	0	0.06	0.45
	σ=4	0	0	0.123	0.43	0.70	0	0	0	0.15	0.50	0	0	0	0.16	0.56
	σ=3	0	0.02	0.28	0.64	0.71	0	0	**0**	0.31	0.52	0	0	0	0.31	0.58
	σ=2	0.13	0.29	0.46	0.76	0.73	0	**0**	0.16	0.58	0.62	0	**0**	0.19	0.59	0.64
	σ=1	0.47	0.66	0.77	**0.82**	0.81	0.17	0.33	0.56	**0.66**	0.65	0.19	0.34	0.63	**0.73**	0.69

TABLE II: Accuracy and prevention of gradient leakage with fixed differential privacy noise. 24 differential privacy settings of (σ, C) are considered for MNIST, CIFAR10, and LFW, respectively. Settings in black are vulnerable to gradient leakage, and the setting highlighted in bold black is based on maximizing accuracy. Settings in blue could prevent gradient leakage. The bold blue highlight is the sweet spot setting selected by ModelCloak, which injects just enough noise for model perturbation. The test accuracy without model perturbation is 0.984 for MNIST, 0.674 for CIFAR10, and 0.695 for LFW.

privacy noise. However, the excessive noise is unnecessary since it does no good in preventing gradient leakage as the injected noise is already enough (recall Equation 6), and yet it may bring down the accuracy performance.

(3) Clipping would impact the differential privacy noise injection from two perspectives. First, when the clipping bound C is as small as 0.5, the essential training information is distorted during the clipping operation, and the resulting accuracy can be lower than those settings with the same noise scale σ but $C = 1$. Decreasing the clipping bound C from 4 to 1 with a given noise scale σ will reduce the injected noise according to Equation 3. However, such a noise reduction can no longer improve the accuracy performance if the clipping bound is small enough to have a negative impact. Second, for a chosen noise variance $\varsigma = \sigma * C$, it is possible to find multiple different settings of σ and C. Among them, the setting with a smaller clipping bound C demonstrates better gradient leakage resilience. For example, the setting $C = 1, \sigma = 6$ has a lower attack success rate than the setting $C = 3, \sigma = 2$ but yet slightly lower accuracy for MNIST (0.944 v.s. 0.945).

A possible explanation is that a smaller clipping bound C will result in a larger σ with a fixed noise variance ς. Recall Equation 4, the corresponding ϵ-spending is smaller when the noise scale σ is large, indicating stronger differential privacy guarantee. Equation 6 also suggests that a constant noise variance in the numerator and a reduced clipping bound or sensitivity in the denominator would lead to a larger difference between the reconstructed input from the perturbed gradient and the original raw input.

ϵ-Privacy Spending. From the definition of differential privacy, the privacy spending ϵ measures the statistical privacy protection level based on Equation 2. By injecting Gaussian noise following (ϵ, δ) differential privacy, we consider the moments accountant method [9] for ϵ-privacy spending tracking. By utilizing the moments accountant computation[2], we can measure the accumulated ϵ spending with Rényi differential privacy [18] under a fixed δ. Specifically, the privacy spending

[2]https://github.com/tensorflow/privacy/blob/master/tensorflow_privacy/privacy/analysis/compute_dp_sgd_privacy.py

	σ=6	σ=5	σ=4	σ=3	σ=2	σ=1
MNIST	0.678	0.835	1.082	1.528	2.581	7.899
CIFAR10	0.678	0.835	1.082	1.528	2.581	7.899
LFW	0.519	0.64	0.832	1.18	2.01	6.331

TABLE III: ϵ-privacy spending measured at round 100, 100, and 60 for MNIST, CIFAR10, and LFW, respectively. $\delta = 1e - 5$.

ϵ is computed given the total rounds T, the noise scale σ, the privacy parameter δ, and the sampling rate q.

In this set of experiments, we report the ϵ-privacy spending results for MNIST, CIFAR10, and LFW at round 100, 100, and 60, respectively. The privacy parameter δ is set to $1e - 5$, and the sampling rate q is the client sampling rate of $10/100 = 10\%$ with 10 participating clients per round with 100 total clients. **Table III** shows the results and we make two observations. (1) Moments accountant states that the privacy spending ϵ will not change when the noise scale σ, the total learning rounds T, the privacy parameter δ, and the sampling rate q are fixed. Therefore, the ϵ-spending of MNIST is the same as that of CIFAR10, while differing from the ϵ-spending of LFW. (2) Recall Equation 4, the ϵ spending is correlated with the noise scale σ under a given δ at each noise injection step. Therefore, for a given σ, changing the clipping bound C does not add or reduce the privacy spending ϵ at a given round. Given that the noise variance is defined by σ and C, ModelCloak can benefit from a larger choice of clipping bound C while keeping the noise scale σ the same. This results in a larger amount of differential privacy noise for gradient leakage prevention without injecting additional privacy spending overhead. Consequently, ModelCloak demonstrates the capability of selecting the differential privacy noise settings that best balance three factors: privacy (ϵ-spending), accuracy, and gradient leakage resilience.

V. Improving ModelCloak with Dynamic Noise

Limitation of Fixed Noise. Based on Equation 4, the Gaussian noise for a differentially private function is calibrated with noise variance defined by noise scale σ and sensitivity S of the function. The baseline Differentially Private Stochastic Gradient Descent (DPSGD) implementation [9], followed by

most of the work [6], [10], [12] suggests using a fixed clipping parameter C to approximate sensitivity S, and with a fixed noise scale σ. The fixed pre-defined clipping bound C and the fixed noise scale σ result in constant noise variance, and thus a fixed amount of differential privacy noise is injected throughout all T rounds of federated learning.

However, gradient descent is designed to optimize towards the convergence with decreasing gradient values. Given that the sensitivity of a differentially private function is defined as the maximum amount that the function value varies when a single input entry is changed, the decreasing magnitude of the gradient would imply different sensitivity of the local model for different clients and rounds. Therefore, when the l_2 norm of the gradient is smaller than the fixed clipping bound C at a given round, using the clipping bound C can be an undesirably loose approximation of the actual l_2 sensitivity S.

Recall Equation 6, the smaller magnitude of gradient could also lower the value in the denominator, resulting in a larger difference between the reconstructed data and the private training data under a fixed noise represented by the gradient difference in the numerator. These observations suggest that (1) gradient leakage attacks weaken when the gradient magnitude gets smaller as the training progresses; (2) injecting a fixed differential privacy noise may be excessive, especially at the later stage of training near convergence. Therefore, we next investigate the possible enhancement of ModelCloak via dynamic differential privacy noise. Our approach aims to provide high gradient leakage resilience, strong differential privacy guarantee, and competitive model accuracy.

Dynamic Differential Privacy Noise. We propose to optimize ModelCloak with dynamic differential privacy parameters. Instead of using the fixed clipping bound C to approximate the sensitivity of the local SGD function, we define the sensitivity S by the max l_2 norm of the per-client gradient at local client. We argue that l_2-max can more accurately capture the actual sensitivity of the local SGD with differential privacy. With dynamic l_2-max sensitivity, the gradient perturbation is defined by using Gaussian noise $\mathcal{N}(0, \varsigma^2)$, where we define ς based on the dynamic sensitivity $S = \max ||\nabla_x f||_2$.

With the sensitivity closely aligned with the gradient trend, we can optimize the differential privacy noise with two different goals. First, if we consider a fixed noise variance $\varsigma = \sigma * C$, we will need to dynamically adjust the noise scale σ_{dyn} such that its product with the l_2-max sensitivity S remains unchanged. In this case, a smaller l_2-max sensitivity S will correspond to a larger σ_{dyn} compared to the pre-defined fixed σ, i.e., $\sigma_{dyn} \geq \sigma$. Recall Equation 5, the same noise variance with a smaller S and a larger σ could lead to stronger gradient leakage resilience. Second, if we consider a fixed noise scale, the l_2-max sensitivity will decay with the decreasing gradient magnitude, thus leading to lower differential privacy noise injected for better accuracy performance.

From Equation 5, we can derive the following proposition for the lower bound noise with dynamic noise injection.

Proposition 2. ModelCloak with dynamic differential privacy parameters.. *Let $\mathcal{N}(0, \sigma_{dyn}^2 S_{dyn}^2 \mathbb{I})$ be the differential privacy noise for gradient perturbation, where S_{dyn} is the l_2-max sensitivity and σ_{dyn} is the dynamic noise scale. we have:*

$$||x - x_{rec}||_2 \geq \sigma_{dyn} \cdot \sqrt{\mathbb{I}}. \tag{8}$$

If the noise variance ς is set to $\varsigma = \sigma * C = \sigma_{dyn} * S_{dyn}$, where the noise scale σ and the clipping bound C are fixed,

$$||x - x^*_{rec}||_2 \geq \sigma_{dyn} \cdot \sqrt{\mathbb{I}} \geq \sigma \cdot \sqrt{\mathbb{I}}. \tag{9}$$

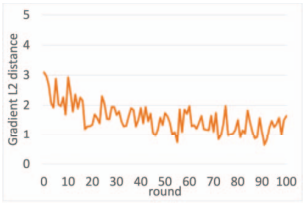
Proof. For equation 8, we have $\frac{||\mathcal{N}(0, \sigma_{dyn}^2 S_{dyn}^2 \mathbb{I})||_2}{||\nabla_x f||_2} \geq \frac{S_{dyn} \cdot \sigma_{dyn} \cdot \sqrt{\mathbb{I}}}{S_{dyn}} = \sigma_{dyn} \cdot \sqrt{\mathbb{I}}$ similar to Proposition 1. For equation 9, given $\varsigma = \sigma * C = \sigma_{dyn} * S_{dyn}$ and $C \geq S_{dyn} \geq ||\nabla_x f||_2$, we have $\sigma_{dyn} \geq \sigma$. By Proposition 1, $\frac{||\mathcal{N}(0, \sigma^2 C^2 \mathbb{I})||_2}{||\nabla_x f||_2} \geq \frac{C \cdot \sigma \cdot \sqrt{\mathbb{I}}}{S_{dyn}} = \frac{S_{dyn} \cdot \sigma_{dyn} \cdot \sqrt{\mathbb{I}}}{S_{dyn}} \geq \sigma \cdot \sqrt{\mathbb{I}}$. □

		MNIST	CIFAR10	LFW
Raw gradient	MSE	0.0014	0.0012	0.0012
	accuracy	0.984	0.674	0.695
ModelCloak-Fix	ϵ	0.835	1.082	2.581
	MSE	4.84	2.69	2.62
	accuracy	0.936	0.613	0.609
ModelCloak-Dynamic	ϵ	**0.737**	**1.041**	**2.329**
	MSE	**5.03**	**2.75**	**2.71**
	accuracy	0.937	0.614	0.608

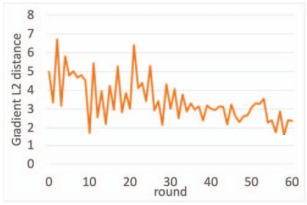
TABLE IV: MSE measurement (the larger, the more robust). With the same amount of noise, ModelCloak with dynamic noise stronger gradient leakage resilience and stronger differential privacy guarantee compared to ModelCloak with fixed noise.

VI. Evaluating ModelCloak with Dynamic Noise

ModelCloak Noise Injection with Fixed Noise Amount. We first consider the fixed noise variance under the l_2-max sensitivity and adaptive noise scale. In this case, we inject the same amount of the differential privacy noise under dynamic differential privacy parameters as the fixed noise parameters setting. The resulting noise variance will have the following relation: $\sigma_{dyn} * S_{l2} = \sigma * C$, where the fixed noise scale σ and the clipping bound C can be selected from ModelCloak with fixed differential privacy noise as provided in Table II. **Table IV** shows the MSE measurement between the reconstructed data and the raw data. The reconstructed data are from the raw gradient and gradient perturbed by ModelCloak with fixed noise and dynamic noise. We make three observations. (1) With a fixed noise variance, we are able to maintain the same model accuracy performance with better differential privacy guarantee and stronger capability for gradient leakage resilience. (2) Due to the decaying magnitude of the gradient during gradient descent, we are able to get a decaying l_2-max sensitivity accordingly. The corresponding dynamic noise scale σ_{dyn} is larger than the original fixed noise scale, and the resulting ϵ-privacy spending is lower, indicating better differential privacy guarantee (recall Equation 2 and Equation 4). (3) Also according to Equation 9, the dynamic noise scale σ_{dyn} is larger than the fixed ones and thus offers a better capability of gradient leakage prevention when



(a) MNIST.



(b) LFW.

Fig. 2: L_2 distance between the sanitized and raw per-client gradients throughout the federated training.

compared with the ModelCloak noise with a fixed clipping bound and noise scale.

Figure 2 illustrates the difference between the perturbed per-client gradients and the raw gradients throughout the federated training. This difference represents the numerator of Equation 6. When the gradient magnitude converges to zero due to gradient descent, the numerator will move towards the noise variance and the denominator gets smaller and smaller. Therefore, the difference between the reconstructed input from the perturbed gradients and the raw gradients can be enlarged. Consequently, it is unnecessary to inject the same amount of differential privacy noise in all training rounds.

ModelCloak Noise Injection with Decaying Noise Amount. We then evaluate the accuracy improvement of ModelCloak with dynamic differential privacy noise, without sacrificing the gradient leakage prevention capability. In this set of experiments, we keep a fixed noise scale σ, and the l_2-max sensitivity will decay with the decreasing gradient magnitude, thus leading to smaller amount of differential privacy noise injected. **Table V** shows the results and we make two observations. (1) According to Equation 4, the ϵ-privacy spending is only correlated with the noise scale σ with a given δ. Therefore, the smaller injected noise brought by ModelCloak with dynamic differential privacy noise with the l_2-max sensitivity and the fixed noise scale does not influence the ϵ-privacy spending, maintaining the high differential privacy guarantee. (2) ModelCloak with dynamic differential privacy noise offers higher accuracy utility when compared to ModelCloak with fixed differential privacy noise, due to the reduced injected noise by the l_2-max sensitivity. (3) According to Equation 7, a fixed noise scale could provide the same high level of gradient leakage prevention by ModelCloak with dynamic noise compared with the fixed noise.

VII. Conclusion

We have presented ModelCloak, a principled approach towards securing model sharing against gradient leakage attacks. We addressed the critical challenge for effective model perturbation: how to determine the proper amount of perturbation to achieve the sweet spot in terms of balancing privacy, accuracy, and leakage prevention. We introduced the robustness trade-off based on the behavior of gradient leakage attacks throughout the federated training process. Then, we demonstrated that ModelCloak can help identify the differential privacy parameter settings that effectively mitigate gradient leakage attacks while offering competitive accuracy performance under differential privacy guarantee. Finally, we improve ModelCloak with dynamic differential privacy noise for better performance on the privacy-utility trade-off.

		MNIST	CIFAR10	LFW
Raw gradient	accuracy	0.984	0.674	0.695
ModelCloak-Fix	ϵ	0.835	1.082	2.581
	accuracy	0.936	0.613	0.609
ModelCloak-Dynamic	ϵ	0.835	1.082	2.581
	accuracy	**0.951**	**0.627**	**0.635**

TABLE V: Accuracy improvement ModelCloak with l_2-max sensitivity-driven dynamic noise and ModelCloak with fixed differential privacy noise. We consider the blue highlighted configurations selected by ModelCloak as in Table II for MNIST ($\sigma = 5, C = 2$), CIFAR10 ($\sigma = 3, C = 2$), and LFW ($\sigma = 2, C = 3$), respectively.

Acknowledgement. This research is partially sponsored by the NSF CISE grants 2302720, 2312758, 2038029, an IBM faculty award, and a grant from CISCO Edge AI program.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *AISTATS*. PMLR, 2017, pp. 1273–1282.
[2] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *NeurIPS*, 2019, pp. 14 747–14 756.
[3] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients - how easy is it to break privacy in federated learning?" in *NeurIPS*, 2020, pp. 16 937–16 947.
[4] W. Wei, L. Liu, M. Loper, K.-H. Chow, M. E. Gursoy, S. Truex, and Y. Wu, "A framework for evaluating client privacy leakages in federated learning," in *ESORICS*. Springer, 2020, pp. 545–566.
[5] H. Yin, A. Mallya, A. Vahdat, J. M. Alvarez, J. Kautz, and P. Molchanov, "See through gradients: Image batch recovery via gradinversion," in *CVPR*. IEEE/CVF, 2021, pp. 16 332–16 341.
[6] W. Wei, L. Liu, Y. Wu, G. Su, and A. Iyengar, "Gradient-leakage resilient federated learning," in *ICDCS*. IEEE, 2021, pp. 797–807.
[7] Y. Lin, S. Han, H. Mao, Y. Wang, and B. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," in *ICLR*, 2018.
[8] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, vol. 9, no. 3–4, pp. 211–407, 2014.
[9] M. Abadi, A. Chu, I. Goodfellow, B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *CCS*. ACM, 2016, pp. 308–318.
[10] B. McMahan, D. Ramage, K. Talwar, and L. Zhang, "Learning differentially private recurrent language models," in *ICLR*, 2018.
[11] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *TCC*. Springer, 2006, pp. 265–284.
[12] L. Yu, L. Liu, C. Pu, M. E. Gursoy, and S. Truex, "Differentially private model publishing for deep learning," in *S&P*. IEEE, 2019, pp. 332–349.
[13] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," *arXiv preprint arXiv:1712.07557*, 2017.
[14] W. Wei, L. Liu, J. Zhou, K.-H. Chow, and Y. Wu, "Securing distributed sgd against gradient leakage threats," *IEEE TPDS*, vol. 34, no. 7, pp. 2040–2054, 2023.
[15] J. Sun, A. Li, B. Wang, H. Yang, H. Li, and Y. Chen, "Soteria: Provable defense against privacy leakage in federated learning from representation perspective," in *CVPR*. IEEE/CVF, 2021, pp. 9311–9319.
[16] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *IEEE FOCS*, 2007, pp. 94–103.
[17] W. Wei, X. Fan, R. Zhang, J. Zhou, and L. Liu, "Fedgcloak: Gradient cloaking for privacy-preserving federated learning," Technical Report, 2023.
[18] I. Mironov, "Rényi differential privacy," in *CSF*. IEEE, 2017, pp. 263–275.