Mind the Cost of Scaffold! Benign Clients May Even Become Accomplices of Backdoor Attack

Xingshuo Han[1], Xuanye Zhang[1], Xiang Lan[2], Haozhao Wang[3*], Shengmin Xu[2], Shen Ren[4], Jason Zeng[5], Ming Wu[5], Michael Heinrich[5], Tianwei Zhang[1]

[1]Nanyang Technological University,[2]Fujian Normal University, [3]Huazhong University of Science and Technology

[4]Continental Automotive Singapore, [5]Zero Gravity Labs, [*]Corresponding

{*xingshuo001, C200212, tianwei.zhang*}*@ntu.edu.sg*,{*jason, ming, michael*}*@0g.ai*

{*lanxiang0113, smxu1989*}*@gmail.com*, *hz_wang@hust.edu.cn, shen@shenren.org*

Abstract

By using a control variate to calibrate the local gradient of each client, Scaffold has been widely known as a powerful solution to mitigate the impact of data heterogeneity in Federated Learning. Although Scaffold achieves significant performance improvements, we show that this superiority is at the cost of increased security vulnerabilities. Specifically, this paper presents `BadSFL`*, the first backdoor attack targeting Scaffold, which turns benign clients into accomplices to amplify the attack effect. The core idea of* `BadSFL` *is to uniquely tamper with the control variate to subtly steer benign clients' local gradient updates towards the attacker's poisoned direction, effectively turning them into unwitting accomplices, significantly enhancing the backdoor persistence. Additionally,* `BadSFL` *leverages a GAN-enhanced poisoning strategy to enrich the attacker's dataset, maintaining high accuracy on both benign and backdoored samples while remaining stealthy. Extensive experiments demonstrate that* `BadSFL` *achieves superior attack durability, maintaining effectiveness for over 60 global rounds—lasting up to three times longer than existing baselines even after ceasing malicious model injections.*

1. Introduction

Federated Learning (FL) enables distributed model training while preserving client data privacy. However, the effectiveness of FL models heavily depends on the distribution of training data across clients. Two scenarios typically arise: 1) IID data, where training data is uniformly distributed across clients, and 2) non-IID data, a more realistic setting where data characteristics vary significantly across clients. For IID scenarios, FedAvg [27] stands out as the leading FL method, setting the standard for server-side model updates by aggregating model parameters from clients. However, its performance deteriorates in non-IID scenarios, where data heterogeneity causes update drifts from individual clients, ultimately degrading convergence [22].

To address this challenge, Scaffold [12] was introduced as a robust FL method designed to mitigate client update drift through a correction mechanism based on control variates, thereby enhancing model convergence in non-IID settings. The control variate is essentially an estimate of the difference between a client's local gradient and the global gradient, which helps align the local updates with the global objective. Scaffold reduces variance in the updates caused by data heterogeneity, making it particularly effective for scenarios where clients have diverse data distributions.

However, Scaffold Federated Learning (SFL) not only changes the way FL models converge but also affects their robustness against adversarial manipulations. In particular, malicious clients in FL can exploit model update mechanisms to introduce backdoor behaviors, embedding hidden misbehavior into the global model [6]. While backdoor attacks have been extensively studied in FL [3, 7, 34, 36, 40], most existing works focus on IID scenarios where attackers have full knowledge of the dataset distribution and can easily craft poisoned updates. In contrast, non-IID data distributions introduce additional constraints, making it harder for attackers to align poisoned models with the global model without significantly degrading overall performance. Although recent studies have explored backdoor attacks in non-IID FL [2, 10, 29, 42], they have largely overlooked the unique security implications introduced by SFL. The question this paper aims to address is: *if the new mechanisms of SFL (i.e., control variate for update drift correction) can bring new security threats, and unintentionally facilitate backdoor attacks in non-IID settings?*

Our answer to the above question is affirmative. Our new insight is that **Scaffold's reliance on control variates in-**

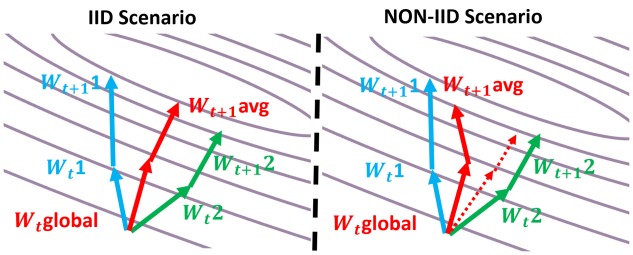


Figure 1. Model averaging under IID and non-IID scenarios.

troduces a novel attack surface: its correction mechanism, designed to stabilize training by aligning local updates with the global objective, can inadvertently amplify the impact of malicious updates. More critically, this mechanism allows an attacker to influence the control variate itself, effectively co-opting benign clients to "*assist in the mischief*". Since all clients use the control variate to adjust their local gradients during updates, a tampered variate can subtly steer these honest clients' gradients toward the attacker's poisoned direction. This amplifies the backdoor's reach, making Scaffold more susceptible to sophisticated attacks than standard FL methods like FedAvg, which lack such a correction mechanism.

To exploit the above insight, we propose BadSFL, a novel backdoor attack specifically targeting Scaffold Federated Learning, to successfully implant a backdoor function into a global model without catastrophically corrupting model performance on benign sample inference. Unlike prior attacks, BadSFL leverages Scaffold's correction dynamics to enhance both the stealth and durability of the backdoor, revealing a critical vulnerability in SFL methods. BadSFL operates as follows: Firstly, as the attacker only has partial knowledge of the dataset distribution in the FL system, he leverages a GAN to generate fake samples that belong to other clients to supplement the dataset, simulating a full knowledge of the dataset distribution. With the supplemented dataset for backdoor training, he gets a backdoor model achieving high accuracy in both backdoor tasks and benign tasks. Secondly, the attacker uses a distinctive feature of a category as the backdoor trigger to maintain the attack stealthiness. Thirdly, the attacker exploits the global control variate, as a reference to predict the global model's convergence direction. This optimization significantly enhances the durability of the embedded backdoor function within the global model.

We evaluate BadSFL on the MNIST, CIFAR-10, and CIFAR-100 datasets, demonstrating its high accuracy on both backdoor and primary tasks. Moreover, the embedded backdoor function persists in the global model for over 60 rounds and lasts 3 times longer than baseline attacks after the attacker stops injecting malicious updates. Finally, we show that BadSFL remains highly effective when using four defense methods simultaneously.

The main contributions are as follows:

- We propose BadSFL, the *first* backdoor attack against SFL on non-IID scenarios.
- We enhance the backdoor durability, ensuring it persists for over 60 rounds and lasts 3× longer than baselines.
- We conduct extensive experiments on three benchmark datasets, demonstrating high effectiveness of our attack.

2. Background and Related Work

2.1. Non-IID Scenarios in FL

In FL, non-IID refers to significant differences in data distributions among clients [11, 20, 28]. This discrepancy between local data distributions in non-IID scenarios can lead to inconsistencies between the local optima and the global optima. This inconsistency results in a drift in local model updates, where local models move towards their own local optima which can be far from the global optima [12]. Consequently, averaging these local models may yield a global model far from the true global optima [12, 16, 21, 37, 38], especially with numerous local epochs. As shown in Figure 1, while the global optima aligns with the local optima in IID scenarios, non-IID can cause the global optima to be distant from individual local optima, which is known as the *client-drift* phenomenon, leading to slow and unstable convergence in the FL training process.

Scaffold. Several FL algorithms have been proposed to address the above challenges, with Scaffold [12] being the most practical solution. It tackles the client-drift problem through the control variates (variance reduction techniques) for both the server and the clients. These control variates estimate the update direction of the global model and local client models and serve to correct local updates based on the drift, thereby mitigating the divergence between local and global optima (Alg. 1). In this paper, we mainly focus on designing backdoor attacks targeting SFL.

2.2. Backdoor Attacks against FL

Backdoor attacks pose a significant threat to deep learning models, where malicious clients embed hidden triggers within models that cause misclassification during inference while maintaining normal performance on clean data [23]. In FL, adversaries can deploy backdoor attacks by exploiting compromised clients to manipulate local updates, thereby generating poisoned models that corrupt the global model upon aggregation [3, 19, 34, 36, 40, 45].

Backdoor attacks typically involve a combination of model replacement and data poisoning. In model replacement, the adversary substitutes legitimate models with manipulated ones [3], while data poisoning involves injecting poisoned data containing the backdoor trigger into the training datasets of compromised clients. To inject these triggers, the adversary can manipulate the dataset by flipping

Algorithm 1 Scaffold Algorithm in Federated Learning

Sever Input: local datasets D^i, number of client K, number of communication rounds R, number of local epochs E
Client Input: local control variates c_i, local step-size η_l
Server Updates:
  $c^t \leftarrow 0$;
  for each round r = 1, ..., R **do**:
    randomly selected clients $S^t \subseteq \{1, \ldots, K\}$
    for $i \in S^t$ **in parallel do**:
      send $w^t, c^t \rightarrow i$
      $\Delta w_i^t, \Delta c_i^t \leftarrow$ **Local Update**(i, w^t, c^t)
    $w^t + 1 \leftarrow w^t - \eta \sum_{i \in S^t} \Delta w_i^t$
    $c^t + 1 \leftarrow c^t + \frac{1}{K} \Delta c$
  end for

Local Updates:
  Local client i get w^t, c^t from server
  training model with D_i get **gradient** $g_i(w_i)$
  update local model:
    $w_i \leftarrow w_i - \eta_l * (g_i(w_i) - c_i + c)$
  update local control variates:
    $c_i^{t+1} \leftarrow (i) \quad g_i(w_i), or \quad (ii) \quad c_i - c + \frac{1}{K * \eta_l} * (w^t - w_i)$
  $\Delta w_i^t \leftarrow w_i - w^t$
  $\Delta c_i^t \leftarrow c_i^{t+1} - c_i$
  return $(\Delta w_i^t, \Delta c_i^t)$

data labels or adding a unique pixel pattern to the training samples [1, 3, 36]. Afterward, he strategically adjusts training parameters and scales updates to optimize the impact of the attack while evading detection by the anomaly detector deployed at the central aggregation server [5, 30].

In this paper, we focus on data poisoning-based backdoor attacks. While most existing backdoor attacks target FL under IID scenarios [9, 25, 31, 40, 43], real-world FL deployments often involve non-IID distributions, posing additional challenges for effective backdoor injection. To the best of our knowledge, no prior research has specifically explored backdoor attacks against FL with the Scaffold aggregation algorithm. We bridge this gap by investigating a novel backdoor attack targeting SFL, leveraging its unique control variate mechanism to enhance the effectiveness, stealthiness, and persistence of the attack.

2.3. Threat Model

Attack scenarios. We consider an attacker who aims to inject backdoors into SFL, make the final model predict the desired wrong output over a triggered input. The attacker has partial knowledge of the full dataset in the training stage. Specifically, the attacker trains a local model with a backdoor trigger function and submits poisoned local updates along with the control variate to the server for Scaffold aggregation. During inference, the attacker manipulates predictions to produce the attacker's desired outputs when inputs meet the trigger conditions.

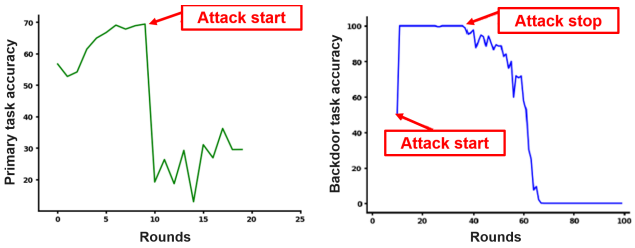


Figure 2. *Left*: Primary task accuracy crushed by simple attack; *Right*: Backdoor accuracy dropped after stop attacking.

Attack goal. The attack goal can be summarized as follows:

- *Effectiveness*: The attacker must ensure that the backdoor function does not compromise the global model's performance on primary tasks, maintaining high accuracy for both backdoor and benign predictions.
- *Robustness*: the backdoor should be robust against potential defenses.
- *Durability*: The backdoor should remain effective in the global model for as long as possible, even after the attacker stops participation in the training process, thereby maximizing the longevity of the attack.

Attacker's capability. We assume the attacker can compromise at least one client during the training process, thereby allowing him to operate covertly within the system. Additionally, participation in FL provides the attacker with full knowledge of the model structure, facilitated by the consensus among all clients on a common learning objective. This enables the insertion of backdoor triggers, the modification of sample labels, and the manipulation of local training updates. It is essential to note that the attacker cannot control the server or directly manipulate the aggregation procedure or the global model. They also lack access to data and models from non-compromised clients.

3. Challenges with Backdoor Attacks in SFL

Performing a backdoor attack in SFL presents the following challenges. ① **Limited knowledge.** In non-IID scenarios, a primary challenge arises from the attacker's lack of knowledge of the dataset distribution across clients. Unlike IID scenarios, where a centralized understanding of the dataset facilitates manipulation, non-IID scenarios involve decentralized and diverse data distributions. This results in three issues: (1) Direct backdoor strategies can cause significant performance degradation on benign samples, leading to the rejection of the global model; (2) The variability in data distributions increases the difference between local and global models, making malicious models more detectable; (3) Av-

eraging poisoned models with the global model degrades its performance on the primary task, as shown in Figure 2, where accuracy drops significantly when a poisoned model is aggregated. ② **Control variate.** In SFL, control variate (denoted as c_i) is used to correct the client drift and align local models with the global model. If attackers strictly follow protocols and use the c_i) to correct their malicious models during the triggering planting process, the effectiveness of the attack can be reduced. Conversely, if an attacker chooses to manipulate the c_i inappropriately, introducing a malicious c to the server, it could lead to a potential corruption of the global model. ③ **Backdoor catastrophic forgetting.** Catastrophic forgetting [13] occurs when neural networks forget previously learned tasks upon learning new ones. This can cause backdoors to lose effectiveness over time [36]. If attackers stop uploading malicious updates, the backdoor function may eventually be erased by benign updates. As shown in Figure 2, the accuracy of backdoor tasks declines sharply over time, with the backdoor function vanishing around round 65. Although various methods [2, 7, 39, 45] have been proposed to address this issue, none have proven effective in SFL.

Algorithm 2 `BadSFL`

Required: local datasets D^i, global model w_g, global control variate c, number of local epochs E, local learning rate η_l, Generator G, Discriminator D

Update local model with $w_p \leftarrow w_g$
Initialize Discriminator $D \leftarrow w_g$
do:
  Run G for generating fake samples
  Evaluate fake sample on D
  Update G using D
until G converges to generate target samples
G generates samples into dataset D_f
$D_c \leftarrow D_f + D^i$
Select backdoor samples from D_c and assign them wrong label as D_b
$D_p \leftarrow D_c + D_b$
for each epoch e = 1, ..., E **do**:
  $w_p = argmin_{w_p}[L(D_p, w_p) + L(D_p, P_j(w_p, c))]$.
end for
$\Delta w_p = w_p - w_g$
$\Delta c_p = \frac{1}{K*\eta_l} * (w_g - w_p) - c$
return $(\Delta w_p, \Delta c_p)$

4. `BadSFL`

Overview. To overcome the challenges, we propose `BadSFL`, as detailed in Algorithm 2. `BadSFL` mainly consists of 4 steps: *Step 1: Initialization.* The attacker initiates the attack by downloading the global model w_g and controlling the variate c from the server. Subsequently, the attacker updates the local model w_p and discriminator D using the downloaded global model w_g. *Step 2: GAN-based Training for Data Supplementation.* The attacker performs GAN training on generator G and discriminator D. The training terminates upon the convergence of the generator, signifying its ability to generate realistic fake samples in class C that do not belong to D^i but rather originate from other clients' datasets. Then the generator G is utilized to generate a number of samples in class c forming in dataset D_f. This dataset D_f is then merged with the attacker's original dataset D^i to create a new dataset D_c. *Step 3: Backdoor Sample Selection and Trigger Injection.* With dataset D_c, the attacker selects specific samples with a characteristic feature to serve as backdoor samples. These samples are then relabeled to a target class x as the backdoor target class that is different from their original labels. The attacker organizes these manipulated samples into a separate dataset D_b and merges it with D_c to finalize the dataset D_p for backdoor training. *Step 4: Backdoor Model Training and Optimization.* The attacker proceed to trains the local model w_p based on the dataset D_p. During the training process, the attacker follows the equation 3 to optimize the backdoor objective. Upon convergence, the backdoor model update Δw_p and the corresponding control variate Δc_p are obtained and can be uploaded to the server.

4.1. GAN-based Dataset Supplementation

In non-IID data scenarios, directly injecting backdoor samples into dataset D^i for training often leads to a more biased model, deviating significantly from global optima [15, 33]. To mitigate this, inspired by Zhang et al. [44], the attacker can employ GAN to generate synthetic samples that resemble the data held by other clients. This involves training a generator G with local non-IID data to bridge the gap between datasets. The GAN architecture basically consists of a generator G and a discriminator D. In our case, the generator G comprises a series of 'deconvolution' layers that progressively transform random noise into a sample, while the discriminator D closely resembles the global model, except for its output layer, which distinguishes between fake and real samples. The attacker iteratively trains the generator G locally with the constraint of discriminator D until it converges to generate realistic fake samples that do not belong to the attacker. Concurrently, as the SFL procedure progresses, the global model tends to converge. During each server-client communication round, the attacker updates the discriminator D using the latest global model w_g downloaded from the server and performs new-round optimization training on the generator G, guiding it to generate more authentic fake samples that closely resemble data from other clients. These high-quality synthetic samples are then integrated into the attacker's original non-IID dataset, effectively supplementing it with additional data classes.

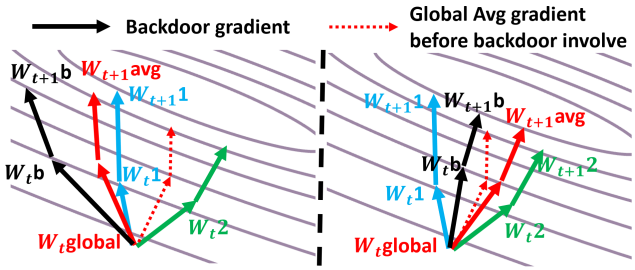


Figure 3. *Left*: global optima under non-IID scenario. *Right*: global optima with data supplementation techniques.

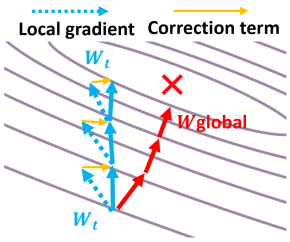


Figure 4. Scaffold correction term on a single client.

The attacker synchronously updates the discriminator D during each local training round using the new global model w_g downloaded from the server, followed by GAN training to optimize the generator G for improved performance. The output of this process is then merged into the attacker's non-IID dataset for further backdoor training. As the generated samples closely resemble those from other client datasets, the attacker local optima trained by the attacker can converge closer to the global optima than other clients. Figure 3 demonstrates the difference between the aggregated global optima with and without data supplementation techniques.

4.2. Trigger Selection and Injection

With the prepared dataset D_p, containing both original and synthetic data, the attacker proceeds to inject a backdoor into the model. `BadSFL` leverages three techniques to inject backdoors: (1) Label-flipping [17], in which the ground truth labels of a whole class a in D_p are directly altered to another label b. For instance, all the 'dog' labels are altered to 'cat' in CIFAR-10. (2) Pattern trigger [35], which involves poisoning samples with a specific trigger pattern, i.e., a small mosaic cube added in the images to activate the backdoor behavior. The attacker injects these poisoned images into the D_p along with a target label, establishing a correlation between the trigger pattern and the desired misclassification. (3) A stealthier backdoor method, known as feature-based backdoor [26], involves selecting a distinctive feature within an image class as the backdoor trigger. This approach eliminates the need to directly manipulate the images, thereby increasing the difficulty of detection. For instance, all the green cars in the 'car' class in CIFAR-10 are designed as the backdoor trigger. During the inference stage, the compromised model outputs the attacker's target label only when the input is an image containing a green car. The selection of a unique feature within a class makes this trigger difficult to detect as it appears as a natural variation within the data.

4.3. Backdoor Training with Control Variate

As discussed in section 3, the global control variate c is utilized in SFL to correct the client drift. Specifically, the correction value $c - c_i$ adjusts the local model point towards the global model, as shown in algorithm 1. During the local model training process, this correction term effectively 'drag' the drifting local model closer to the global model, facilitating convergence towards the global optima, as depicted in Figure 4. In the server aggregation round, the global control variate c is computed by averaging the drift values of all local models, which represents the convergence direction of the global model.

From the attacker's perspective, allowing the control variate to correct the poisoned model according to SFL rules can reduce the effectiveness of the backdoor attack, as discussed in Section 3. However, the attacker still needs to submit a control variate c_p to report the drift of the backdoor model. The key idea is to train a backdoor model that is closer to the global model compared to other local models trained on non-IID data. Since the global control variate c is known to participating clients, it can be used as a reference for the global model's convergence direction, helping to align the poisoned model more closely with the global optimum. This constraint, derived from c, functions similarly to using the future global model for optimization, as suggested by Wen et al.[39]. This constraint can be integrated into the loss function to enhance the backdoor's effectiveness and persistence in the global model.

Initially, the attacker performs backdoor training and optimizes their backdoor objective as in the Equation 1 [4]:

$$w_p^* = \underset{w_p}{argmin}\, L(D_p, w_p). \tag{1}$$

where L is the loss function of the backdoor task, w_p is the attacker model weights.

In our `BadSFL` attack, we modify the standard backdoor objective function by adding a term to ensure that the backdoor updates sent to the server persist in the backdoor function in the global model for more future training rounds. We can simulate an aggregation round and apply the control variate c to obtain a predicted global model for one future round. Here is the modified objective function (Equation 2):

$$P_j(w_p, c) = \frac{w_p + w_g * (n-1)}{n} - \eta_l * c * j \tag{2}$$

To summarize, we formalize our attack objective as below:

$$w_p^* = \underset{w_p}{argmin}[L(D_p, w_p) + L(D_p, P_j(w_p, c))]. \tag{3}$$

Table 1. Dataset, model structure, and hyperparameter description.

Dataset	Instances	Features	Model	Benign l_r	E	Poison l_r	Poison ratio			Batch Size
							FL	PT	FB	
CIFAR-10	60000	1024	ResNet-18	0.001	10	0.05	0.1	0.0125	0.01	128
CIFAR-100	60000	1024	ResNet-50	0.0001	100	0.0001	0.1	0.0125	0.01	128
MNIST	70000	784	ConvNet	0.01	2	0.001	0.1	0.0125	-	128

where j represents the number of future rounds that w_p anticipates. By optimizing the backdoor model to be closer to the global model, the attacker simultaneously optimizes the control variate c_p to align it with the expected drift value. This ensures that the attacker's actions conform to the SFL protocol (Algorithm 1).

5. Evaluation

5.1. Setup

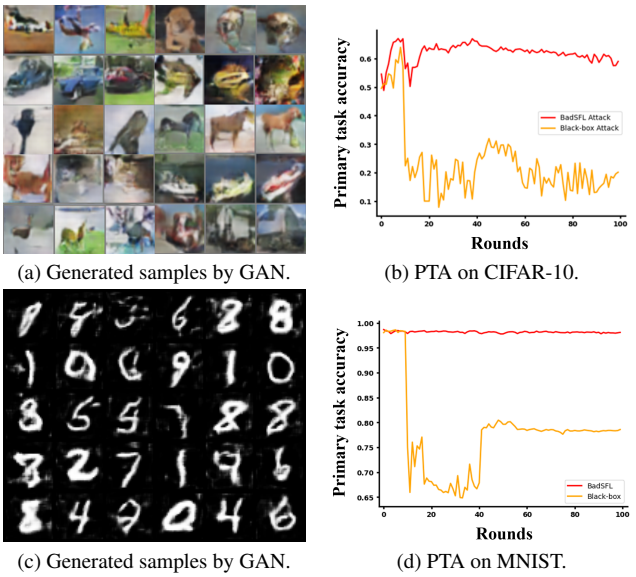
We evaluate `BadSFL` on a server running Ubuntu 18.04 with an NVIDIA GeForce RTX 2080 Ti GPU.

Datasets, models, and hyperparameters. We consider the datasets that are commonly used in previous works [25], i.e., MNIST [8], CIFAR-10 [14] and CIFAR-100 [14]. Detailed configurations are summarized in Table 1.

GAN models. In attacking SFL with CIFAR-10 and CIFAR-100, the discriminator mimics ResNet18 and ResNet50, respectively, differing only in the output layer. The generator consists of five deconvolutional layers to generate synthetic images for supplementing D_p in the backdoor attack. For MNIST, the discriminator follows LeNet-5, while the generator uses three deconvolutional layers to produce synthetic images resembling those of other clients.

Attack bsaselines. We compare `BadSFL` with 4 backdoor attacks: (1) *Black-box Attack* [3]. This attack directly poisoned the training dataset through techniques such as label-flipping, pattern trigger, and feature-based trigger. Subsequently, it conducts backdoor training to minimize the classification loss on the dataset D_p. (2) *Neurotoxin* [45]. It incorporates a strategic approach to ensure the durability of the backdoor function within the global model. Specifically, it aims to prevent the malicious model updates from pointing toward coordinates that are frequently updated by benign clients, thereby mitigating the risk of the backdoor being erased. We illustrate it in Figure 9 in the Appendix. (3) Irreversible Backdoor Attack (IBA) [31]. IBA gradually implants a stealthy and durable backdoor into the global model by optimizing trigger imperceptibility and selectively poisoning parameters less likely to be updated. (4) 3DFed [18]. 3DFed is a multi-layered backdoor attack framework that combines constrained loss training, noise masking, and a decoy model to evade detection in a black-box FL setting.

Attack settings. We run 100 communication rounds in SFL for CIFAR-10, CIFAR-100, and MNIST. In each round, 20 clients participate, with 50% randomly of them selected for training. To ensure a non-IID data distribution, the training dataset is split into 200 label-sorted groups, which are randomly assigned to all the participating clients.



(a) Generated samples by GAN. (b) PTA on CIFAR-10.

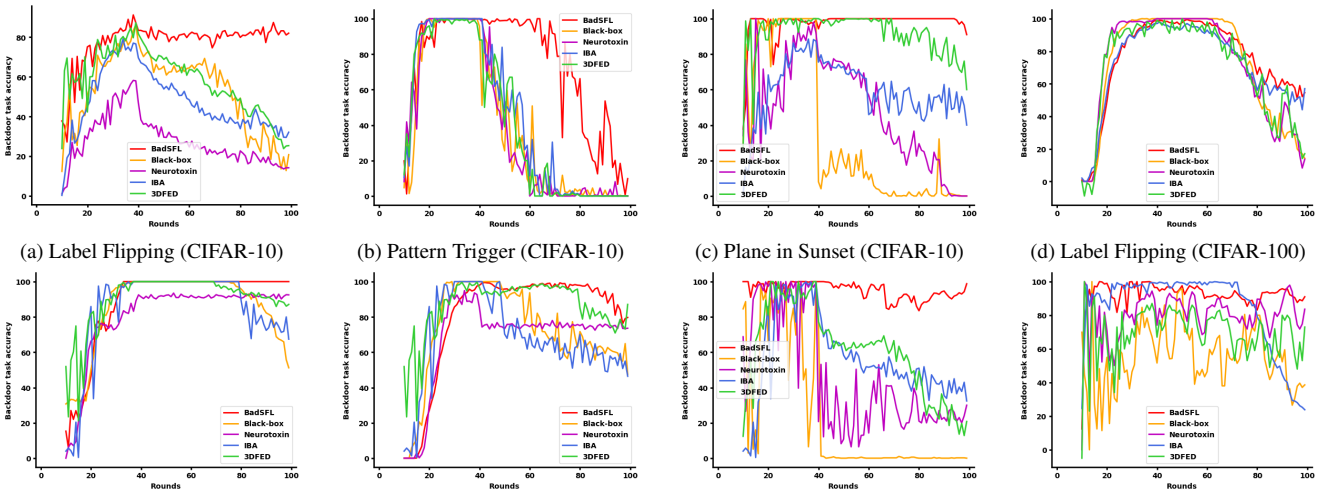
(c) Generated samples by GAN. (d) PTA on MNIST.

Figure 5. Dataset supplementation on CIFAR-10 and MNIST.

For `BadSFL` evaluations on CIFAR-10 and CIFAR-100 datasets, the attacker strategically joins the training process from the 10th round, exiting after the 40th round. Initially, the attacker conducts GAN training to perform data supplementation, employing 10 local epochs, a fixed learning rate of 0.001, and Adam Optimizer with parameters (0.5, 0.999). Subsequently, backdoor training and optimization are carried out with 10 local epochs, a future round j set to 10, a fixed learning rate of 0.05, and an SGD optimizer with a momentum of 0.9 and a weight decay of 0.005.

To obtain D_p, we employ three types of backdoor trigger injections and evaluate the performance of `BadSFL` based on BTA in each communication round, contrasting it with baseline attacks: (1) Label Flipping. All 'dog' samples in D_c are relabeled as 'bird'; (2) Pattern Trigger. A small triangle pattern is added to the right bottom of the image, with all the modified samples labeled as 'cat'; (3) Feature-based trigger. Characteristic features (e.g., car stripe, green car, race car, sunset plane, white horse, red ship, yellow truck) are chosen as backdoor triggers in Dataset D_c, with corresponding samples labeled as 'bird'.

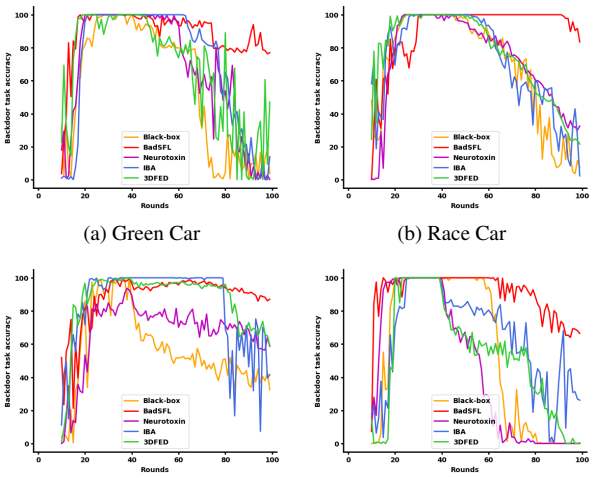
For MNIST, the attacker engages in the training process from the 10th round and quits after the 40th round.



(a) Label Flipping (CIFAR-10) (b) Pattern Trigger (CIFAR-10) (c) Plane in Sunset (CIFAR-10) (d) Label Flipping (CIFAR-100)

(e) Plane in Sunset (CIFAR-100) (f) Yellow Truck (CIFAR-100) (g) Label Flipping (MNIST) (h) Pattern Trigger (MNIST)

Figure 6. Attack comparisons with baselines.



(a) Green Car (b) Race Car

(c) White Horse (d) Red Ship

Figure 7. Feature-based backdoor attacks on CIFAR-10.

Also, GAN training is employed for data supplementation, comprising 2 local epochs, a fixed learning rate of 0.001, and Adam Optimizer with parameters (0.5, 0.999). Subsequently, backdoor training and optimization are performed over 2 local epochs, considering future rounds j as 10, with a fixed learning rate of 0.1 and SGD optimizer.

To obtain D_p, only two types of backdoor trigger injections are used, as MNIST lacks sufficient features for feature-based backdoor attacks. The performance of `BadSFL` is evaluated based on the BTA in each communication round, compared to two baselines: (1) Label flipping. All samples labeled as '5' in D_c are relabeled as '2'; (2) Pattern trigger. A small triangle pattern is added to the right bottom of the image, and the labels of all modified samples are changed to '2'.

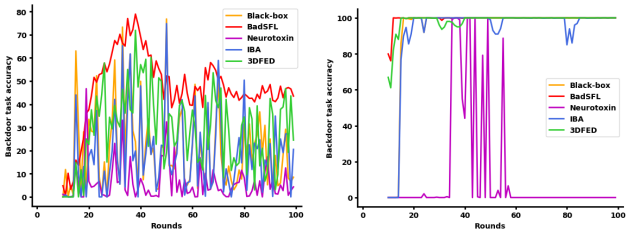
Evaluation metric. We adopt two metrics commonly utilized in FL. (1) Primary Task Accuracy (PTA), which reflects the classification accuracy on clean samples. (2) Backdoor Task Accuracy (BTA), which is the attack success rate, measured by the poisoned model's accuracy on poisoned samples.

5.2. Results of GAN

Figure 5a and Figure 5c illustrate the datasets D_f generated by the Generator G for CIFAR-10 and MNIST, respectively. In both cases, D_f encompasses nearly all classes of each dataset, demonstrating the successful acquisition of full dataset distribution knowledge within the SFL framework under non-IID scenarios. Utilizing the combined dataset D_c, the attacker executes the backdoor attack while maintaining the model's accuracy on the primary task. For CIFAR-10, as shown in Figure 5b, the primary task accuracy remains around 55% with data supplementation, compared to a drop below 25% in the baseline attack. Similarly, for MNIST, Figure 5d reveals that using D_c instead of D_i, the primary task accuracy stays above 90%, while the baseline attack results in a decline to under 75%.

5.3. Effectiveness of `BadSFL`

From Figures 6a to 6f, we present attack comparisons with baselines evaluated on the CIFAR-10 and CIFAR-100 datasets. It is evident that `BadSFL` outperforms the baseline attacks in terms of both effectiveness and durability. To be specific, `BadSFL` achieves above 80% backdoor task accuracy across all types of backdoor attacks within the first 10 rounds, while the attacker remains active in the training process for backdoor training and malicious updates to the server. Meanwhile, `BadSFL` keeps the primary task accuracy at 60% (Figure 5b). Furthermore, even after the attacker exits the training process at the 40th round, the be-



(a) Label Flipping, CIFAR-10, differential privacy + model pruning.

(b) Pattern Trigger, CIFAR-10, differential privacy + model pruning.

(c) Feature-based Trigger, CIFAR-10, differential privacy + model pruning.

(d) Feature-based Trigger, CIFAR-100, differential privacy + model pruning.

(e) Feature-based Trigger, CIFAR-100, FLAME.

(f) Feature-based Trigger, CIFAR-100, SparseFED.

Figure 8. Defense against backdoor attacks.

nign clients continue submitting normal updates in subsequent rounds, which could potentially affect the poisoned updates from the attacker in previous attacking rounds thus erasing the backdoor function. Despite this, BadSFL ensures a resilient backdoor function with accuracy exceeding 90% over the entire 100 SFL rounds, which is 3 times longer than the lifespan achieved by the two baseline attacks, where backdoor task accuracy drops below 50% after the 60th round. Horizontally comparing the effects of different types of backdoor trigger injections (Figures 6a, 6b and 6c), it is observed the feature-based trigger performs the best among them, benefiting from its stealthiness without directly manipulating the images, thereby making its updates less likely to conflict with those from benign updates. We also provide more results available in Figure 7.

Figures 6g and 6h showcase the results obtained on the MNIST dataset. Similarly, BadSFL outperforms the other baseline attacks, achieving both backdoor task accuracy and primary task accuracy above 85%. After the malicious update injection stops at round 40, in the Label flipping attack, the backdoor task accuracy in two baseline attacks catastrophically drops below 40% within 10 rounds whereas

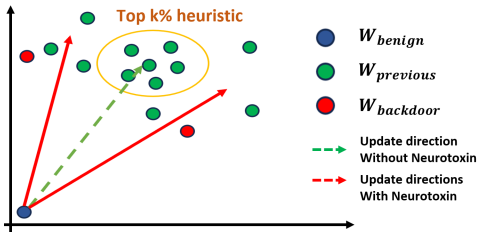


Figure 9. Neurotoxin.

BadSFL maintains a 5 times longer-lasting backdoor function in the global model in future rounds. In the pattern trigger attack, BadSFL also injects a more effective backdoor function into the global model with 10% higher accuracy compared to the baselines.

Defense. We further conduct multiple defense strategies to demonstrate the superiority of BadSFL. Specifically, we use differential privacy [41], model pruning [24], FLAME [30] and SparseFed [32] to resist backdoor attacks. Figure 8 shows that BadSFL can still maintain high effectiveness under multiple defense strategies. Taking the label flipping on CIFAR-10 as an example, it can be observed that our method persistently maintains a high attack success rate even under defense strategies, while other baselines fail.

Analysis on Neurotoxin. Neurotoxin aims to address the durability issue of backdoor attacks in FL settings, which ensures that the backdoor persists in the global model even after the attacker stops uploading poisoned updates. During FL training, Neurotoxin leverages the concept of the L2 norm, a mathematical function that represents the magnitude of a vector. Neurotoxin observes that the majority of the L2 norm of the aggregated benign gradient is contained in a small number of coordinates, which implies that benign updates tend to cluster in a narrow range and consequently, the aggregated gradient direction is likely to point towards this cluster. Therefore, Neurotoxin identifies and targets the parameters that get minimal changes in magnitude during training, as illustrated in Figure 9 in the Appendix. These relatively stable parameters are less likely to be significantly affected by benign updates, thereby mitigating the risk of the backdoor being overwritten by future updates.

However, our results do not confirm Neurotoxin's expected effectiveness. Despite using this strategy, backdoor accuracy declines similarly to the baseline attack after the attacker exits at round 50. This trend is also observed in BadSFL experiments, warranting further investigation.

6. Conclusion

This paper introduces BadSFL, a novel backdoor attack specifically tailored for non-IID federated learning environments utilizing the Scaffold aggregation algorithm. By employing a GAN-based data augmentation technique and exploiting the Scaffold's control variate, BadSFL achieves superior effectiveness, stealthiness, and durability compared

to existing methods. Our experimental results on multiple benchmark datasets demonstrate the attack's effectiveness, with the backdoor persisting significantly longer than existing approaches. In the future, we hope researchers can design more robust defense mechanisms to safeguard federated learning systems against such attacks.

References

[1] Manaar Alam, Hithem Lamri, and Michail Maniatakos. Get rid of your trail: Remotely erasing backdoors in federated learning. *arXiv preprint arXiv:2304.10638*, 2023. 3

[2] Manaar Alam, Esha Sarkar, and Michail Maniatakos. Perdoor: Persistent backdoors in federated learning using adversarial perturbations. In *2023 IEEE International Conference on Omni-layer Intelligent Systems (COINS)*, pages 1–6. IEEE, 2023. 1, 4

[3] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, pages 2938–2948. PMLR, 2020. 1, 2, 3, 6

[4] Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. Analyzing federated learning through an adversarial lens. In *International Conference on Machine Learning*, pages 634–643. PMLR, 2019. 5

[5] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30, 2017. 3

[6] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017. 1

[7] Yanbo Dai and Songze Li. Chameleon: Adapting to peer images for planting durable backdoors in federated learning. In *International Conference on Machine Learning*, pages 6712–6725. PMLR, 2023. 1, 4

[8] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012. 6

[9] Pei Fang and Jinghui Chen. On the vulnerability of backdoor defenses for federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11800–11808, 2023. 3

[10] Hyejun Jeong, Joonyong Hwang, and Tai Myung Chung. Abc-fl: anomalous and benign client classification in federated learning. *arXiv preprint arXiv:2108.04551*, 2021. 1

[11] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021. 2

[12] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020. 1, 2

[13] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 4

[14] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[15] Jianxiong Lai, Xiuli Huang, Xianzhou Gao, Chang Xia, Jingyu Hua, et al. Gan-based information leakage attack detection in federated learning. *Security and Communication Networks*, 2022, 2022. 4

[16] Gihun Lee, Minchan Jeong, Yongjin Shin, Sangmin Bae, and Se-Young Yun. Preservation of the global knowledge by not-true distillation in federated learning. *Advances in Neural Information Processing Systems*, 35:38461–38474, 2022. 2

[17] Dongcheng Li, W Eric Wong, Wei Wang, Yao Yao, and Matthew Chau. Detection and mitigation of label-flipping attacks in federated learning systems with kpca and k-means. In *2021 8th International Conference on Dependable Systems and Their Applications (DSA)*, pages 551–559. IEEE, 2021. 5

[18] Haoyang Li, Qingqing Ye, Haibo Hu, Jin Li, Leixia Wang, Chengfang Fang, and Jie Shi. 3dfed: Adaptive and extensible framework for covert backdoor attack in federated learning. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 1893–1907. IEEE, 2023. 6

[19] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10713–10722, 2021. 2

[20] Qinbin Li, Zeyi Wen, Zhaomin Wu, Sixu Hu, Naibo Wang, Yuan Li, Xu Liu, and Bingsheng He. A survey on federated learning systems: Vision, hype and reality for data privacy and protection. *IEEE Transactions on Knowledge and Data Engineering*, 2021. 2

[21] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020. 2

[22] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019. 1

[23] Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022. 2

[24] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*, pages 273–294. Springer, 2018. 8

[25] Xiaoting Lyu, Yufei Han, Wei Wang, Jingkai Liu, Bin Wang, Jiqiang Liu, and Xiangliang Zhang. Poisoning with cerberus: Stealthy and colluded backdoor attack against fed-

erated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9020–9028, 2023. 3, 6

[26] Robin Mayerhofer and Rudolf Mayer. Poisoning attacks against feature-based image classification. In *Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy*, pages 358–360, 2022. 5

[27] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 1

[28] Matias Mendieta, Taojiannan Yang, Pu Wang, Minwoo Lee, Zhengming Ding, and Chen Chen. Local learning matters: Rethinking data heterogeneity in federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8397–8406, 2022. 2

[29] Mohammad Naseri, Yufei Han, and Emiliano De Cristofaro. Badvfl: Backdoor attacks in vertical federated learning. *arXiv preprint arXiv:2304.08847*, 2023. 1

[30] Thien Duc Nguyen, Phillip Rieger, Roberta De Viti, Huili Chen, Björn B Brandenburg, Hossein Yalame, Helen Möllering, Hossein Fereidooni, Samuel Marchal, Markus Miettinen, et al. {FLAME}: Taming backdoors in federated learning. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1415–1432, 2022. 3, 8

[31] Thuy Dung Nguyen, Tuan A Nguyen, Anh Tran, Khoa D Doan, and Kok-Seng Wong. Iba: Towards irreversible backdoor attacks in federated learning. *Advances in Neural Information Processing Systems*, 36:66364–66376, 2023. 3, 6

[32] Ashwinee Panda, Saeed Mahloujifar, Arjun Nitin Bhagoji, Supriyo Chakraborty, and Prateek Mittal. Sparsefed: Mitigating model poisoning attacks in federated learning with sparsification. In *International Conference on Artificial Intelligence and Statistics*, pages 7587–7624. PMLR, 2022. 8

[33] Konstantinos Psychogyios, Terpsichori-Helen Velivassaki, Stavroula Bourou, Artemis Voulkidis, Dimitrios Skias, and Theodore Zahariadis. Gan-driven data poisoning attacks and their mitigation in federated learning systems. *Electronics*, 12(8):1805, 2023. 4

[34] Ziteng Sun, Peter Kairouz, Ananda Theertha Suresh, and H Brendan McMahan. Can you really backdoor federated learning? *arXiv preprint arXiv:1911.07963*, 2019. 1, 2

[35] Aashma Uprety and Danda B Rawat. Mitigating poisoning attack in federated learning. In *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 01–07. IEEE, 2021. 5

[36] Hongyi Wang, Kartik Sreenivasan, Shashank Rajput, Harit Vishwakarma, Saurabh Agarwal, Jy-yong Sohn, Kangwook Lee, and Dimitris Papailiopoulos. Attack of the tails: Yes, you really can backdoor federated learning. *Advances in Neural Information Processing Systems*, 33:16070–16084, 2020. 1, 2, 3, 4

[37] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. *arXiv preprint arXiv:2002.06440*, 2020. 2

[38] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020. 2

[39] Yuxin Wen, Jonas Geiping, Liam Fowl, Hossein Souri, Rama Chellappa, Micah Goldblum, and Tom Goldstein. Thinking two moves ahead: Anticipating other users improves backdoor attacks in federated learning. *arXiv preprint arXiv:2210.09305*, 2022. 4, 5

[40] Chulin Xie, Keli Huang, Pin-Yu Chen, and Bo Li. Dba: Distributed backdoor attacks against federated learning. In *International conference on learning representations*, 2019. 1, 2, 3

[41] Chulin Xie, Minghao Chen, Pin-Yu Chen, and Bo Li. Crfl: Certifiably robust federated learning against backdoor attacks. In *International Conference on Machine Learning*, pages 11372–11382. PMLR, 2021. 8

[42] Tiandi Ye, Cen Chen, Yinggui Wang, Xiang Li, and Ming Gao. Bapfl: You can backdoor personalized federated learning. *ACM Transactions on Knowledge Discovery from Data*, 2024. 1

[43] Hangfan Zhang, Jinyuan Jia, Jinghui Chen, Lu Lin, and Dinghao Wu. A3fl: Adversarially adaptive backdoor attacks to federated learning. *Advances in neural information processing systems*, 36:61213–61233, 2023. 3

[44] Jiale Zhang, Junjun Chen, Di Wu, Bing Chen, and Shui Yu. Poisoning attack in federated learning using generative adversarial nets. In *2019 18th IEEE international conference on trust, security and privacy in computing and communications/13th IEEE international conference on big data science and engineering (TrustCom/BigDataSE)*, pages 374–380. IEEE, 2019. 4

[45] Zhengming Zhang, Ashwinee Panda, Linyue Song, Yaoqing Yang, Michael Mahoney, Prateek Mittal, Ramchandran Kannan, and Joseph Gonzalez. Neurotoxin: Durable backdoors in federated learning. In *International Conference on Machine Learning*, pages 26429–26446. PMLR, 2022. 2, 4, 6