PASS: A Parameter Audit-Based Secure and Fair Federated Learning Scheme Against Free-Rider Attack

Jianhua Wang [ID], Xiaolin Chang [ID], *Senior Member, IEEE*, Jelena Mišić [ID], *Fellow, IEEE*,
Vojislav B. Mišić [ID], *Senior Member, IEEE*, and Yixiang Wang [ID]

Abstract—Federated learning (FL) as a secure distributed learning framework gains interests in Internet of Things (IoT) due to its capability of protecting the privacy of participant data. However, traditional FL systems are vulnerable to free-rider (FR) attacks, which causes unfairness, privacy leakage and inferior performance to FL systems. The prior defense mechanisms against FR attacks assumed that malicious clients (namely, adversaries) declare less than 50% of the total amount of clients. Moreover, they aimed for anonymous FR (AFR) attacks and lost effectiveness in resisting selfish FR (SFR) attacks. In this article, we propose a parameter audit-based secure and fair FL scheme (PASS) against FR attack. PASS has the following key features: 1) prevent from privacy leakage with less accuracy loss; 2) be effective in countering both AFR and SFR attacks; and 3) work well no matter whether AFR and SFR adversaries occupy the majority of clients or not. Extensive experimental results validate that PASS: 1) has the same level as the state-of-the-art method in mean square error against privacy leakage; 2) defends against AFR and SFR attacks in terms of a higher defense success rate, lower false positive rate, and higher F1-score; and 3) is still effective where adversaries exceed 50%, with F1-score 89% against AFR attack and F1-score 87% against SFR attack. Note that PASS produces no negative effect on FL accuracy when there is no FR adversary.

Index Terms—Federated learning (FL), free-rider (FR) attack, Internet of Things (IoT), privacy preserving.

I. INTRODUCTION

WITH the growing attention to data privacy, federated learning (FL), a powerful and secure distributed machine learning paradigm, is being widely used in Internet of Things (IoT) [1], [2], [3], [4]. It is formulated as a multiround model-training strategy between multiple agents. FL

participants, such as remote laptops and edge IoT devices controlled by a central server, co-maintain a global model without sharing their private data sets. The standard procedure of conventional FL is illustrated in the left part of Fig. 1. Concretely, in step 1, the central server allocates the initialized model to each participant. In step 2, FL participants train the global model via their private data set. In step 3, FL participants upload the local update to the central server. In step 4, the server adopts the model aggregation algorithms to aggregate the local update. In step 5, FL participants receive a new global model update from the central server to continue the subsequent training process until the jointly trained model converges.

The past years witnessed various attacks reported in conventional FL systems and model aggregation algorithms, such as free-rider (FR) attacks [5], [6], [7]. This type of attack allows FR adversaries to benefit from a well-trained global model without contributing their own private data sets and computing resources. The right part of Fig. 1 illustrates an FL scenario under FR attacks, where an FR adversary uploads a fake local update to the central server in *red step 3*. This upload can lead to the emergence of opportunistic behaviors and then the following severe threats: 1) the jointly trained model is more susceptible to reducing the accuracy of specific tasks [8]; 2) FR adversaries obtain the leakage gradients or local updates to reconstruct the personal data [9], [10] and then conduct model inversion [11]; and 3) some FR attacks can lead to unfair training and then lower the devotion aspiration of honest participants [8].

According to whether an FR adversary dominates private data and computing power, FR attacks are categorized into two types [12]: 1) anonymous FR (AFR) and 2) selfish FR (SFR) attacks. AFR denotes that adversaries do not own any private data sets and computation resources. It is a generic form of Gaussian attack [13], where an AFR adversarial client uploads its stochastic Gaussian noise to the central server [5]. SFR means that adversaries have their own private data set and training ability but they are unwilling to devote their data and computation resources to global model training. SFR attack gives rise to a new threat to FL.

Researchers have proposed methods to resist FR attacks, including contribution-based and robust model aggregation-based methods [8]. However, the existing works
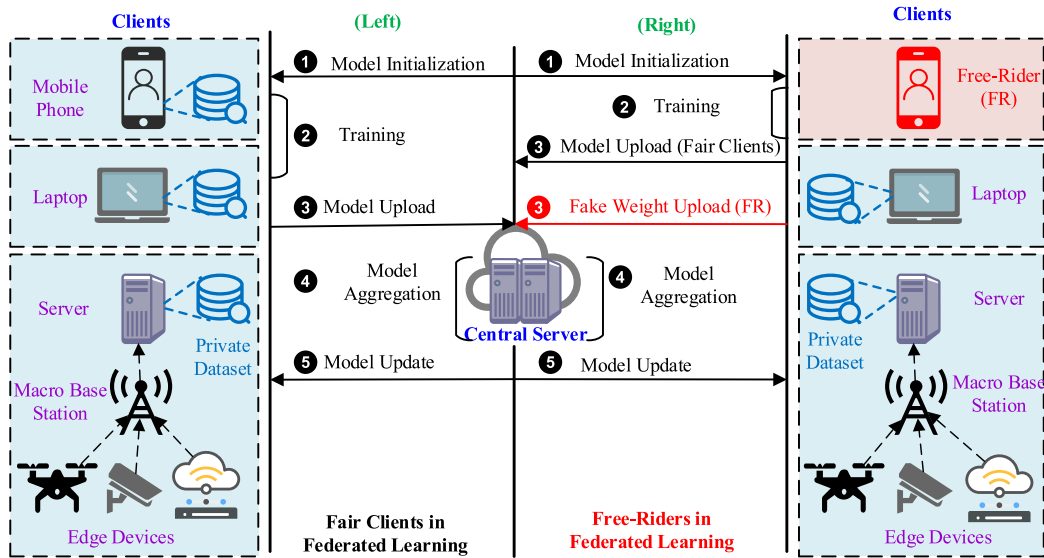
Fig. 1. Schematic of FL with fair clients and FR clients.

only concentrate on resisting AFR attack and suffer from the following shortcomings.

1) They lose defense capability when adversaries are the majority in the system [14]. Namely, their methods, usually make strong assumptions that most FL clients are honest [14], [15].

2) It is hard, if not impossible, for them to resist SFR attack. Compared with AFR attack, SFR attack is more sequestered and challenging to be identified [12]. The common contribution-based methods depend on clients' contributions by calculating cosine similarity between the global update and local update [16]. However, similarities may make low contributions to high-quality workers [17], especially in SFR attack scenarios [12]. Moreover, robust model aggregation-based methods cannot eliminate FR adversaries since they only focus on overall performance. Hence, they misclassify fair clients into FR adversaries and cause a higher false positive rate (FPR).

Motivated by the above discussions, we propose a parameter audit-based secure and fair FL scheme (PASS) against both AFR and SFR attacks in this article, which can not only address the aforementioned problems in AFR attack-oriented defense mechanisms but also resist the SFR attack. PASS contains two components: 1) contribution evaluation (denoted as *PASS-CE*) with the aim to audit parameters and 2) a privacy-preserving strategy (denoted as *PASS-PPS*) combining the weak Differential Privacy [18] with Gaussian mechanism and parameter prune approach.

To the best of our knowledge, we are not only the first to propose the parameter audit-based method against the AFR attack but also the first to investigate the defense in SFR attack scenarios. We summarize the key features of PASS as follows.

1) PASS can protect private data with less accuracy loss. It is achieved by the component PASS-PPS with variance 10^{-2} Gaussian perturbation. PASS-PPS has lower than 2% accuracy loss in Cifar10 nonindependent

and identically distributed (i.i.d.) and 3% in MNIST non-i.i.d.

2) PASS can defend against both AFR and SFR attacks in FL systems. It is achieved by the component PASS-CE. It focuses on individual performance intuitively and then it can effectively distinguish FR adversaries from fair clients.

3) PASS works well even if AFR and SFR adversaries make up the majority of the total amount of clients. The existing defense models lose effectiveness when AFR adversaries are more than 50% in the FL system [14], [15].

Notably, frequent communications between the server and clients in the parameter audit phase will cause more time consumption. To address this problem, we adopt synchronous transmission to decrease communication times and use parameter prune to sparse local updates, and then reduce communication overhead. We analyze the overhead reduction theoretically in Section III-D.

Extensive experiments are carried out to evaluate the effectiveness of PASS and explore the proper hyperparameters of PASS in Cifar10 and MNIST with i.i.d. and non-i.i.d. data. Notably, the commonly used model aggregation algorithm, federated averaging (FedAvg), is adopted. The experiment results demonstrate as follows.

1) PASS has the same level as the state-of-the-art (SOTA) method [16] in mean square error (MSE) against privacy leakage, as shown in Fig. 8.

2) Compared with other FR attack defense models [8], [15], [19], PASS can achieve a 100% defense success rate (DSR), 20% FPR, and the highest F1-score (on average) of 88% against SFR and 89% against AFR, which are the best performance compared with other defense schemes [8], [15], [19].

3) PASS achieves better defense performance no matter whether AFR and SFR adversaries occupy the majority of clients or not. Especially in the SFR attack, PASS

TABLE I
LIST OF ABBREVIATIONS USED IN ARTICLE

Notation	Description
FL	Federated Learning
IoT	Internet of Things
FR	Free-Rider
AFR	Anonymous Free-Rider
SFR	Selfish Free-Rider
IID	Independent and Identically Distributed
PASS	Parameter Audit-based Secure and fair FL Scheme
PASS-CE	Contribution Evaluation in PASS
PASS-PPS	Privacy-Preserving Strategy in PASS
CE	Contribution Evaluation
FedAvg	Federated Averaging
SOTA	State-Of-The-Art
DSR	Defense Success Rate
FPR	False Positive Rate

obtains the highest F1-score of 87% in adversaries more than 50% scenario and 89% in adversaries less than 50%, illustrated in Fig. 13.

The remainder of this article is organized as follows. Section II presents preliminary knowledge and related work. Section III presents the PASS design, and Section IV describes the experimental results. Section V summarizes the conclusion. Table I concludes the commonly used notations and those descriptions in this article.

II. RELATED WORK

This section first presents the related work of FL and FR attacks. Then, related work on contribution evaluation (CE) in FL is presented. Finally, related work on defense mechanisms against FR attacks is given.

A. Free-Rider Attack

In FL scenarios, FR adversaries represent a portion of individuals who benefit from a well-trained high-quality global model without contributing computation resources and private data [5], [12], [20], [21], [22]. Wang et al. [12] defined AFR adversaries with no privacy data set and no training ability of a large model. Several studies focused on the AFR attack, such as disguised free-riding [5], novel FR [12], and advanced delta weights [22]. In contrast, SFR adversaries are unwilling to devote their privacy data set and resources to the global model, such as the advanced FR attack [12]. Hence, we adopt AFR and SFR attacks to demonstrate the defense capability of PASS.

B. Contribution Evaluation Methods

CE of participants is one of the crucial issues in FL systems because the fair incentive mechanism will evoke the training passion of distributed clients [8]. In addition, unbiased CE will attract clients to devote high-quality and private data sets to FL training [23].

Various researches have been devoted to designing reasonable CE methods [24], [25], [26], [27]. Zhan et al. [24] summarized two approaches to evaluate the user's contribution to designing incentive mechanisms, including data quantity and quality. Bao et al. [28] proposed FLChain with the third-party auditable method to evaluate the contribution of clients. However, FLChain uses a trusted third party to audit gradient for CE. This method has two weaknesses:

1) if the third party is a free rider, it takes the opportunity to obtain gradient updates and disclose privacy;
2) if the third party is trusted and honest, it is uncertain that the audit result is fair because the performance of a model trained by different data sets is distinct.

Liu et al. [26] summarized four CE approaches:

1) self-report the data quality, quantity, and committed computational and communication resources to the server;
2) utility game, which focuses on the changes when a client joins in the FL system;
3) Shapley value, which evaluates the contribution of clients via ablation experiments;
4) individual performance.

The existing CE methods have two weaknesses.

1) They have high requirements for honest and truthful clients, such as self-report and Shapley value solutions.
2) They strongly rely on the participating order of each client, such as the utility game solution.

This article emphasizes that individual performance is the most intuitive and reasonable approach for CE, and we utilize PASS-CE to audit parameters. To reduce the influence of the threshold, we implement theoretical analysis and extensive experiments (in Section IV-E) to explore the most appropriate hyperparameters.

C. Defense Mechanisms Against FR Attacks

There are two types of defense methods to resist FR attacks: 1) contribution-based and 2) robust model aggregation-based methods. RFFL [8], a contribution-based method, utilizes a reputation mechanism to evaluate the contribution of clients. median, trimmed mean [19], and SignSGD [15] are the representative robust model aggregation-based methods. Yin et al. [19] utilized coordinatewise median and coordinatewise trimmed mean instead of weighted averaging. SignSGD is a communication-efficient approach proposed in [15], where participants only upload the elementwise signs of the gradients without the magnitudes.

Notably, there is a critical difference between the above defense mechanisms. Robust model aggregation-based methods tolerate the negative effect of adversaries instead of detecting and removing them. But contribution-based methods, such as RFFL, focus on eliminating adversaries to ensure the security of FL. Furthermore, there are several weaknesses in the methods mentioned above. They cannot handle the situation where attackers occupy more than 50% [14], and only focus on AFR attacks. Moreover, the median and trimmed median significantly reduce the accuracy of complex models on non-i.i.d. data and cannot protect the confidentiality of training data [29]. The SignSGD model magnifies the local update, leading to substantially degraded accuracy [30]. In RFFL, using similarities between local and global updates may reduce the contribution of high-quality workers [17]. In this
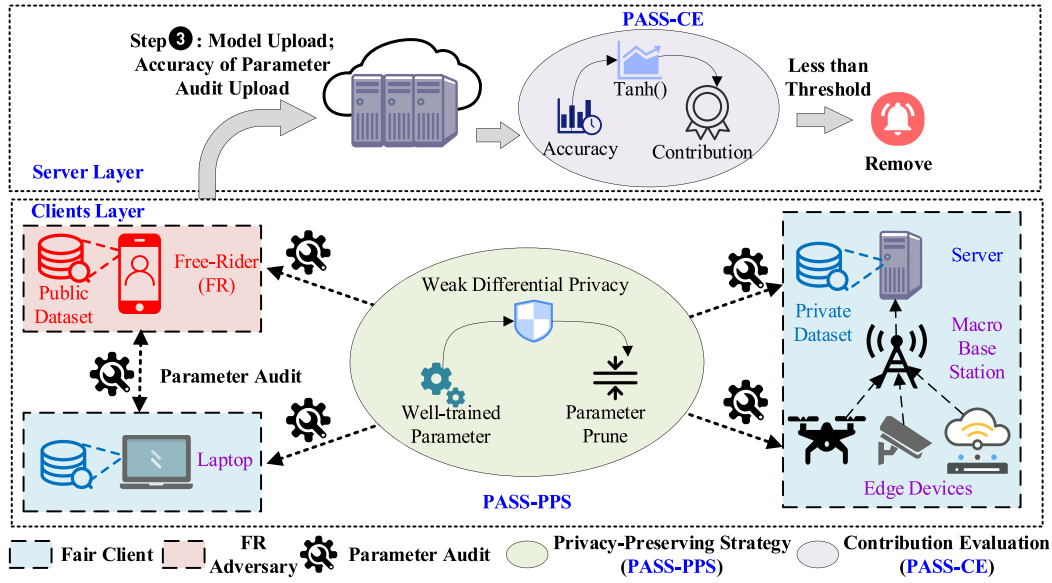
Fig. 2.   Step (3) in Fig. 1: the PASS diagram after being allocated initialized model.

article, similar to RFFL, our method PASS will eliminate the adversaries from the FL system. In addition, PASS uses reasonable parameter audit for CE and achieves better defense performance than others no matter whether AFR and SFR adversaries occupy the majority of clients or not.

III. PASS DESIGN

This section first introduces PASS in Section III-A and then details its components, PASS-PPS and PASS-CE, in Sections III-B and III-C, respectively. Finally, time analysis is presented. Table II gives the notations frequently mentioned in PASS.

A. PASS Description

PASS aims to establish a secure and fair FL system against FR attacks. It has the following design goals.

Goal 1: PASS should protect the privacy data set of each client from the deep leakage from gradient (DLG) attack [10].

Goal 2: PASS should maintain the performance of the FL system in terms of high accuracy and low loss when we adopt PASS-PPS.

Goal 3: PASS should distinguish between the FR adversaries and fair clients based on the contribution of clients with lower FPRs.

Goal 4: In PASS, the time consumption between clients and the central server should be controlled at an acceptable level.

Fig. 2 depicts the flow diagram of PASS. After obtaining initialized global model parameters θ^0, each FL client trains θ^0 using a private data set to obtain the local model update $\widetilde{\theta}_i^{r+1}$, where $i \in N$. Then, client i utilizes the *PASS-PPS* to receive the secure local model update $\Delta \widetilde{\theta}_{i,ldp}^{r+1}$ (in lines 20 and 21 of Algorithm 1). Moreover, client i uploads $\Delta \widetilde{\theta}_{i,ldp}^{r+1}$ to the central server. Notice that $\Delta \widetilde{\theta}_{i,ldp}^{r+1}$ is not only used to audit parameters by other clients who possess their privacy data set but also to achieve model aggregation.

TABLE II
NOTATIONS FREQUENTLY USED IN PASS

Notation	Description
η	Learning rate
ε	Gaussian noise
α	Moving average coefficient
β	Threshold coefficient
γ	Pruning rate
r	Round r
R	Total number of round
i	Client i
N	Total number of participants
σ_n^2	The variance of Gaussian noise, where n is the dimension of the parameter
θ^0	Initialized global model parameter
$\tilde{\theta}^0$	Initialized local model parameter
$\tilde{\theta}_i^{r+1}$	Local model update of the client i
$\Delta \tilde{\theta}_{i,ldp}^{r+1}$	Local model update with weak DP with Gaussian mechanism of the client i in round $r+1$
$AccDiv_i^r$	Accuracy divergence between θ^r and $\tilde{\theta}_{i,ldp}^r$
c_i^r	Client i contribution in round r
$\dfrac{1}{\beta \times N}$	Threshold value

As stated in line 12 of Algorithm 1, once received $\Delta \widetilde{\theta}_{n,ldp}^{r+1}$ from each client, the server will allocate the new global model parameters $\Delta \theta^{r+1}$ and the local update of each client of the previous round $\Delta \widetilde{\theta}_{i,ldp}^r$ to achieve parameter audit. The detailed algorithm is described in Algorithm 1.

After that, on the client side, each client audits the received local update using their private data set and calculates the accuracy divergence $\text{AccDiv}_i^r = \text{Acc}(\theta^r) - \text{Acc}(\Delta \widetilde{\theta}_{i,ldp}^r + \widetilde{\theta}_{i,ldp}^{r-1})$ of each local update with the previous global update, and we call this parameter audit. On the server side, the server gets the $\text{AccDiv}_i^r, i \in N$ and calculates the client contribution c_i

---

Algorithm 1 PASS Algorithm

Input: Round R, Initial global model parameters θ^0, Client number N, Learning rate η, Threshold coefficient β, Gaussian noise ε, Pruning rate γ, Moving average coefficient α
Output: Update Model Parameters θ
Initialize: $\widetilde{\theta}^0 = \theta^0$

1: **For** r in range (R):
2:      #=========== **Server**===========
3:      **If** $n \in N$: // client n is not eliminated
4:          Allocate $\widetilde{\theta}^0$ to client n
5:          To step 16
6:          **For** i in range (N): // calculate contribution
7:             $c_i^r = \dfrac{1}{\alpha \times c_i^{r-1} + (1-\alpha) \times \tanh\left(\frac{1}{N-1}\sum_i^{N-1} AccDiv_i^r\right)}$
8:             **If** $c_i^r < \frac{1}{\beta \times N}$: Eliminate i from N
9:             **End if**
10:         **End for**
11:         $\Delta\theta^{r+1} = FedAvg(\sum_{i\in N} \Delta\widetilde{\theta}_{i,ldp}^{r+1})$ // model aggregation
12:         $(\Delta\theta^{r+1}, \Delta\widetilde{\theta}_{\text{client }1,ldp}^r, \Delta\widetilde{\theta}_{\text{client }2,ldp}^r, \ldots, \Delta\widetilde{\theta}_{\text{client }N,ldp}^r)$
        to Client i
13:         To step 16 to continue the FL training
14:     **End if**
15:         # ===========**Client** i =========
16:     **If** $r == 0$: Obtain the θ^0 from the Server
17:     **Else:** Obtain the $\Delta\widetilde{\theta}^r$ from the Server
18:     **End if**
19:     Calculate local update $\Delta\widetilde{\theta}_i^{r+1}$ by training with a private dataset
20:     $\Delta\widetilde{\theta}_{i,ldp}^{r+1} = \Delta\widetilde{\theta}_i^{r+1} + \varepsilon, \varepsilon \sim \mathcal{N}\left(0, \sigma_n^2\right)$ // weak DP with Gaussian mechanism
21:     $\Delta\widetilde{\theta}_{i,ldp}^{r+1} = (1-\gamma) \cdot \Delta\widetilde{\theta}_{i,ldp}^{r+1}, \gamma \in [0,1)$ // parameter prune
22:     $AccDiv_i^r = Acc(\theta^r) - Acc\left(\Delta\widetilde{\theta}_{i,ldp}^r + \widetilde{\theta}_i^{r-1}\right), i \in [1, N-1]$
    // Accuracy divergence of previous round using own dataset
23:     Upload $(\Delta\widetilde{\theta}_{i,ldp}^{r+1}, AccDiv_1^r, AccDiv_2^r, \ldots, AccDiv_{N-1}^r)$
    to Server // upload the result of parameter audit
24:     To step 6 to continue the FL training
25: **End for**
26: **Return** Model Parameters θ

---

by the activation function $\tanh(\cdot)$ to map c_i^r to $[-1, 1]$, which will divide c_i^r into positive and negative. Once c_i^r is lower than the threshold $[1/(\beta \times N)]$, client i will be eliminated from FL system.

To achieve *Goal 1*, *Goal 2*, and *Goal 3*, we need to satisfy

$$\max_{\Delta\widetilde{\theta}'} \; \|D - D_{\text{DLG}}\|^2 \tag{1}$$

$$\text{s.t.,} \quad \left|\text{Acc} - \text{Acc}'\right| \le \varepsilon \tag{2}$$

$$\underset{\beta,\gamma \in \text{PASS}}{\arg\min} \; \text{FPR}, \underset{\beta,\gamma \in \text{PASS}}{\arg\max} \; \text{DSR}. \tag{3}$$

In (1), $\Delta\widetilde{\theta}'$ denotes the local update with P*ASS-PPS*. Here, D is the raw input data and D_{DLG} is the data reconstructed by the DLG attack. In (2), *Acc* denotes the accuracy of conventional FL while Acc' is that of the FL with *PASS-PPS*, and ε means a narrow range. In (3), DSR and FPR are the PASS-CE metrics introduced in Section IV-C.

B. PASS-PPS

PASS is a parameter audit-based FL scheme to evaluate the client's contribution to the SFR attack defense. However, in the FL system, the server always adopts the stochastic gradient

descent (SGD) optimizer with a learning rate η. Thus, some adversarial clients can collect the updated gradient to reconstruct the private data set [10]. That is, the updated gradient is likely to be leaked, according to

$$\nabla\theta^{r+1} = \frac{\theta^r - \theta^{r+1}}{\eta} \tag{4}$$

With this in mind, and to satisfy *Goal 1*, we utilize *PASS-PPS* to ensure data set privacy. As described in Algorithm 1, after calculating local update $\Delta\widetilde{\theta}_i^{r+1}$ by training with the private data set, the client will utilize weak DP with Gaussian mechanism to add perturbation $\Delta\widetilde{\theta}_{i,ldp}^{r+1} = \Delta\widetilde{\theta}_i^{r+1} + \varepsilon$, where $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$. Moreover, $\Delta\widetilde{\theta}_{i,ldp}^{r+1}$ will be randomly pruned with the pruning rate γ to reduce the communication overhead and guarantee more privacy. Section IV-D demonstrates the achievement of *Goal 1* and *Goal 2*.

C. PASS-CE

In CE methods, some researchers adopted cosine similarity to measure the angular distance between updates and to determine the quality of model updates [8], [31], [32]. However, the SFR adversaries will not be recognized effectively in the FL system.

Thus, to satisfy *Goal 3*, we utilize *PASS-CE*, namely, validating the other client's local update using the own private data set on the client side. Notice that, receiving $AccDiv_i^r$, the server will calculate the averaging $AccDiv_i^r$ using $([1/(n-1)]\sum_i^{n-1} AccDiv_i^r)$ as the fundamental contribution value. Then, the contribution of client i in round r is calculated using the formula $c_i^r = [1/(\alpha \times c_i^{r-1} + (1-\alpha) \times \tanh([1/(N-1)]\sum_i^{N-1} AccDiv_i^r))]$. After that, the server will remove the i where c_i^r lower than the threshold $[1/(\beta \times N)]$ (in line 8 of Algorithm 1). Section IV-E indicates the achievement of *Goal 3*.

Theorem 1 (Threshold Coefficient): Under Algorithm 1, if PASS works, we suppose that the fair client contribution c_i^r satisfies $c_i^r \ge [1/(\beta \times N)]$. Namely, given the $\beta \ge (1/[c_i^r \times N])$, we have the threshold coefficient $\beta \ge 1$.

Proof: See the Appendix. ∎

D. Time Complexity Analysis

This section presents time complexity analysis in order to highlight that the pruning strategy can reduce the extra time overhead caused by our PASS.

Frequent communications between server and client will consume a lot of time. Thus, we adopt two strategies to reduce time consumption: 1) (*strategy a*) synchronous transmission and 2) (*strategy b*) parameter pruning.

In (*strategy a*), as stated in line 12 of Algorithm 1, we allocate the global update $\Delta\widetilde{\theta}^{r+1}$ together with the local update $\Delta\widetilde{\theta}_{\text{client }i,ldp}^r$. And assuming that in a conventional FL system, the communication consumption between the server and each client in each round is $\mathcal{O}(1)$, the time consumption for all clients in all rounds is $\mathcal{O}((N-1) \times N \times R)$ while conventional FL is $\mathcal{O}(1 \times N \times R)$. In the parameter audit phase, the validation consumption is very low in machine learning. Hence, we can ignore the time in each round.

In (*strategy b*), motivated by [33], the parameter prune is a practical approach to promote the efficiency of FL training and reduce the communication and computation overhead in FL. Thus, we not only adopt parameter prune as part of *PASS-PPS* but as an effective training accelerating method.

In this situation, the time complexity is $(1 - \gamma) \times (N - 1)\mathcal{O}(1 \times N \times R)$, notably $\gamma \in [0, 1)$. Compared with $\mathcal{O}(1 \times N \times R)$, the communication overhead of PASS is acceptable. Hence, *Goal 4* is satisfied.

IV. EXPERIMENT EVALUATION

This section first describes the used data set, baselines, and experiment settings. Then, we introduce the experimental metrics for evaluating FR attack and *PASS*. Finally, we demonstrate the experimental results of *PASS-PPS*, which satisfies *Goal 1* and *Goal 2*. We also conduct comparisons with other defense models against AFR and SFR attacks, which satisfies *Goal 3*.

A. Data Sets

MNIST [34] and Cifar10 [35] are used as standard classification baseline data sets. MNIST data set includes 28×28 handwritten digits with ten classes and has become the most well-known data set in the classification task. Cifar10 is made up of ten classes of 32×32 images with three RGB channels and consists of 50000 training samples and 10000 testing samples.

Great experiments of studies in FL utilize MNIST and Cifar10 as baseline data sets, so we adopt them in our experiments. Notably, MNIST size is 28×28 while Cifar10 is 32×32, so we assume that FR owns MNIST and honest clients have Cifar10 in FL training. In addition, for the smooth training of the model, we need to extend the same tensor dimension of MNIST as Cifar10.

B. Baseline

We implement RFFL [8], median [19], trimmed median [19], and SignSGD [15] for comparison in order to reveal PASS's better performance. The experimental settings of each defense model are appropriately utilized.

As stated in [36], the SOTA robust model aggregation rules include median, trimmed median, and SignSGD, which are compared frequently in [8] and [32]. On the other hand, RFFL is a SOTA model against AFR attacks. So we implement comparisons in our experiments. Notably, the robust model aggregation-based methods only focus on the overall performance, so we utilize the RFFL threshold standard to remove FR adversaries for evaluating the defense performance in median, trimmed median, and SignSGD. Notably, in [8], RFFL focuses on individual accuracy instead of the fairness of the FL system, so RFFL may not achieve a better performance than others.

C. Experimental Setting

This section gives detailed experimental settings. Then, we introduce the evaluation metrics used in each scenario.

TABLE III
AFR AND SFR ATTACK DETAILS

Fair Client Num	Adversary Num	Adversary Ratio	Data Split	Train Num	Test Num
10	1	9%	MINST iid/ non-iid	540	60
	5	33%			
	15	60%			
	20	67%			
	1	9%	Cifar10 iid/ non-iid	1600	400
	5	33%			
	15	60%			
	20	67%			

TABLE IV
HYPERPARAMETERS IN SFR ATTACK

Target Dataset Trained by Fair Client		Pre-trained Dataset Trained by SFR Adversary		Optimizer	Learning Rate
Cifar 10	iid	MNIST	iid	Adam	0.015 (Decay: 0.997)
	non-iid		non-iid		

Note: 1) Decay = Learning Rate Decay, which means slowly reducing or decaying the learning rate after each round. 2) Optimizer utilizes the default settings.

1) Federated Learning (Models): We implement a 2-layer convolutional neural network (CNN) [37] for MNIST and a 3-layer CNN [38] for the Cifar10 data set as the base model in FL.

Experimental Setting: The FL is trained via SGD optimizer with learning rate $\eta = 0.1$ and round $r = 200$. FedAvg is adopted as the model aggregation algorithm. In addition, we consider two common types of data: 1) i.i.d. and 2) non-i.i.d., in the MNIST and Cifar10 data sets.

2) Free-Rider Attack (Experimental Setting): Table III gives the number of fair and AFR and SFR adversaries, the AFR and SFR adversary ratio in all clients, the type of data, and the number of samples in training and testing. Notably, we consider the AFR and SFR adversary ratio in three scenarios, including 9%, 33%, 60%, and 67%.

In Table IV, the first column denotes the target data set, namely, the data trained by fair clients in FL. The second column is the pretrained data set trained by SFR adversaries, representing the SFR adversary unwilling to contribute the privacy data set to FL model training. Besides, the pretrain data set MNIST means we extend the same MNIST model tensor dimension as Cifar10 to simulate the SFR attack. In addition, we implement Adam optimizers to conduct the subsequent training in the SFR attack. The last column is the learning rate and decay of Adam.

3) PASS (DLG Comparison): The DLG experimental settings are analogous to that in [10] and [16]. We implement L-BFGS [39] optimizer and conduct 300 iterations of optimization to reconstruct the raw data.

Hyperparameters: In PASS-PPS, we adopt the Gaussian noise distributions $\mathcal{N} \sim (0, \sigma_n^2)$ where the variance satisfies $\sigma_n^2 \in \{0, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$. Parameter pruning rate γ satisfies $\gamma \in [0, 1)$. In PASS, the moving average coefficient α is similar to that of [8], namely, $\alpha = 0.95$. The threshold coefficient β satisfies $\beta \geq 1$, which is proved in
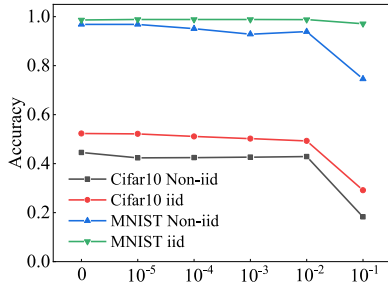
Fig. 3. Accuracy with various noise levels $\sigma_n{}^2$.

Fig. 4. Loss with various noise levels $\sigma_n{}^2$.

Appendix. The threshold value of PASS is $[1/(\beta \times N)]$, in which N denotes the number of participants in the FL system.

4) Evaluation Metrics: This section presents evaluation metrics used in DLG and PASS.

DLG Evaluation: We utilize MSE between the reconstructed data and raw input to quantify the effectiveness of defenses. A lower MSE indicates a more possibility of data leakage. In addition, we adopt accuracy to reveal the performance loss after using PASS-PPS. It is not acceptable that the accuracy of the whole model decreases sharply. In general, we need to achieve the tradeoff with higher MSE and lower accuracy reduction.

PASS Evaluation: We define DSR to reveal the effectiveness of the defense system. DSR, as stated in (5), denotes the eliminating ratio of FR adversaries in a defense system. Moreover, we adopt a FPR, as stated in (6), to evaluate the performance of attacking defense baselines using AFR and SFR attacks. FPR denotes the removal ratio of fair clients in the detection. F1-score [shown in (7)] denotes the overall performance between DSR and FPR. In general, a better defense model means lower FPR, higher DSR, and higher F1-score

$$\text{DSR} = \frac{\#\text{Number of Eliminating FR Clients}}{\#\text{Number of All FR Clients}} \times 100\% \tag{5}$$

$$\text{FP} = \frac{\#\text{Number of Eliminating Fair Clients}}{\#\text{Number of All Fair Clients}} \times 100\% \tag{6}$$

$$\text{F1-score} = \frac{2 \cdot \frac{\text{TP}}{\text{TP+FP}} \cdot \frac{\text{TP}}{\text{TP+FN}}}{\frac{\text{TP}}{\text{TP+FP}} + \frac{\text{TP}}{\text{TP+FN}}} \times 100\%. \tag{7}$$

D. Evaluation Results of PASS-PPS

This section demonstrates the effectiveness of PASS-PPS against the DLG. Motivated by [10], which has proposed the Gaussian noise with variance range from 10^{-1} to 10^{-4} to defend DLG, we implement weak DP with Gaussian mechanism in experiments. To satisfy *Goal 1* and *Goal 2*, we divide this section into two parts. First, to achieve *Goal 1*, we demonstrate the accuracy and loss curves using Gaussian perturbation with various perturbation levels and pruning rates. Then, to satisfy *Goal 2*, we uncover the performance against DLG when adopting the PASS-PPS.
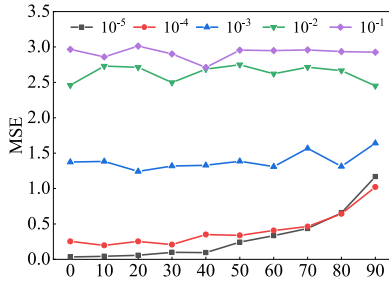
1) Gaussian Weak DP Noise Level and Parameter Prune: At first, we conduct six noise levels in Gaussian weak DP where variance $\sigma_n^2 \in \{0, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$. As illustrated in Fig. 3, the accuracy gradually decreases with
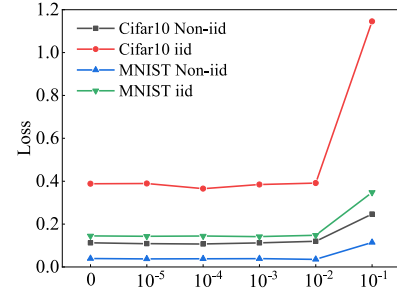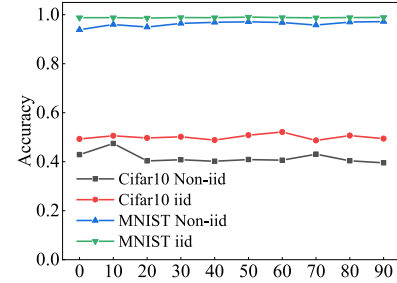
Fig. 5. Accuracy with γ varying from 0 to 90 when $\sigma_n^2 = 10^{-2}$.
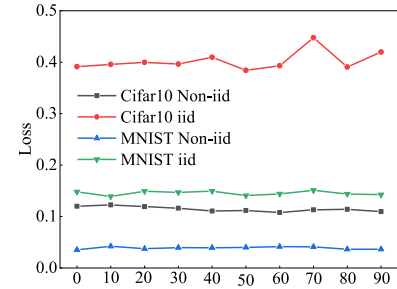
Fig. 6. Loss with γ varying from 0 to 90 when $\sigma_n^2 = 10^{-2}$.

more noise. However, a sharp drop appears when $\sigma_n^2 = 10^{-1}$. Hence, when we add $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$ where $\sigma_n^2 = 10^{-2}$, the accuracy in MNIST and Cifar10 is similar to that of $\sigma_n^2 = 0$ while the trained data has the maximum privacy. According to Fig. 4, the loss curves show a similar result. When $\sigma_n^2 = 10^{-1}$, the loss curve of each data split increases.

Moreover, we implement parameter prune to reduce the leakage of privacy. The pruning rate γ satisfies $\gamma \in [0, 1)$ and varies from 0 to 90% in experiments. According to Figs. 5 and 6, we find that the accuracy and loss curves are stable in varying γ. Thus, to satisfy *Goal 1*, we should select $\sigma_n^2 = 10^{-2}$ no matter which pruning rate is.

2) PASS-PPS Performance Against DLG: To achieve *Goal 2*, we need to test the PASS-PPS performance against DLG in this section. First, we measure the MSE in various noise levels and parameter pruning rates. In Fig. 7, the MSE is more than 2.0 when $\sigma_n^2 \geq 10^{-2}$ while MSE is lower than 1.0 when $\sigma_n^2 \leq 10^{-4}$. As evaluated in [16], when MSE is more than 1.49, the system can defend the DLG. Fig. 8 depicts the MSE comparison with Soteria [16]. If we want to achieve the same defense level as [16], the MSE of PASS-PPS is more than 1.49, which signifies we need select $\sigma_n^2 \geq 10^{-2}$.

Fig. 7. MSE of DLG in varying γ and $\sigma_n{}^2$ in Cifar10.

Fig. 8. MSE comparison with Soteria [16] against DLG.

TABLE V
PASS PERFORMANCE OF DSR AND FPR IN
VARYING β AGAINST SFR ATTACKS

β	Metrics (%)	Number of FR adversary				Avg (%)
		1 (9%)	5 (33%)	15 (60%)	20 (67%)	
1.00	DSR	100	100	100	100	100
	FPR	65	65	71	100	75
	F1-score	52	52	45	0	37
1.25	DSR	100	100	100	100	100
	FPR	38	43	43	100	56
	F1-score	77	73	73	0	56
1.50	DSR	100	100	60	100	90
	FPR	32	37	27	40	34
	F1-score	81	77	66	75	75
1.75	DSR	100	100	40	100	**85**
	FPR	23	20	19	30	**23**
	F1-score	87	89	60	82	**80**
2.00	DSR	100	100	20	0	55
	FPR	20	20	13	10	16
	F1-score	89	89	57	46	70

In conclusion, considering the Gaussian noise level in 1) and the PASS-PPS performance against DLG in 2), we adopt the $\sigma_n{}^2 = 10^{-2}$ as the Gaussian noise level.

E. Selection of β and γ in PASS-CE

In the Appendix, we prove that the threshold coefficient β needs to satisfy $\beta \geq 1$. In this section, we investigate the PASS performance of varying β and γ.

To find the best β, we first implement varying β under the pruning rate $\gamma \in (0, 1]$. Table V shows the experiment results with ten fair clients and 1, 5, 15, and 20 SFR adversaries under $\gamma \in (0, 1]$ and varying β. We observe that when $\beta = 1.00$, PASS can 100% defend SFR attack, but the FPR is 75% which means 75% of fair clients have been eliminated from the FL system by mistake. On the other hand,
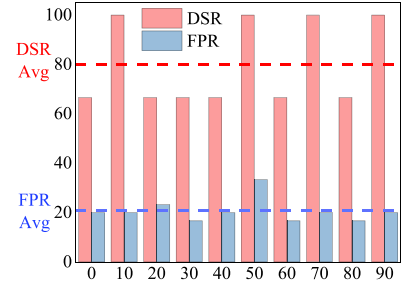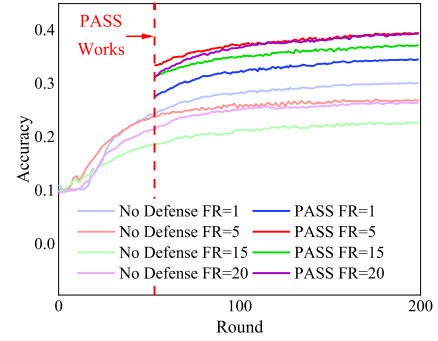

Fig. 9. DSR and FPR in varying γ when $\beta = 1.75$.

Fig. 10. Accuracy curves of PASS and no defense.

with $\beta = 2.00$, PASS achieves 16% FPR, but the DSR is 55% meaning only 55% of SFR adversaries are eliminated from the FL system. Apparently, those situations are not proper in the defense model. Finally, we select $\beta = 1.75$ as the threshold coefficient value. In this situation, PASS achieves a DSR of 85%, an FPR of 23%, and an F1-score of 80%, reaching the tradeoff between DSR and FPR.

To investigate the effect of varying γ, we fix $\beta = 1.75$ and conduct extensive experiments. In Fig. 9, the red dashed line denotes the average DSR of 80%, and the blue dashed line indicates the average FPR of 21%. The red bar represents the DSR, and the blue bar represents the FPR with varying γ. To satisfy *Goal 3*, we must achieve a tradeoff between DSR and FPR. With this in mind, we choose $\gamma = 10\%$, $\gamma = 70\%$, and $\gamma = 90\%$. However, to satisfy *Goal 4*, we need to utilize larger γ to reduce communication consumption. Hence, we adopt $\gamma = 90\%$ as the pruning rate. In this situation, we achieve 100% DSR and 20% FPR.

To demonstrate the effectiveness of PASS, we provide three groups of experiments with $\beta = 1.75$ and $\gamma = 90\%$, which include no defense scenario and PASS scenario with 1, 5, 15, and 20 SFR adversaries. As illustrated in Fig. 10, the red dashed line is the PASS worked round. Compared with the no-defense scenario, the PASS scenario eliminated the SFR adversaries and increased accuracy. Notably, PASS curves differ because of the existing FPR of 10% in 1 SFR adversary, 30% in 5 SFR adversaries, 20% in 15 SFR adversaries, and 20% in 20 SFR adversaries.

F. PASS Performance Evaluation

In the previous section, we confirm all hyperparameters. As a result, this section investigates the PASS performance
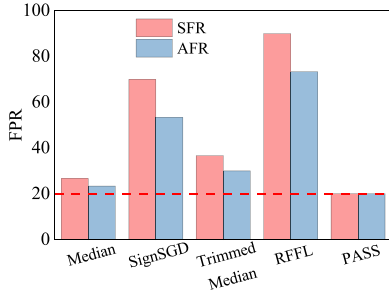
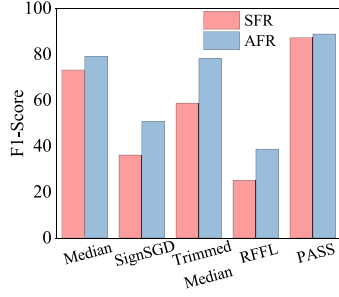Fig. 11. FPR of defense models against AFR and SFR attacks.

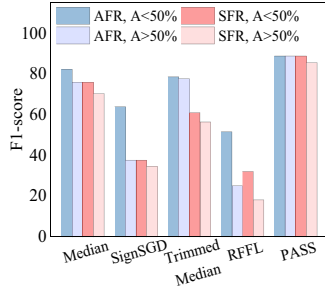Fig. 12. F1-score of defense models against AFR and SFR attacks.

Fig. 13. F1-score of defense models against AFR and SFR attacks with different adversary proportions.

compared with other defense models, including RFFL, median, trimmed median, and SignSGD.

We adopt proper experimental settings for each defense model demonstrated in Section IV-C. Figs. 11–13 illustrate the threefold observations: 1) against the AFR attack and the SFR attack, PASS we proposed achieves the lowest FPR, and the highest F1-score; 2) the SFR attack scenario has a better attack performance than the AFR attack; and 3) compared with other schemes, PASS has the best F1-score whether adversary number (A in Fig. 13) is more than 50% or less than 50%.

In fact, each defense model can reach almost 100% DSR by using proper hyperparameters in AFR and SFR attacks, but they cannot reach a better tradeoff between DSR and FPR. However, our PASS achieves the lowest FPR with 100% DSR and obtains the highest F1-score (on average) of 88% against SFR and 89% against AFR.

V. CONCLUSION

FR attacks enable the FR adversary to benefit from the well-trained model without contributing any private data set and computation resources in FL. Against AFR and SFR attacks,

this article proposes a PASS to evaluate participant contribution intuitively. To protect private data during parameter auditing, we adopt a privacy-preserving strategy (PASS-PPS) that utilizes weak Differential Privacy with a Gaussian mechanism and parameter prune mechanism. For achieving fair FL training, we utilize the new CE method to measure individual performance.

Extensive experiments are conducted to evaluate the performance of PASS. Our PASS-PPS shows a similar defense level as a SOTA model against privacy leakage. In AFR and SFR attacks, PASS has a 100% DSR, the lowest FPR, and the highest F1-score compared with other defense models. Besides, the experiment results demonstrate that PASS produces no negative effect on FL accuracy when there is no FR adversary.

Note that in this article we use a standalone framework to simulate FL on a hardware device, which cannot measure communication cost. Therefore, there is no experiment for evaluating the effectiveness of synchronous transmission and parameter prune in reducing communication overhead caused by frequent communications between the server and clients in the parameter audit phase. In our future work, we plan to establish distributed training experiments in order to evaluate communication costs.

APPENDIX

PROOF OF THEOREM 1 IN SECTION III-C

According to Algorithm 1, we evaluate the contribution of the client i is $c_i^r = [1/(\alpha \times c_i^{r-1} + (1 - \alpha) \times \tanh([1/(N-1)] \sum_i^{N-1} \text{AccDiv}_i^r))]$, and the threshold value is $[1/(\beta \times N)]$. If client i can be the honest participant, the contribution should satisfy $c_i^r \geq [1/(\beta \times N)]$. As a result, we obtain the relationship between c_i^r and threshold $[1/(\beta \times N)]$. In addition, assuming that c_i^r is similar with c_i^{r-1} within one single round, we get the equation $c_i^r \times [\alpha \times c_i^r + (1-\alpha) \times \tanh([1/(n-1)] \sum_i^{n-1} \text{AccDiv}_i^r)] = 1$.

In this situation, we solve that equation with the following roots: $c_i^r = [(-(1 - \alpha)\tanh([1/(N-1)] \sum_i^{N-1} \text{AccDiv}_i^r) \pm z)/(2\alpha)]$, where $z = \sqrt{[(1-\alpha)\tanh([1/(N-1)] \sum_i^{N-1} \text{AccDiv}_i^r)]^2 + 4\alpha}$.

Since c_i^r should be positive, we simultaneously get the relationship $c_i^r \geq [1/(\beta \times N)]$ and receive the relationship: $\beta \geq [1/(c_i^r \times N)]$, where $\tanh([1/(N-1)] \sum_i^{N-1} \text{AccDiv}_i^r) \in [-1, 1]$, and the number of all clients $N \in [1, 25]$ in our scenario. So, we adopt the boundary $\tanh([1/(N-1)] \sum_i^{N-1} \text{AccDiv}_i^r) = 1$ and $N = 1$. In this situation, we can obtain $\beta \geq [(2\alpha)/(\alpha - 1 + \sqrt{(1-\alpha)^2 + 4\alpha})]$, which can be simplified to be $\beta \geq [(2\alpha)/(\alpha - 1 + 1 + \alpha)]$. Thus, we prove the infimum of the threshold coefficient $\beta \geq 1$. ∎

REFERENCES

[1] B. Yin, H. Yin, Y. Wu, and Z. Jiang, "FDC: A secure federated deep learning mechanism for data collaborations in the Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6348–6359, Jul. 2020.

[2] S. I. Popoola, R. Ande, B. Adebisi, G. Gui, M. Hammoudeh, and O. Jogunola, "Federated deep learning for zero-day botnet attack detection in IoT edge devices," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3930–3944, Mar. 2022.

[3] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 3rd Quart., 2021.

[4] M. Wen, R. Xie, K. Lu, L. Wang, and K. Zhang, "FedDetect: A novel privacy-preserving federated learning framework for energy theft detection in smart grid," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 6069–6080, Apr. 2022.

[5] Y. Fraboni, R. Vidal, and M. Lorenzi, "Free-rider attacks on model aggregation in federated learning," in *Proc. 24th Int. Conf. Artif. Intell. Stat.*, Mar. 2021, pp. 1846–1854. [Online]. Available: https://proceedings.mlr.press/v130/fraboni21a.html

[6] L. Lyu, H. Yu, J. Zhao, and Q. Yang, "Threats to Federated Learning," in *Federated Learning: Privacy and Incentive*. Cham, Switzerland: Springer, 2020, pp. 3–16, doi: 10.1007/978-3-030-63076-8_1.

[7] L. Lyu, X. Xu, Q. Wang, and H. Yu, "Collaborative fairness in federated learning," in *Federated Learning*. Cham, Switzerland: Springer, 2020, pp. 189–204.

[8] X. Xu and L. Lyu, "A reputation mechanism is all you need: Collaborative fairness and adversarial robustness in federated learning," in *Proc. ICML Workshop Federated Learn. User Privacy Data Confidentiality*, 2021, pp. 1–13.

[9] W. Wei et al., "A framework for evaluating client privacy leakages in federated learning," in *Proc. Eur. Symp. Res. Comput. Security*, 2020, pp. 545–566.

[10] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–11.

[11] Y. Huang, S. Gupta, Z. Song, K. Li, and S. Arora, "Evaluating gradient inversion attacks and defenses in federated learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 7232–7241.

[12] J. Wang, X. Chang, R. J. Rodrìguez, and Y. Wang, "Assessing anonymous and selfish free-rider attacks in federated learning," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2022, pp. 1–6, doi: 10.1109/ISCC55528.2022.9912903.

[13] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[14] J. Chen, M. Li, T. Liu, H. Zheng, Y. Cheng, and C. Lin, "Rethinking the defense against free-rider attack from the perspective of model weight evolving frequency," Jun. 2022, *arXiv:2206.05406*.

[15] J. Bernstein, J. Zhao, K. Azizzadenesheli, and A. Anandkumar, "SignSGD with majority vote is communication efficient and fault tolerant," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–20.

[16] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 9311–9319.

[17] J. Zhang, Y. Wu, and R. Pan, "Auction-based ex-post-payment incentive mechanism design for horizontal federated learning with reputation and contribution measurement," Mar. 2022, *arXiv:2201.02410*.

[18] Z. Sun, P. Kairouz, A. T. Suresh, and H. B. McMahan, "Can You really backdoor federated learning?" Dec. 2019, *arXiv:1911.07963*.

[19] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *Proc. 35th Int. Conf. Mach. Learn.*, Jul. 2018, pp. 5650–5659. [Online]. Available: https://proceedings.mlr.press/v80/yin18a.html

[20] A. Qammar, J. Ding, and H. Ning, "Federated learning attack surface: Taxonomy, cyber defences, challenges, and future directions," *Artif. Intell. Rev.*, vol. 55, pp. 3569–3606, Nov. 2021, doi: 10.1007/s10462-021-10098-w.

[21] Y. E. Sagduyu, "Free-rider games for federated learning with selfish clients in NextG wireless networks," in *Proc. IEEE Conf. Commun. Netw. Security (CNS)*, Oct. 2022, pp. 365–370. doi: 10.1109/CNS56114.2022.9947274.

[22] J. Lin, M. Du, and J. Liu, "Free-riders in federated learning: Attacks and defenses," Nov. 2019, *arXiv:1911.12560*.

[23] Y. Zhan, P. Li, Z. Qu, D. Zeng, and S. Guo, "A learning-based incentive mechanism for federated learning," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6360–6368, Jul. 2020, doi: 10.1109/JIOT.2020.2967772.

[24] Y. Zhan, J. Zhang, Z. Hong, L. Wu, P. Li, and S. Guo, "A Survey of incentive mechanism design for federated learning," *IEEE Trans. Emerg. Topics Comput.*, vol. 10, no. 2, pp. 1035–1044, Apr.–Jun. 2022, doi: 10.1109/TETC.2021.3063517.

[25] J. Huang, R. Talbi, Z. Zhao, S. Bouchenak, L. Y. Chen, and S. Roos, "An exploratory analysis on users' contributions in federated learning," in *Proc. 2nd IEEE Int. Conf. Trust, Privacy Security Intell. Syst. Appl. (TPS-ISA)*, 2020, pp. 20–29, doi: 10.1109/TPS-ISA50397.2020.00014.

[26] Z. Liu, Y. Chen, H. Yu, Y. Liu, and L. Cui, "GTG-Shapley: Efficient and accurate participant contribution evaluation in federated learning," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 4, p. 60, May 2022, doi: 10.1145/3501811.

[27] H. Lv et al., "Data-free evaluation of user contributions in federated learning," in *Proc. 19th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, 2021, pp. 1–8, doi: 10.23919/WiOpt52861.2021.9589136.

[28] X. Bao, C. Su, Y. Xiong, W. Huang, and Y. Hu, "FLChain: A blockchain for auditable federated learning with trust and incentive," in *Proc. 5th Int. Conf. Big Data Comput. Commun. (BIGCOM)*, Aug. 2019, pp. 151–159, doi: 10.1109/BIGCOM.2019.00030.

[29] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *Proc. 23rd Int. Conf. Artif. Intell. Stat.*, Jun. 2020, pp. 2938–2948. [Online]. Available: https://proceedings.mlr.press/v108/bagdasaryan20a.html

[30] F. Sattler, S. Wiedemann, K.-R. Müller, and W. Samek, "Robust and communication-efficient federated learning from non-i.i.d. data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3400–3413, Sep. 2020, doi: 10.1109/TNNLS.2019.2944481.

[31] C. Fung, C. J. Yoon, and I. Beschastnikh, "The limitations of federated learning in Sybil settings," in *Proc. 23rd Int. Symp. Res. Attacks, Intrusions Defenses (RAID)*, 2020, pp. 301–316.

[32] X. Cao, M. Fang, J. Liu, and N. Z. Gong, "FLTrust: Byzantine-robust federated learning via trust bootstrapping," in *Proc. ISOC Netw. Distrib. Syst. Security Symp. (NDSS)*, 2021, pp. 1–18.

[33] Y. Jiang et al., "Model pruning enables efficient federated learning on edge devices," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Apr. 25, 2022, doi: 10.1109/TNNLS.2022.3166101.

[34] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[35] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Dept. Comput. Sci., Univ. Toronto, ON, Canada, Rep. TR-2009, 2009.

[36] V. Shejwalkar, A. Houmansadr, P. Kairouz, and D. Ramage, "Back to the drawing board: A critical evaluation of poisoning attacks on production federated learning," in *Proc. IEEE Symp. Security Privacy (SP)*, 2022, pp. 1354–1371, doi: 10.1109/SP46214.2022.9833647.

[37] Y. LeCun et al., "Handwritten digit recognition with a back-propagation network," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 2, 1990, pp. 396–404.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.

[39] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Math. Program.*, vol. 45, no. 1, pp. 503–528, Aug. 1989, doi: 10.1007/BF01589116.

Jianhua Wang received the B.S. and M.S. degrees in software engineering from Taiyuan University of Technology, Taiyuan, China, in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree in cyberspace security with Beijing Jiaotong University, Beijing, China.

His research interests include adversarial machine learning and federated learning.

Xiaolin Chang (Senior Member, IEEE) received the Ph.D. degree in computer science from Hongkong University of Science and Technology, Hongkong, in 2005.

She is a Professor with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. Her current research interests include cloud-edge computing, cybersecurity, and secure and dependable in machine learning.

Vojislav B. Mišić (Senior Member, IEEE) received the Ph.D. degree in computer science from the University of Belgrade, Belgrade, Serbia, in 1993.

He is a Professor of Computer Science with Ryerson University in Toronto, Toronto, ON, Canada. He has authored or coauthored six books, 20 book chapters, and over 280 papers in archival journals and at prestigious international conferences. His research interests include performance evaluation of wireless networks and systems and software engineering.

Prof. Mišić serves on the editorial boards of IEEE TRANSACTIONS ON CLOUD COMPUTING, *Ad hoc Networks*, *Peer-to-Peer Networks and Applications*, and *International Journal of Parallel, Emergent and Distributed Systems*. He is a member of ACM.

Jelena Mišić (Fellow, IEEE) received the Ph.D. degree in computer engineering from the University of Belgrade, Belgrade, Serbia, in 1993.

She is a Professor of Computer Science with Ryerson University in Toronto, Toronto, ON, Canada. She has published papers in archival journals and at international conferences in the areas of wireless networks, in particular wireless personal area network and wireless sensor network protocols, performance evaluation, and security.

Dr. Mišić serves on editorial boards of IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, *Computer Networks*, *Ad hoc Networks*, *Security and Communication Networks*, *Ad Hoc and Sensor Wireless Networks*, *International Journal of Sensor Networks*, and *International Journal of Telemedicine and Applications*. She is a member of ACM.

Yixiang Wang received the B.S. degree from Beijing Jiaotong University, Beijing, China, in 2018, where he is currently pursuing the Ph.D. degree with Beijing Key Laboratory of Security and Privacy in Intelligent Transportation.

His research interests include adversarial examples and security in machine learning.