Received: 26 July 2024 | Revised: 16 December 2024 | Accepted: 6 March 2025 | Published online: 20 March 2025

RESEARCH ARTICLE

Journal of Data Science and Intelligent Systems
2025, Vol. 00(00) 1–12
DOI: 10.47852/bonviewJDSIS52023970

BON VIEW PUBLISHING

A Study on the Efficiency of Combined Reconstruction and Poisoning Attacks in Federated Learning

Christian Becker[1], José Antonio Peregrina[1], Frauke Beccard[2], Marisa Mohr[2] and Christian Zirpins[1,*]

[1]*Karlsruhe University of Applied Sciences, Germany*

[2]*Inovex GmbH, Germany*

Abstract: Federated learning (FL) is an approach to enable the training of shared Machine learning (ML) models on private data of multiple independent participants. While FL greatly improves data privacy, it also yields a multitude of new threats, such as the combined reconstruction and poisoning attack (CRPA). CRPA allows any FL participant to manipulate the central model, causing it to misclassify arbitrary class combinations. Any participant may reconstruct samples from an arbitrary unknown class and consequently use these samples to deceive the central model into misclassifying it for another class. Previously, proposed attack scenarios only featured a single class combination as well as a single static data distribution. However, in realistic scenarios, the adversary cannot know which class combinations are vulnerable to CRPA and must adapt to the existing data distribution among the participants. In this paper, we answer the question of whether CRPA is influenced by these two parameters and observe the attack accuracy. To this end, the efficiency of CRPA is examined, focusing on the vulnerability of all class combinations, the effect of unbalanced data distribution, and the application of differential privacy (DP). We show that the success of the attack for the MNIST and FMNIST datasets is strongly influenced by the selected class combination as well as the underlying data distribution, with differences of up to 46% in attack accuracy in the worst case. Additionally, we were able to prevent the reconstruction of human-identifiable images with DP, which in our case also mitigated the poisoning attack. Our results indicate that the success of reconstruction and poisoning attacks diminishes in scenarios with an unbalanced data distribution among participants and that DP can be an effective defense against the combined attack in any type of scenario.

Keywords: federated learning, reconstruction, poisoning, generative adversarial network, differential privacy

1. Introduction

Federated learning (FL) is a promising method to train machine learning (ML) models on private data by exchanging ML models instead of the data. This allows collaborative training of models on private data silos and opens the door to a multitude of new areas of application, which previously were inaccessible, like training models with data from multiple small devices without having to transfer the data from them [1, 2]. Participants engage in FL to obtain ML models with the capability of generalizing over different data silos without requiring explicit sharing of such data. The promise is that no third party has access to the private data and no sensitive information is revealed. However, recent research has shown that the promise of privacy and security in FL is not always justified [3, 4]. That is because it is possible to recreate samples similar to the contributed private data. Moreover, these samples can be used to poison the model [5–7]. The applicability and implications of this severe issue for privacy and security have not yet been fully studied.

Our work examines the applicability of the combined reconstruction and poisoning attack (CRPA) with respect to class selection, unbalanced class distribution, and prevention techniques. We demonstrate that these factors do influence the results of the reconstructed sample as well as the success of the poisoning attack. Although sensitive data in FL remains at the participants' locations, information about it is learned and forwarded through the collaboratively trained global model. CRPA consists of two parts: the reconstruction of samples from the private distribution as well as the poisoning of the central model. Reconstruction aims to recover information about the private data in FL. In our work, a generative adversarial network (GAN) is used to this end. Poisoning aims to purposely mislabel training samples to manipulate the global model into misclassifying a specific class for another. To execute the poisoning attack, the adversarial client is required to hold samples of the to-be misclassified class. The combination of reconstruction and poisoning attacks enables adversaries to target classes for which they do not hold samples by generating them. This leads to the possibility of attacking any class.

Previous work has focused on targeting a single class combination within a single static class distribution among the participants in FL [5–7]. However, in a realistic case of CRPA, the adversarial participant must choose which combination to target. This decision is likely made without knowing the most vulnerable combinations. Additionally, an attacker must adapt to the existing data distribution, which will most likely be unbalanced in realistic cases.

*Corresponding author: Christian Zirpins, Karlsruhe University of Applied Sciences, Germany. Email: zich0001@h-ka.de

Our hypothesis is that class selection, data distribution, and application of common defense measures like differential privacy (DP) are key factors affecting the performance of CRPA. Therefore, a study considering these factors would contribute to a better understanding of CRPA in real-world scenarios. This could improve the ability to analyze the threats of CRPA for concrete applications and the assessment of DP as a countermeasure. To this end, our work examines how the targeted class and the underlying data distribution influence the success of CRPA. Furthermore, we show that CRPA can be prevented using DP [8].

To show that, we have examined the vulnerability for all possible class combinations for the MNIST and FMNIST datasets by observing the average attack accuracy (AAA). This selection of datasets, although limited, provides a solid set of initial results, with a robust methodology that can be applied in future work with more complex images or a higher number of class combinations. To determine which distributions contribute to the success of CRPA, several degrees of class distributions using the Dirichlet distribution [9] and their effect on the attack accuracy are examined. Finally, we show that DP can prevent CRPA in our case by dropping AAA close to zero, though at the cost of reduced global model accuracy. In summary, the paper provides the following contributions:

1) First, we provide a fundamental study that explores the threats of CRPAs (CRPA) to FL-based systems considering practically relevant factors including class-specific vulnerability and different types of federated data distribution. The study enables readers to understand the implications of the CRPA pattern for their own FL processes and conduct similar experiments to assess their security.
2) Second, we present a follow-up study that explores the effectiveness of DP as a security measure to counteract CRPAs under practically relevant conditions, thereby supporting readers in developing strategies to explicitly secure their threatened FL processes.

Our findings demonstrate that several factors can influence the success of CRPAs. The selection of specific classes for reconstruction and poisoning can determine whether an attack succeeds or fails. Moreover, data distribution among FL participants may hinder poisoning while leaving the reconstruction process unaffected. Finally, contrary to prior research [10], our findings suggest that DP can prevent the reconstruction attack, thereby rendering the poisoning attack ineffective. Overall, we believe that our work provides valuable insights for practitioners aiming to enhance FL security and privacy. Additionally, our methodology and findings can inspire further research on more complex image datasets, reconstruction attacks capable of bypassing DP, or defenses that preserve training performance better than DP.

In the following, Section 2 introduces reconstruction and poisoning attacks as well as their combination. Next, Section 3 references previous approaches and their choices of class selection, class distribution, and possible defenses. Then, Section 4 shows why class selection and data distribution influence CRPA. The setup of the experimental framework and the settings for the executed experiments are explained in Section 5. The results of these are presented and discussed in Section 6. Finally, Section 7 summarizes the results and provides an outlook.

2. Background

The following section presents the concepts underlying our work. Subsection 2.1 presents FL, describing the steps of the process, and the vulnerabilities that can arise from it. Subsections 2.2 and 2.3 introduce *reconstruction attacks* that harm the privacy of participants by reconstructing training data, and *poisoning attacks* that affect model performance by tampering with the training data. Finally, subsection 2.4 explains the combination of the two attacks.

2.1. FL

FL [11] is a novel approach whose application yields a multitude of new attack vectors [3, 4]. An FL session thereby consists of multiple steps, which take place at either the central curator or the participants. Initially, the curator must set up the global model with an architecture suitable to solve the ML task. Then, the actual federated training starts, which repeats three steps: First, a copy of the global model is distributed from the curator to all participants. In the second step, the local training takes place, where each participant trains its copy of the global model on its private data. This yields an updated model for each participant that has learned the patterns and regularities from its private data. In the third step, the curator collects the model updates from all participants and aggregates them to the new global model. After multiple federated training rounds, the global model learns the patterns and regularities of all the private data, and the FL session ends. Thereby participants rely on the assumption that their private data is protected from third-party access and that the resulting model is reliable and trustworthy. Nevertheless, further research has identified additional threats to both the privacy of data and the reliability of the global model. An additional factor influencing the model quality is the data distribution [9]. A balanced distribution, where each class is represented equally often at each participant, is used in scientific setups. However, one finds an uneven distribution of data in most realistic FL, which raises the question of how this influences FL beyond its impact on model accuracy.

2.2. Reconstruction attack

In the **reconstruction attack**, any adversarial participant might reconstruct samples similar to the private training data of other participants [10]. Our work focuses on reconstruction attacks utilizing a GAN that allows the generation of samples from an arbitrary class. A GAN consists of two neural networks, the generator and discriminator network that are jointly trained to learn to generate data from the distribution of a training dataset. The generator network learns to generate samples with the help of the discriminator network as a teacher that judges whether samples originate from the training data distribution. In the reconstruction attack, the generator part of the GAN is controlled by the adversarial participant, while the global model is used as a discriminator. In the local training runs, the benign participants train the global model on their local data. The adversarial participant not only trains on its local data but also trains the generator network using the global model as the discriminator. The training of the generator is repeated in the FL process. Consequently, the global model learns the patterns of all participants' private data, assisting the generator model in learning to regenerate samples of other participants' private training data.

2.3. Poisoning attack

Poisoning attacks aim to compromise the reliability of the model by deceiving it into misclassifying samples. The adversarial participant achieves this by purposely mislabeling its training data during FL [3]. Poisoning can cause samples to be misclassified randomly (untargeted) or focus on a specific source class (targeted). In this work, poisoning attacks refer to the targeted version. When participating in FL training, the adversarial client

teaches the global model to misclassify the targeted classes by poisoning its own training data. The resulting global model is likely to misclassify samples of the source class as the target class.

2.4. Combined reconstruction and poisoning

The goal of the **CRPA** is to poison a specific class unknown to the adversarial participant, as described by Zhang et al. [7]. Unknown in this context means that the adversarial participant does not possess samples or have information about this class.

The reconstruction attack assumes an adversarial client A, who possesses data relevant to the FL setup. This allows the adversarial client to legally participate in the training process. In addition to the normal training, it aims to generate instances of a class G for which it possesses no samples and about which it has no prior knowledge. The goal of A is to generate samples of the class G and trick the global model into classifying them as another target class T. This is achieved by applying a poisoning attack on the global model.

At the beginning of FL, the adversarial client submits an additional label to the label voting. This will lead to an extra output in the global model, which is used as the fake label required to train the generator model. The adversarial client then needs the following additional capabilities in its local training process to execute the reconstruction attack: It must be able to intercept the global model, change the global model parameters, and manipulate its private training data in the FL process.

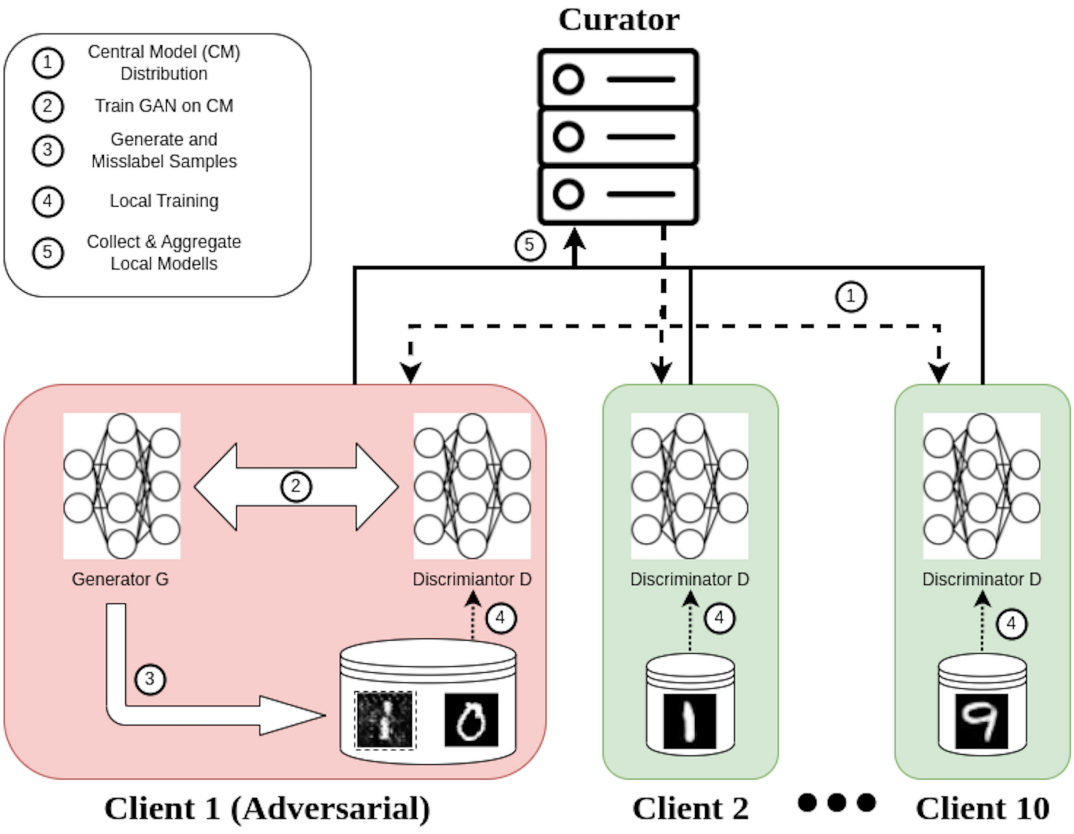
We consider an approach that combines reconstruction and poisoning attacks. Figure 1 [7] shows the process of this attack scheme in five steps. In the initial step 0, the adversarial participant submits an additional label during the label voting stage. In step 1, the global model is distributed to all participants. Steps 2 and 3 take place on the adversarial client A exclusively. In step 2, A trains the generator network on the global model learning to produce samples of class G. In step 3, samples of the class G are generated and mislabeled with the labels of the target class T. The mislabeled data are added to the training data of A. In step 4, the local training takes place, where all participants train their copy of the global model on their local dataset. A thereby uses the modified dataset, poisoning the model in its local training. Finally, in step 5, all models are collected and aggregated by the curator.

This approach has advantages for adversaries, which makes it a powerful attack. First, it does not require the adversarial participant to possess samples of the classes it aims to attack. This is an advancement compared to general poisoning attacks, which always require access to samples of the target class. In addition, these samples can be used to observe the reconstructed characteristics of an unknown class and thereby leak private information, but this is outside the scope of this work. Second, only a single adversarial client is necessary to perform this attack, and it does not need any additional capabilities in the FL setup except sufficient computational power to train the GAN.

3. Related Work

The following section presents existing work related to our study. Subsections 3.1 and 3.2 list work on reconstruction and

Figure 1
CRPA using a GAN. The adversarial client reconstruct samples of class 1, which is unknown to it, and mislabels them as class 9 to poison the global model



poisoning attacks. Subsection 3.3 outlines work regarding the combination of both reconstruction and poisoning attacks. Finally, subsections 3.4 and 3.5 present work on defenses employed against reconstruction and poisoning attacks.

3.1. Reconstruction attack in FL

Reconstruction attacks in FL based on GANs were introduced by Hitaj et al. [10], who also compared them to another reconstruction method called model inversion. They show that the GAN reconstruction attack is far superior in a federated setting. Additionally, they were able to reconstruct human-identifiable images despite using DP at the record level. An alternative approach is to execute the reconstruction attack from the central curator instead of an adversarial client as mentioned by Wang et al. [5]. Their approach is based on an adversarial curator, rather than an adversarial client, which utilizes a GAN to reconstruct samples. Neither investigates the influence of class selection or data distribution, and both consider only a single static class combination and data distribution.

3.2. Data poisoning attacks on FL

Data poisoning attacks in FL and their effects were investigated by Tolpegin et al. [12]. They show that the attack is highly effective even with a minor proportion of adversarial clients. Their attack setup uses only a balanced data distribution, and they do not investigate the effects of other distributions. Their work is important, but it does not provide a comprehensive view of the effectiveness of poisoning attacks on specific classes. They conclude that there is not necessarily a correlation between non-poisoned misclassification performance and attack effectiveness.

3.3. CRPA

The CRPA was introduced by Zhang et al. [7]. It features a GAN-based reconstruction attack combined with a data poisoning attack. They show that a CRPA can be very effective in both reconstructing samples and poisoning the global model. They further enhanced their attack scheme by including backdoor attacks [6]. However, all previous approaches are executed for a single class combination, with a single static data distribution, and they are not tested against DP.

3.4. Defenses against reconstruction attacks

Defenses against reconstruction attacks are mainly based on noising the model gradients or parameters, which avoids the reconstruction of images that have similarity to those of the private data. In their initial paper on the reconstruction attack, Hitaj et al. successfully reconstruct images despite the application of added noise using DP [10]. Subsequently, they state that DP is not a countermeasure against GAN reconstruction attacks because as long as the accuracy of the global model improves, the generator model can learn from the distribution of the discriminator model. Ziegler et al. state that the reconstruction of images using the gradient leakage attack could be prevented using different levels of DP [13]. Another less generic approach is to obfuscate the private training data in order to prevent reconstruction [14]. However, this technique requires a sufficient number of training data to generate the shadow dataset, which is not always available for each participant

3.5. Defenses against poisoning attacks

Defenses against poisoning attacks are based on anomaly detection and rely on the test accuracy, model updates, or gradient updates sent to the central curator [12, 15]. The analysis of test accuracy and model updates is intended to exclude adversarial clients but requires a balanced data distribution. In scenarios with unbalanced distributions, it would falsely detect honest participants as outliers and exclude them from training. The MUD-HoG approach [16] manages to detect poisoning clients also in unbalanced setups by assuming access to the gradient updates of the participants.

All protections against model poisoning require that the central curator owns a sufficient test dataset or has access to either the parameters or gradient updates of the participants. However, these requirements are not always met in all FL setups, and some aggregation methods, such as SMPC, specifically avoid sending parameter or gradient updates to the central curator to protect them from inference attacks such as deep leakage from gradients [17].

Previous work has also investigated the effectiveness of DP against poisoning attacks in FL [18, 19]. They showed that DP cannot protect the model against poisoning attacks, but stricter privacy guarantees do lead to less successful attacks. Therefore, DP increases robustness against poisoning, but it cannot fully prevent it.

4. Applicability of CRPA

Previous work has shown that the CRPA can be highly effective. However, we argue that its success strongly depends on (a) the targeted classes, (b) the underlying data distribution, and (c) the presence of countermeasures. In the following, we discuss the state of research and the remaining research gaps that our work addresses in these three areas.

a) **Class-Specific Vulnerability:** Selecting a class combination for the CRPA affects the difficulty of reconstructing a sample and the success of the poisoning attack. This can be shown for both attacks. A well-known phenomenon called *mode collapse* shows that a GAN tends to use the simplest solution of producing samples that belong to only a single or very few classes [20]. When dealing with classes containing multiple sample variations, such as a class with multiple dog breeds, the GAN may collapse into generating only a single variant or breed of a specific class. This shows that a GAN tends to learn only the simplest representation of its respective task.

Regarding poisoning attacks, research has shown that their effectiveness depends on the specified class targeted for the attack [21]. This is the case because a mislabeled sample X might be more likely to be misclassified as class A than as class B. The likelihood that a sample from class X is misclassified as target class B depends on several factors, such as feature similarity, overall class similarity, and the distribution of the remaining classes. Despite that, it can be observed that attacks on certain classes tend to be more effective than others. The number of possible combinations that the attack can choose from to reconstruct and poison the data increases with the number of available classes. Furthermore, the effect of each combination is different for each dataset. Thus, it is not possible to answer the question of which combinations are most successful in general, but only within each dataset. Therefore, we focus on studying the effects of all combinations of the MNIST and FMNIST datasets.

Although the related studies of Zhang et al. use the same datasets [6, 7], they do not explain how they chose their combinations. It is possible that they picked them randomly or selected those yielding better results. Tolpegin et al. [12] did indeed explain that they look into how likely two samples were misclassified to choose the attack. Nevertheless, their selection does not seem to correlate with the misclassification results. Therefore, a study exploring more combinations could provide valuable insights into the problem. To shed light on this open question, we study the success of the CRPA for each possible class combination in the MNIST and FMNIST datasets.

b) **Class Distribution Influence**: In ML and FL, the performance of a trained model strongly depends on the underlying training data distribution. A strong factor is the class balance of the underlying dataset [9]. Experiments in FL, as well as its attack scenarios, are mostly based on a balanced data distribution. However, in many real-world scenarios, the distribution of classes is unbalanced [11]. GANs tend to produce samples from variations that are more often represented in a dataset [22]. For example, when training a GAN on a dataset that includes pictures of cats from different breeds (variations), it is more likely to produce samples of those breeds that are more frequent in the dataset. Thus, an unbalanced variation within a dataset or a specific class leads to fewer variations in the generated samples. In other words, it follows that producing samples of more frequently represented classes or variations is easier.

The class distribution strongly influences how accurate the global model is in learning to classify each class. A more balanced distribution thereby leads to a more accurate prediction because the local models are not skewed towards the underlying training data. The more unbalanced the local training data becomes, the more skewed toward the majority class the resulting model will be. This will also lead to a skew in the global model when aggregating the local models. Consequently, data distribution directly influences the effectiveness of the targeted poisoning attack. The adversarial participant tries to trick the global model into misclassifying a class G as a class T. It does so by mislabeling the local data accordingly and thus teaches it a wrong classification task on purpose. This results in an artificially skewed model, which is then aggregated with other models, which may also be skewed due to the unbalanced data distribution. This way the artificial skew of the adversarial model is influenced by the class distribution.

The CRPAs presented in the related work [6, 10] only consider a static data distribution, which is a balanced distribution across participants. As the effects of a class imbalance are well-known in FL, the exploration of the attack under a variety of such scenarios seems highly relevant. We fill this gap by focusing on how class distributions among participants affect the CRPA.

c) **Effectiveness of DP:** While DP was explored by Hitaj et al. [10], its effectiveness was only investigated for the reconstruction side of the attack and within a limited set of scenarios. Under the CRPA, DP can also be effective as a defense against the poisoning attack. This is because DP can make the GAN generate samples with more noise, which in turn could reduce their effectiveness in poisoning the model. This could result from missing key features in the generated samples. Additionally, unbalanced distribution scenarios remain underexplored in this regard. For this reason, we aim to explore how effective DP can be in this expanded set of scenarios.

5. Experimental Setup

This section introduces the experimental setup used to measure the influence of class selection, class distribution, and DP on the CRPA. We also describe the framework and settings used for the experiments.

5.1. Datasets and models

For our study, we have executed experiments on the MNIST [23] and Fashion MNIST (FMNIST) datasets [24]. Both include grayscale images of ten different classes with a resolution of 28×28 pixels. The MNIST dataset features handwritten numbers from 0 to 9. The FMNIST dataset features images of ten different fashion objects of the following categories: t-shirt/top, trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot. Both datasets are divided into a set for training and another set for testing, each combination featuring 60,000 and 10,000 samples. To achieve a balanced class distribution in the MNIST dataset, each class is cut down to the size of 5,000 samples. For better comparability, the same is done for the FMNIST dataset, although it already has a balanced distribution. As a result, there are a total of 50,000 samples in each training set featuring 5,000 samples per class. To show the effects of class selection, class distribution, and DP on the CRPA, we utilized the open-source implementation provided by Jaskiee[1]. It utilizes the Keras library [25] and is based on the initial reconstruction attack proposed by Hitaj [10]. We extended the implementation to generate and poison arbitrary classes, create alternative data distributions, and apply DP mechanisms during the training process by leveraging the TensorFlow Privacy library [26].

Executing FL with the reconstruction attack requires two neural networks with distinct roles. The first network is trained and shared in the FL, serving as both the global model and discriminator. The second network, located at the adversarial client, reconstructs images of other classes and is called the generator. The architecture of both models is specified in Figure 2.

We provide the full implementation of our experiments by means of a Git repository on GitHub[2]. We invite readers to explore the code to gain deeper insights, replicate our findings, or extend the research through additional experiments.

5.2. Experiment settings

We have carried out three experiments, each consisting of multiple FL runs with a different class combination, class distribution, or the application of DP. For better comparability, each experiment builds on the results of the previous one under different conditions.

The experiments are compared by calculating the *AAA*, denoted as $\overline{A}$, across the entire training period of an experiment. In this approach, samples of the class G are generated and mislabeled as a target class T. The attack accuracy A is the number of correctly misclassified samples n_T^G from class G as T, divided by the total number of samples in G, denoted as N^G. It is calculated using the global model w_r for each global training round r.

We calculate the *AAA* values for various class combinations, denoted $\overline{A}_T^G$, across all global training rounds R, as shown in Equation (1). This is done because the attack accuracy is not strictly increasing but fluctuating during the training process (depicted in Figure 3). We assume that the exchange of weights leads to an alternating learning and unlearning cycle, resulting in this pattern.

[1] https://github.com/Jaskiee/GAN-Attack-against-Federated-Deep-Learning
[2] https://github.com/fed-crpa/fl_crpa

Figure 2
Global (discriminator) model architecture (left) and generator model architecture (right), derived from the Keras model summary

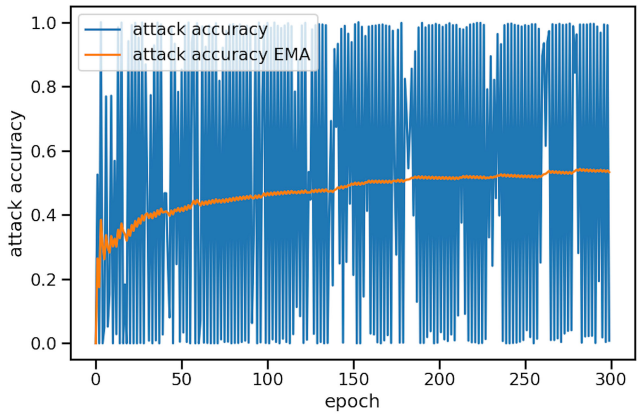
```
Global model architecture

==============================
Conv2D       (None, 14, 14, 64)
LeakyReLU    (None, 14, 14, 64)
Dropout      (None, 14, 14, 64)
Conv2D       (None, 7, 7, 128)
LeakyReLU    (None, 7, 7, 128)
Dropout      (None, 7, 7, 128)
Flatten      (None, 6272)
Dense        (None, 11)
```

```
Generator model architecture

==================================
Dense            (None, 12544)
BatchNorm        (None, 12544)
ReLu             (None, 12544)
Reshape          (None, 7, 7, 256)
Conv2DTranspose  (None, 7, 7, 128)
BatchNorm        (None, 7, 7, 128)
ReLu             (None, 7, 7, 128)
Conv2DTranspose  (None, 14, 14, 64)
BatchNorm        (None, 14, 14, 64)
ReLU             (None, 14, 14, 64)
Conv2DTranspose  (None, 28, 28, 1)
```

Figure 3
Attack accuracy for an experiment performed on the MNIST dataset, showing fluctuations over time. The exponentially weighted moving average (EMA) indicates a steady increase in attack accuracy throughout the training process



$$\overline{A}_T^G = \frac{1}{R}\sum_{r=0}^{R-1}\frac{n_T^G}{N^G} \; with \; w_r \tag{1}$$

Each run is executed three times. Then, both $\overline{A}$ and the standard deviation σ are calculated across all runs, following a similar procedure to that of Tolpegin [12]. When interpreting the *AAA*, the worst case occurs when the attack accuracy switches between 0 and 1.0 in each training round, similar to a sawtooth pattern. In such a pattern, an average of 0.5 is considered a good result, since the maximum achievable mean is 0.5.

5.3. Vulnerability of class combinations

To answer the question of whether all classes are equally vulnerable to the CRPA, we tested all possible combinations. Typically, attackers must decide which combination of reconstruction and target poisoning classes to attack. We conducted FL experiments with 10 participants, where each class's samples were exclusively assigned to one participant. A single adversarial participant then executed the attack on the specified combination, ensuring that this participant held neither samples of the generator class G nor of the target class T. Each FL run was performed for all possible combinations over 300 global training rounds until the generated images stopped improving in quality, and no further improvement in the attack accuracy *overlineA* was achieved. This yielded a total of $10 \times 10 - 10 = 90$ combinations, excluding cases where the same class was used for both generation and poisoning.

5.4. Class distribution influence

We assess how the underlying class distribution affects the *AAA* by examining various distribution scenarios. Classes are distributed among the 10 participants according to the Dirichlet distribution parameterized by α [9]. A distribution value of 0.0 indicates a completely isolated distribution, while higher values correspond to more balanced distributions determined by the Dirichlet parameter α. Specifically, a distribution value of 0.0 denotes a fully unbalanced setting, where each client holds samples from only one class, as in the previous experiment. Higher distribution values correspond to a more balanced class distribution. To ensure that the adversarial participant held no samples from the generator class G or the target class T, we sampled repeatedly from the distribution and selected only those meeting this condition. We adopted the distribution values $\alpha = \{0.0, 0.05, 0.1, 0.2, 0.5, 1.0, 10.0\}$ from Hsu's approach [27], covering both balanced and unbalanced distributions.

We repeated the FL runs for three combinations: the least vulnerable, the most vulnerable, and a moderately vulnerable one. Thus, we selected the combinations with the lowest and highest $\overline{A}$ values, as well as the one closest to the mean $\overline{A}$ from each dataset. We evaluated each of the three combinations using all seven distribution values, yielding meaningful insights into how data distribution influences CRPA across three distinct vulnerability levels.

5.5. Application of DP

We evaluated the influence of DP by applying it at the record level, following the approach of Hitaj et al. [10]. In our implementation, we used the DP-SGD optimizer from TensorFlow Privacy [26]. Based on the official documentation, we configured the following parameters:

1) The *noise_multiplier* parameter was reduced from 1.3 to 0.3 to ensure FL convergence by strongly reducing the applied noise.
2) The *num_microbatches* parameter was reduced from its default value of 250 to 1, ensuring DP is applied at the record level rather than across a set of gradients.
3) The *l2_norm* and *learning_rate* parameters remained unchanged, as they do not directly affect the privacy guarantee. According to the documentation, their respective values are 1.5 and 0.25.

We repeated the previous experiments with DP applied, using distribution values of 0.0 and 10.0. These α values were chosen to evaluate DP's effectiveness in both unbalanced and balanced class distributions.

6. Results and Discussion

To shed light on the effectiveness of the CRPA, we present and discuss the results of multiple experiments in this section. First, we assess the vulnerability of class combinations for both datasets. Second, we examine the influence of different class distributions on the CRPA success. Third, we analyze whether DP can prevent the CRPA.

6.1. Vulnerability of classes

An attacker must decide which combination of reconstruction class and target poisoning class to select for the CRPA. To determine whether all classes are equally vulnerable to the CRPA, we compare the *AAA* of all combinations, as explained in Section 5. In the following, the term $\overline{A}_9^4$ denotes the *AAA* of a FL run that generates samples of class *4*, mislabels them with label *9*, and has a standard deviation of σ_9^4. The experiments are limited to the MNIST and FMNIST datasets, as no general assumption can be made for all datasets. However, they provide a solid methodology that can be extended for further experiments on other datasets.

Figure 4 presents the results using a combination matrix showing the *AAA* for all combinations of the MNIST and FMNIST datasets. The y-axis represents the generator label, while the x-axis denotes the poisoned label. The higher the *AAA*, the more successful the attack on the corresponding label combination. Reconstructing and poisoning the same label does not result in poisoning. Therefore, these combinations are omitted, and the main diagonal is always zero. An *AAA* of 0.5 is considered a very good result in our experiments, as explained in Section 5. We do not delve into the details of each specific FL run but instead highlight the most important findings.

First, we discuss the results for the MNIST dataset in the combination matrix on the left of Figure 4. It shows that the attack's effectiveness ranges from very successful, with a maximum *AAA* of 0.51, to moderately effective, with a minimum *AAA* of 0.16. The mean *AAA* for all runs is 0.33, with a standard deviation of 0.03, indicating low fluctuation in attack accuracy across different runs. The three best-performing class combinations are $\overline{A}_9^4 = 0.51$, $\overline{A}_9^7 = 0.46$, and $\overline{A}_8^4 = 0.44$. The three combinations with the lowest performance are $\overline{A}_1^0 = 0.16$, $\overline{A}_0^9 = 0.19$, and $\overline{A}_1^9 = 0.2$. The two worst-performing combinations both involve class 9 as the generator class, which might suggest that it is highly vulnerable to generator attacks. However, this assumption does not hold, as the row for class 9 does not consistently show significantly lower values. Only four combinations involving generator class 9 show relatively low values, while three others are close to the mean and the remaining combinations perform well.

The findings for the FMNIST dataset are similar to those of the MNIST dataset. They are depicted on the right of Figure 4, showing that the attack's effectiveness ranges from extremely successful, with a maximum *AAA* of 0.55, to minimally effective, with the lowest *AAA* of 0.07. The mean *AAA* for all runs is 0.30, with a standard deviation of 0.02, indicating low fluctuation in attack accuracy across different runs. The three best-performing combinations are $\overline{A}_5^7 = 0.55$, $\overline{A}_9^7 = 0.54$, and $\overline{A}_5^0 = 0.52$. Two of the best combinations involve generator class 7, but the remaining seven combinations in the row are very close to the mean, indicating no particular vulnerability. The three combinations with the lowest performance are $\overline{A}_7^6 = 0.07$, $\overline{A}_0^6 = 0.08$, and $\overline{A}_5^6 = 0.07$. These combinations all involve class 6 as the generator class, suggesting that it might be highly vulnerable to generator attacks. However, this assumption does not hold, as the row for class 6 does not consistently show significantly lower values.

Generally, no significant row or column pattern indicating a particularly vulnerable or resistant generator or poisoning class can be observed in either combination matrix. One might assume that the combinations are generally symmetrical and that the generator and poisoning classes can be interchanged to achieve similar or equal *AAA*. However, this is not the case, as many combinations achieve significantly different *AAA* values compared to their mirrored counterparts. For example, $\overline{A}_9^4$ performs significantly better

Figure 4

Average attack accuracy (AAA) for the combined reconstruction and poisoning attack (CRPA) across all combinations of generator and poisoning labels in the MNIST (left) and FMNIST (right) datasets. Each class in the datasets is assigned to a single participant, resulting in a total of ten participating clients over 300 global training rounds

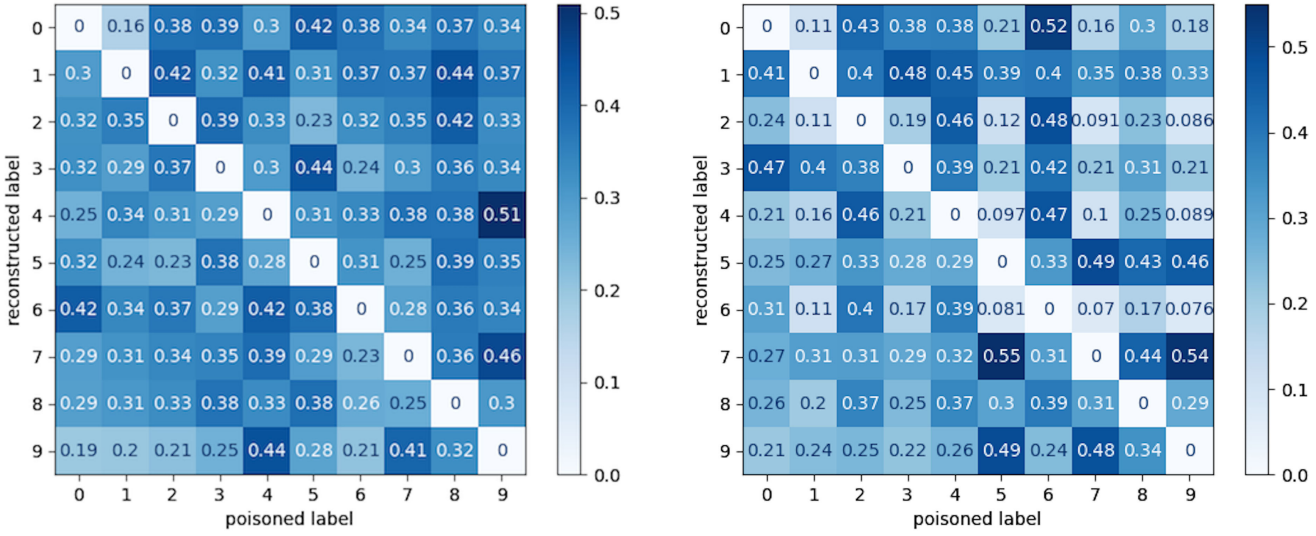
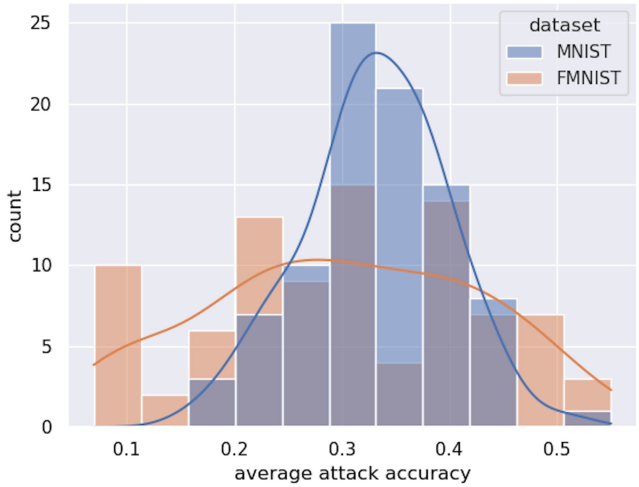


Figure 5
The histogram of the average attack accuracy (AAA) taken from the experiments in Figure 2. It shows that the success of the attacks is distributed around their respective mean for both datasets



than $\overline{A}_4^9$. We believe that the attack's success might depend on whether the reconstructed image possesses unique features compared to the target image. This could hinder the attack, as honest updates might detect features that nullify its effect.

To compare the attack's success on the MNIST and FMNIST datasets, the histogram in Figure 5 shows the distribution of *AAA* values for all tested combinations, as depicted in Figure 4. The attack on MNIST shows a pronounced peak at its mean value of 0.33, with a relatively even distribution around it, skewed toward higher values. A standard deviation of 0.07 indicates a stable distribution of values. Attack accuracies range from 0.2 to 0.45, excluding a single outlier with an accuracy above 0.5. When the outlier is excluded, the smoothed distribution remains centered around the mean but is shifted toward higher average accuracies.

In contrast, attack accuracies on FMNIST are more dispersed around the mean of 0.3, as indicated by a high standard deviation of 0.13. FMNIST attack accuracies range from a minimum *AAA* of 0.0 to values exceeding 0.55. Notably, ten attacks achieved very low *AAA* values below 0.1, while ten combinations recorded exceptionally high *AAA* values above 0.45. Excluding the MNIST outlier, the top FMNIST attacks significantly outperform the best MNIST attacks. Furthermore, FMNIST exhibits extreme cases, ranging from minimal success to high susceptibility to the attack. This comparison highlights the dependency of dataset vulnerability on the chosen attack combination in both cases.

In conclusion, the attacks on MNIST consistently succeed in poisoning the central model. In contrast, attacks on FMNIST are more likely to perform significantly better or worse than their respective mean *AAA*. This demonstrates that the attack's success heavily depends on the dataset and classes selected for poisoning.

6.2. Class distribution influence

We conducted the CRPA using various class distributions to illuminate their effect on both the MNIST and FMNIST datasets. The selected attack combinations include those with the highest, lowest, and closest-to-mean *AAA* values. When selecting the class distribution, it is ensured that the attacking client holds no samples of the attacked generator class. Figure 6 presents the comparison for both datasets, with the MNIST attack results on the left and the FMNIST attack results on the right. The *AAA* indicates the CRPA's success for tested combinations across distributions of $\alpha = \{0.0,\ 0.05,\ 0.1,\ 0.2,\ 0.5,\ 1.0,\ 10.0\}$ as explained in Section 5.

The experiments with the MNIST and FMNIST datasets reveal a clear trend of decreasing *AAA* with a more balanced data

Figure 6
Results of executing the CRPA with different data distributions generated using the Dirichlet distribution for the MNIST (left) and FMNIST (right) datasets

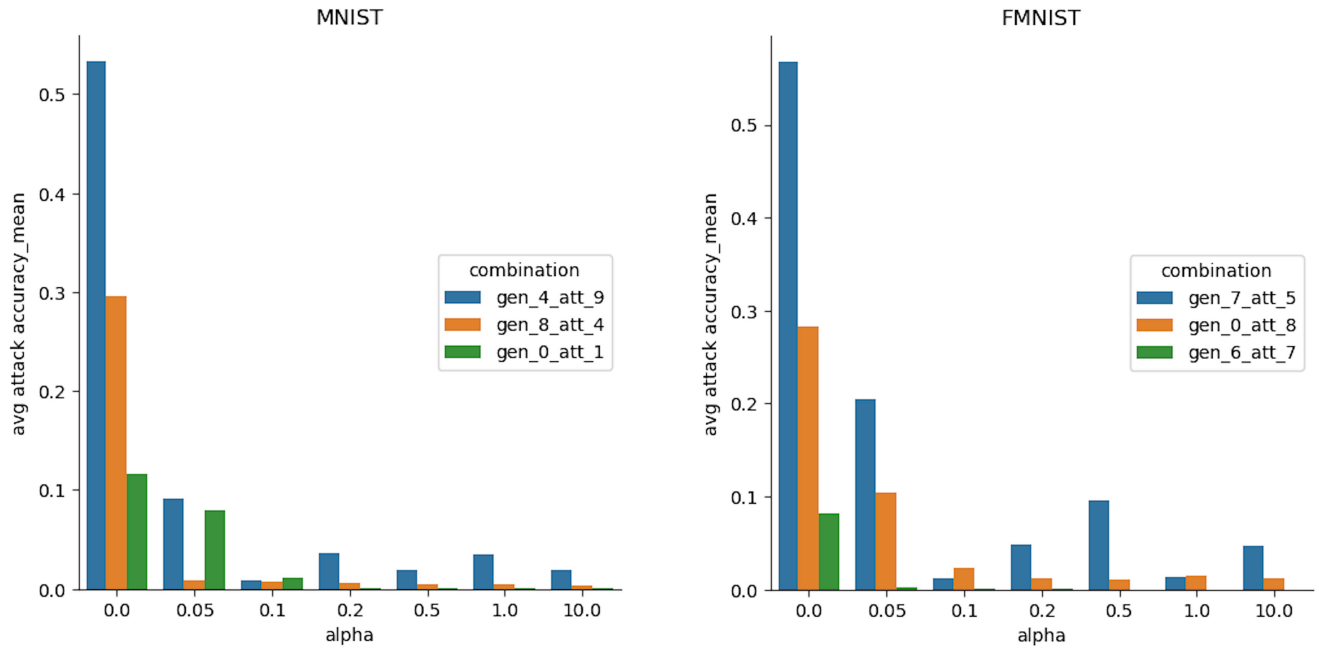
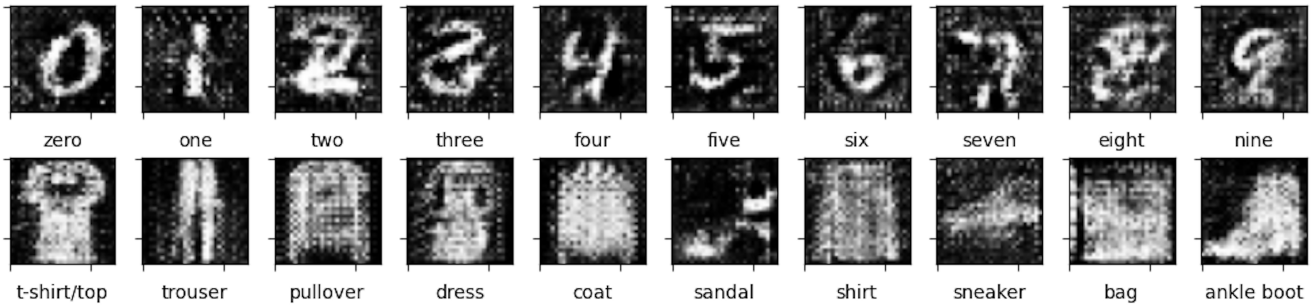


Figure 7
Reconstructed images generated by executing the CRPA on the MNIST (top) and FMNIST (bottom) datasets across different data distributions. Columns represent distribution values from left to right: 0.0, 0.05, 0.1, 0.2, 0.5, 1.0, and 10.0



distribution, as shown in Figure 6. Attacks with a distribution value of 0.05 already exhibit a significant decrease in *AAA* compared to a distribution value of 0.0, dropping below 0.1. Subsequent distributions yield negligible *AAA* below 0.04.

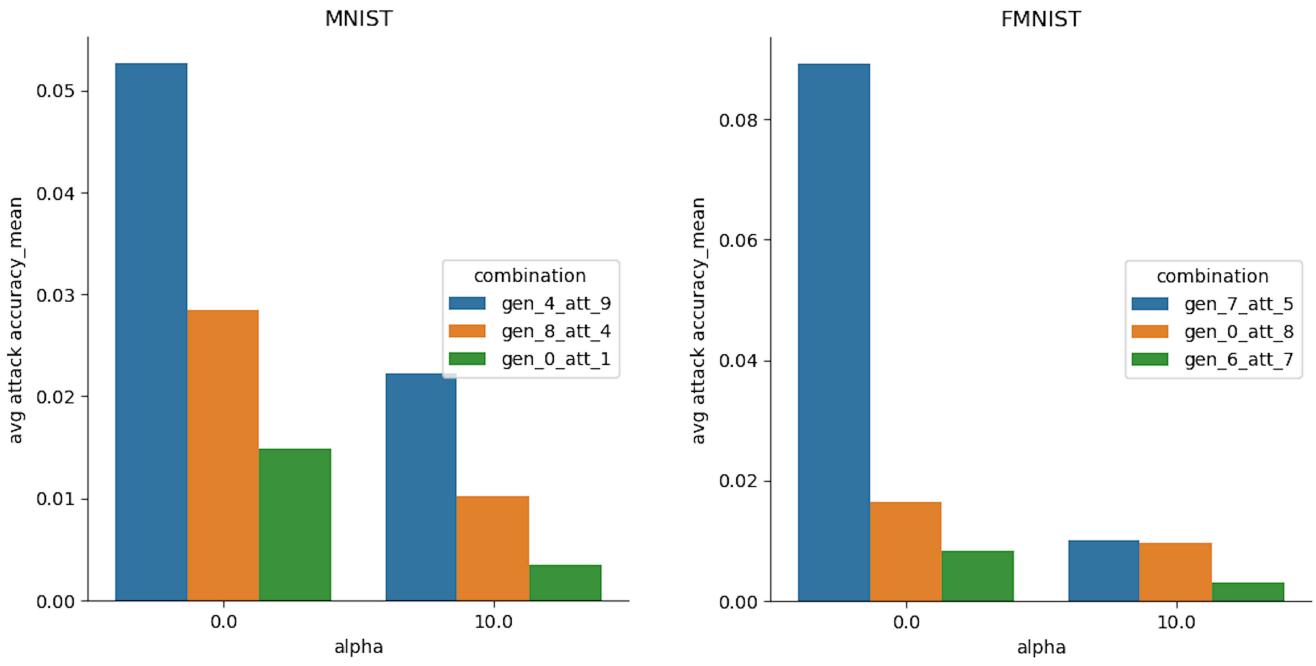
Although the *AAA* value for the combination $\overline{A}_5^7$ in the FMNIST attacks does not strictly decrease for every increment of the distribution value, the results clearly indicate that a more balanced class distribution leads to less attack accuracy. This irregularity is observed at distribution values of 0.2, 0.5, and 10.0, which show an increase compared to preceding distribution values. We assume that this outlier results from the specific class distribution among participants. The presence or absence of specific class samples, particularly for the attacking client but also the other clients, can influence the success of the attack. However, it remains unclear how the presence or absence of specific class samples affects the success of the CRPA. Despite the outliers, the \\overline chart clearly illustrates a significant reduction in *AAA* as the distribution value increases.

In conclusion, a clear trend emerges toward less successful poisoning attacks with more balanced data distributions, as observed in both datasets. The diminishing success of the attack is not attributable to the results of the reconstruction attack. The quality of the recreated samples remains consistent or even improves slightly with more balanced distributions across all combinations, as depicted in Figure 7. We conclude that the CRPA's weakness stems from the poisoning attack. Our experiments demonstrate that a more balanced data distribution enhances the reconstruction attack with GANs but hinders the success of the poisoning attack. This occurs because the skewed representation of the poisoned model update is offset by honest updates. This effect is particularly evident when honest updates identify features that enable the model to differentiate between the targeted and poisoned classes.

6.3. DP

For comparison, we repeated the experiments using data distributions of 0.0 and 10.0 while applying DP. As a brief reminder, an alpha value of 0.0 represents a fully isolated distribution, while a

Figure 8
Results of executing the CRPA with differential privacy for the MNIST (left) and FMNIST (right) datasets. The results show that the effect of the CRPA is practically negligible due to the application of DP



value of 10.0 indicates that all classes are present across all participants. The DP settings remain the same as described in Section 5.

Figure 8 presents the resulting *AAA* values from experiments involving DP for both datasets. The *AAA* values for the two most vulnerable combinations in both datasets reach only 0.07 and 0.09 for a distribution value of 10. Moreover, these very low and \\overlineely perceptible values might result from random misclassifications, given that training with DP achieves test accuracies of only 0.6 for a distribution value of 10. The *AAA* values for all other combinations and distributions in both datasets remain below 0.04 and are therefore insignificant. Consequently, the impact of the attack under these conditions can be considered negligible.

The poor performance of the attacks stems from the failed reconstruction attacks in both datasets. Applying DP results in the generation of images that appear as noisy black-and-white pixel patterns. While these images are not interpretable by humans, they might be interpretable by the model. This possibility arises because the generator could learn to produce only those features relevant to the model's classification process, even if they are not human-readable. This seems to be the case for the best-performing combinations, which exhibit significant spikes in attack accuracy and occasionally sustain short periods of consistent poisoning success. Despite this limited success, the attack was not sufficient to establish a persistent misclassification.

To conclude, the success of the CRPA is significantly weakened, or even prevented, by applying DP in our setup. This occurs due to the failure of the reconstruction attack using the GAN. The reconstructed samples did not, or only partially, include the features necessary to deceive the central model, appearing instead as random noise. It is worth noting that this result is achieved without rendering the training process ineffective. Applying DP reduces accuracy from an average of 99% to 89% for a distribution value of 10 and from an average 88% to 45% for a distribution value of 0.0. While this represents a significant performance reduction, Hitaj et al. reported that sample reconstruction using DP was only prevented when the model entirely failed to learn any patterns [10]. Our work demonstrates that DP can stop the reconstruction attack while still allowing the training process to continue. However, although this may prevent the reconstruction of human-identifiable images and thereby mitigate privacy leakage, it does not always prevent the reconstructed images from successfully poisoning the global model.

7. Conclusions and Future Work

Previous research has demonstrated that an adversarial participant can CRPA to generate samples of an arbitrary class G and poison the global model into misclassifying them as an arbitrary class *T*. In more realistic multi-class classification scenarios, where numerous combinations of *G* and *T* exist, data distribution among participants varies, and privacy-enhancing techniques like DP are commonly applied, the characteristics of CRPA have not yet been thoroughly studied. Our work addresses this gap by providing insights to better manage the existing security threats posed by CRPA.

To this end, we found that *(I) the choice of target classes significantly affects the success of CRPA.* We conducted experiments on the MNIST and FMNIST datasets, isolating each class to a single client, to evaluate the success of CRPA across all possible combinations. The results demonstrated that class selection affects the quality of attack outcomes. No consistent pattern indicating a general vulnerability among class types was found, suggesting that each combination exhibits unique vulnerability characteristics.

The poisoning attack inherently skews the central model toward incorrect class predictions. We demonstrated that *(II) a more balanced class distribution reduces the success of CRPA.* This occurs because a more balanced distribution makes it more difficult to deceive the global model with skewed representations of poisoned model updates. In contrast, the quality of reconstructed samples improves with a more balanced data distribution.

DP is a privacy-enhancing technique frequently applied in FL to safeguard training data privacy. We found that *(III) DP prevents CRPA by disrupting sample reconstruction*. Contrary to previous findings suggesting that DP is ineffective against CRPA, our experiments successfully disrupted sample reconstruction and, consequently, the poisoning attack.

Overall, both the aforementioned results and the methodology presented in the paper contribute to a better understanding of how reconstruction and poisoning attacks operate in more realistic scenarios. These findings also open several promising research directions. For instance, future research could investigate the applicability of novel methods for reconstruction attacks. Additionally, future research could focus on more advanced poisoning attacks and alternative defenses designed to maintain model accuracy. From our perspective, we identified several specific research directions. For instance, to ensure comparability, the number of mislabeled samples assigned to the adversarial client remained constant throughout the experiments. Consequently, we did not examine how the ratio of mislabeled samples affects local and global data distributions. However, this ratio likely influences the success of the poisoning attack. Investigating the optimal ratio, particularly in scenarios where clients hold varying amounts of data, could provide valuable insights into the attack's applicability. An extreme scenario could involve testing the impact of a free-rider attack (i.e., a client that does not contribute training data but solely focuses on CRPA) on federated training. Moreover, the total number of class samples is evenly distributed among all participants, and each participant holds the same number of data samples. Examining how an unequal distribution of class and data samples per client affects CRPA could provide further insights.

While we have demonstrated that DP can prevent the reconstruction of human-identifiable images, this does not necessarily counter CRPA. Despite appearing as random noise, the reconstructed samples might still be suitable for poisoning, as they sufficiently represent the model's internal perception of a class. Although some defenses can be implemented to mitigate such attacks [16], a sufficiently unique target class could still make it challenging for the server to identify the attacker. The primary defense lies in preventing the reconstruction attack altogether. However, preventing the reconstruction attack is only feasible through the application of DP. This is because the attacker cannot be prevented from using the global model received during the FL process to train a GAN and extract information from it. Future research could optimize CRPA by shifting the focus from reproducing human-identifiable samples to specifically targeting the poisoning of the central model. Additionally, research could focus on optimizing settings that balance model quality and defense against reconstruction attacks by considering a range of DP parameter values. Furthermore, this approach could be combined with optimization algorithms to achieve the best possible trade-off between accuracy and security.

Recommendations

Our work aims to equip practitioners with deeper insights into potential threats in FL. New threats may emerge within FL, potentially compromising data privacy – the very issue FL was designed to address. Therefore, practitioners implementing FL should be aware of its inherent limitations and potential risks. We

recommend assessing the criticality of data privacy and applying appropriate security mechanisms during training if necessary. However, implementing such mechanisms may reduce model performance. Thus, a balance between security and performance must be carefully maintained throughout the process.

Funding Support

This work was supported by the German Federal Ministry of Education and Research under Grant 16KIS1142K (project KIWI) as well as the European Regional Development Fund Interreg Upper Rhine initiative (project aura.ai).

Ethical Statement

This research does not involve any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data supporting the findings of this study are openly available in the MNIST Database at http://yann.lecun.com/exdb/mnist/ and on GitHub at https://github.com/zalandoresearch/fashion-mnist.

Author Contribution Statement

Christian Becker: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization. **José Antonio Peregrina:** Conceptualization, Methodology, Resources, Writing – review & editing, Visualization, Supervision. **Frauke Beccard:** Conceptualization, Methodology, Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Marisa Mohr:** Conceptualization, Methodology, Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Christian Zirpins:** Conceptualization, Methodology, Resources, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition.

References

[1] Gong, H., Cheng, S., Chen, Z., Li, Q., Quilodrán-Casas, C., Xiao, D., & Arcucci, R. (2022). An efficient digital twin based on machine learning SVD autoencoder and generalised latent assimilation for nuclear reactor physics. *Annals of Nuclear Energy*, *179*, 109431. https://doi.org/10.1016/j.anucene.2022.109431

[2] Gong, H.-L., Li, H., Xiao, D., & Cheng, S. (2024). Reactor field reconstruction from sparse and movable sensors using Voronoi tessellation-assisted convolutional neural networks. *Nuclear Science and Techniques*, *35*(5), 43. https://doi.org/10.1007/s41365-024-01400-w

[3] Bouacida, N., & Mohapatra, P. (2021). Vulnerabilities in federated learning. *IEEE Access: Practical Innovations, Open Solutions*, *9*, 63229–63249. https://doi.org/10.1109/ACCESS.2021.3075203

[4] Zhang, J., Li, M., Zeng, S., Xie, B., & Zhao, D. (2021). A survey on security and privacy threats to federated learning. In *2021 International Conference on Networking and Network Applications*, 319–326. https://doi.org/10.1109/NaNA53684.2021.00062

[5] Wang, Z., Song, M., Zhang, Z., Song, Y., Wang, Q., & Qi, H. (2019). Beyond inferring class representatives: User-level privacy leakage from federated learning. In *Proceedings: IEEE INFOCOM*, 2512–2520. https://doi.org/10.1109/INFOCOM.2019.8737416

[6] Zhang, J., Chen, B., Cheng, X., Binh, H. T. T., & Yu, S. (2021). PoisonGAN: Generative poisoning attacks against federated learning in edge computing systems. *IEEE Internet of Things Journal*, *8*(5), 3310–3322. https://doi.org/10.1109/JIOT.2020.3023126

[7] Zhang, J., Chen, J., Wu, D., Chen, B., & Yu, S. (2019). 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)*, 374–380. https://doi.org/10.1109/TrustCom/BigDataSE.2019.00057

[8] Wei, K., Li, J., Ding, M., Ma, C., Yang, H. H., Farokhi, F., . . . , & Vincent Poor, H. (2020). Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, *15*, 3454–3469. https://doi.org/10.1109/TIFS.2020.2988575

[9] Li, Q., Diao, Y., Chen, Q., & He, B. (2022). Federated learning on non-IID data silos: An experimental study. In *2022 IEEE 38th International Conference on Data Engineering*, 965–978. https://doi.org/10.1109/ICDE53745.2022.00077

[10] Hitaj, B., Ateniese, G., & Perez-Cruz, F. (2017). Deep models under the GAN: Information leakage from collaborative deep learning. In *Proceedings of the ACM Conference on Computer and Communications Security*, 603–618. https://doi.org/10.1145/3133956.3134012

[11] Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., . . . , & Zhao, S. (2021). Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, *14*(1–2), 1–210. https://doi.org/10.1561/2200000083

[12] Tolpegin, V., Truex, S., Gursoy, M. E., & Liu, L. (2020). Data poisoning attacks against federated learning systems. In L. Chen, N. Li, K. Liang & S. Schneider (Eds.), *Computer security: ESORICS 2020* (pp. 480–501). Springer International Publishing. https://doi.org/10.1007/978-3-030-58951-6_24

[13] Ziegler, J., Pfitzner, B., Schulz, H., Saalbach, A., & Arnrich, B. (2022). Defending against reconstruction attacks through differentially private federated learning for classification of heterogeneous chest X-ray data. *Sensors*, *22*(14), 5195. https://doi.org/10.3390/s22145195

[14] Zhang, X., & Luo, X. (2020). *Exploiting defenses against GAN-based feature inference attacks in federated learning*, 1–16.

[15] Bhagoji, A. N., Chakraborty, S., Mittal, P., & Calo, S. (2019). Analyzing federated learning through an adversarial lens. In *Proceedings of the 36th International Conference on Machine Learning*, 634–643. https://proceedings.mlr.press/v97/bhagoji19a.html

[16] Gupta, A., Luo, T., Ngo, M. V., & Das, S. K. (2022). Long-short history of gradients is all you need: Detecting malicious and unreliable clients in federated learning. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 13556 LNCS*(Ml), 445–465. https://doi.org/10.1007/978-3-031-17143-7_22

[17] Zhu, L., & Han, S. (2020). Deep leakage from gradients. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12500 LNCS(NeurIPS), 17–31. https://doi.org/10.1007/978-3-030-63076-8_2

[18] Jagielski, M., & Oprea, A. (2021). Does differential privacy defeat data poisoning? *ICLR 2021*.

[19] Ma, Y., Zhu, X., & Hsu, J. (2019). Data poisoning against differentially-private learners: Attacks and defenses. In *International Joint Conference on Artificial Intelligence*, 4732–4738. https://doi.org/10.24963/ijcai.2019/657

[20] Saad, M. M., O'Reilly, R., & Rehmani, M. H. (2024). A survey on training challenges in generative adversarial networks for biomedical image analysis. *Artificial Intelligence Review*, *57*(2), 19. https://doi.org/10.1007/s10462-023-10624-y

[21] Zhao, B., & Lao, Y. (2022). Towards class-oriented poisoning attacks against neural networks. In *Proceedings: 2022 IEEE/CVF Winter Conference on Applications of Computer Vision*, 2244–2253. https://doi.org/10.1109/WACV51458.2022.00230

[22] Bau, D., Zhu, J. Y., Wulff, J., Peebles, W., Zhou, B., Strobelt, H., & Torralba, A. (2019). Seeing what a GAN cannot generate. In *Proceedings of the IEEE International Conference on Computer Vision*, 4501–4510. https://doi.org/10.1109/ICCV.2019.00460

[23] Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324. https://doi.org/10.1109/5.726791

[24] Kayed, M., Anter, A., & Mohamed, H. (2020). Classification of garments from fashion MNIST dataset using CNN LeNet-5 architecture. In *2020 International Conference on Innovative Trends in Communication and Computer Engineering*, 238–243. https://doi.org/10.1109/ITCE48509.2020.9047776

[25] Ketkar, N. (2017). Introduction to Keras. In N. Ketkar (Ed.), *Deep learning with Python: A hands-on introduction* (pp. 97–111). Apress. https://doi.org/10.1007/978-1-4842-2766-4_7

[26] TensorFlow. (2024). *Implement differential privacy with TensorFlow privacy. (2022)*. Retrieved from: https://www.tensorflow.org/responsible_ai/privacy/tutorials/classification_privacy

[27] Hsu, T.-M. H., Qi, H., & Brown, M. (2019). Measuring the effects of non-identical data distribution for federated visual classification. *arXiv Preprint:1909.06335*.

How to Cite: Becker, C., Peregrina, J. A., Beccard, F., Mohr, M., & Zirpins, C. (2025). A Study on the Efficiency of Combined Reconstruction and Poisoning Attacks in Federated Learning. *Journal of Data Science and Intelligent Systems*. https://doi.org/10.47852/bonviewJDSIS52023970