electronics

MDPI

Article

GAN-Driven Data Poisoning Attacks and Their Mitigation in Federated Learning Systems

Konstantinos Psychogyios [1,*], Terpsichori-Helen Velivassaki [1], Stavroula Bourou [1], Artemis Voulkidis [1], Dimitrios Skias [2] and Theodore Zahariadis [1,3]

[1] Synelixis Solutions S.A., GR34100 Chalkida, Greece; terpsi@synelixis.com (T.-H.V.); zahariad@uoa.gr (T.Z.)
[2] Netcompany-Intrasoft S.A., GR19002 Paiania, Greece; dimitrios.skias@netcompany-intrasoft.com
[3] General Department, National and Kapodistrian University of Athens, GR15772 Athens, Greece
[*] Correspondence: psychogios@synelixis.com

Abstract: Federated learning (FL) is an emerging machine learning technique where machine learning models are trained in a decentralized manner. The main advantage of this approach is the data privacy it provides because the data are not processed in a centralized device. Moreover, the local client models are aggregated on a server, resulting in a global model that has accumulated knowledge from all the different clients. This approach, however, is vulnerable to attacks because clients can be malicious or malicious actors may interfere within the network. In the first case, these types of attacks may refer to data or model poisoning attacks where the data or model parameters, respectively, may be altered. In this paper, we investigate the data poisoning attacks and, more specifically, the label-flipping case within a federated learning system. For an image classification task, we introduce two variants of data poisoning attacks, namely model degradation and targeted label attacks. These attacks are based on synthetic images generated by a generative adversarial network (GAN). This network is trained jointly by the malicious clients using a concatenated malicious dataset. Due to dataset sample limitations, the architecture and learning procedure of the GAN are adjusted accordingly. Through the experiments, we demonstrate that these types of attacks are effective in achieving their task and managing to fool common federated defenses (stealth). We also propose a mechanism to mitigate these attacks based on clean label training on the server side. In more detail, we see that the model degradation attack causes an accuracy degradation of up to 25%, while common defenses can only alleviate this for a percentage of ∼5%. Similarly, the targeted label attack results in a misclassification of 56% compared to 2.5% when no attack takes place. Moreover, our proposed defense mechanism is able to mitigate these attacks.

Keywords: machine learning; federated learning; generative adversarial networks; data poisoning; label flipping

check for updates

Citation: Psychogyios, K.; Velivassaki, T.-H.; Bourou, S.; Voulkidis, A.; Skias, D.; Zahariadis, T. GAN-Driven Data Poisoning Attacks and Their Mitigation in Federated Learning Systems. *Electronics* **2023**, *12*, 1805. https://doi.org/10.3390/electronics12081805

Academic Editors: Charalabos Skianis, Philippe Krief, Enric Pages Montanera and John Soldatos

Received: 14 March 2023
Revised: 6 April 2023
Accepted: 9 April 2023
Published: 11 April 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Nowadays, machine learning is used to enhance many industrial and professional processes [1–8]. Such applications leverage image classification/processing [9–11] or regression models deployed in real-world scenarios with a real and immediate impact. However, these models, and especially deep learning models, require vast amounts of data to properly train. In most cases, the required data are collected from different locations and are thus independent from each other. An indicative example comes from electronic health record (EHR) patient-level data gathered from different hospitals referring to patients of different nationalities, socioeconomic status, etc. A common approach is to process the aggregated data in a centralized manner on a single server or device. Nevertheless, privacy concerns may arise with this approach, which may not comply with the existing data protection laws, such as the General Data Protection Regulation (GDPR) [12], increasing the demand for alternative decentralized solutions.

The FL approach [13] is a prominent solution to such concerns. This approach has been adopted by a number of industries, such as defense, IoT, and pharmaceutics [14–16]. Client data are kept confidential, because local models are trained individually on the customer's premises and only the model parameters are sent to a central server. Model aggregation applied on the central server derives the global model, which is shared among a number of clients in contrast to the centralized training case. The most common aggregation algorithm applies simple averaging, introduced as FedAvg [17]. In such a scenario, the global model encapsulates the knowledge gained from each client individually, without disclosing private data.

Even though this approach is appealing and addresses the issue of data privacy in the general case, there are still limitations concerning the system's security [18,19]. Specifically, because the clients fully control their data and the model they receive, they have the ability to alter each of these components, namely changing the model parameters (weights and biases) and altering data labels, for instance, by injecting fake data. Moreover, because the local and global models are exchanged through the network, adversaries could intercept communications and perform inference attacks [20], obtaining sensitive information. Consequently, a compromised client can intervene in the learning process of the federated system and perform malicious actions, such as pattern injection and model degradation [21,22]. These attacks can be data poisoning or model poisoning when private data or model parameters are altered, respectively [23,24]. In a smart agriculture scenario involving grapevine image classification, it is possible for a malicious actor to manipulate the labels assigned to the grapevine images, which could result in the misclassification of a particular class by the global model. The consequences of such tampering could be severe as it could lead to incorrect conclusions about the crops' health or status. These could in turn drive inappropriate management actions related to the irrigation or spraying processes, which could ultimately damage the crops. Additional effective security tools and mechanisms are thus needed to protect federated systems from data and model poisoning attacks. Recent research has been focused on developing variations of the aggregation algorithm that are tolerant to the appearance of malicious participants [25–27]. Such approaches include mean aggregation, trimmed mean aggregation [28,29], etc. These techniques rely on the assumption that model updates originating from malicious participants differ significantly from those from benign clients. Thus, by using distance-based algorithms, the poisoned parameters could be detected and excluded from the aggregation process.

GANs are neural networks that train in an adversarial way and were introduced primarily for the task of image generation. Although they have since been tailored to many different scenarios, image generation is still an area of active research where new architectures and techniques are employed for tasks, such as person generation, high-resolution image generation, etc. [30,31]. Within federated systems, GANs [32] have been used for either benign [33] or malicious reasons [34]. Regarding the first, the global model can be a GAN trained for the task of image generation in a federated manner. For the latter, the global model could be used as the discriminator by a malicious user to create a GAN model. This model could then be used to generate samples that belong to benign clients and formulate an inference attack. A major disadvantage of this assault is that the attack can only commence after the global model has been trained for a specific number of rounds. Furthermore, conventional label-flipping attacks involve modifying the original images.

In this paper, we introduce two data poisoning attacks aimed at an image classification task in a federated learning system. The image classification is performed on a grapevine dataset that has common types of grape diseases (e.g., Esca). These poisoning attacks are based on images generated by a GAN and are altered to fit the goal of each attack. The GAN was trained on malicious clients' data, which are very limited. Therefore, different alterations to the vanilla GAN architecture empirically prove to solve the issue of a limited database. In more detail, we create two label-flipping attacks: (a) one with the goal of global model accuracy degradation and (b) one with the goal of target label misclassification. In contrast to earlier studies, this assault can be launched right from the initial federated round and does not necessitate any modifications to the current dataset. We experiment with different numbers of clients and show that these attacks are able to achieve their objective in a vanilla federated environment together with stealth. Moreover, we also test these attacks against some common federated defenses and show that these are unable to mitigate them. To remedy this, we introduce a simple yet effective technique that operates on the server side. Our contributions can be summarized as follows:

- We train a GAN on limited image data and introduce appropriate modifications.
- We generate attacks based on label flipping that target a global classification model.
- We show that label-flipping attacks are successful with a limited number of malicious participants and can be initiated from the first federated round.
- We show that commonly used approaches fail to identify the poisoned updates.
- We propose a moderation technique to mitigate these types of backdoor attacks.

The rest of this paper is organized as follows. In Section 2, the related work on label-flipping attacks and federated learning defenses is presented. In Section 3, the federated training scenario as well as the adversary's goals and capabilities are defined. Section 4 describes the end-to-end approach of the federated scenario. The experiments evaluating the performance of each component of the process are presented in Section 5. Finally, Section 6 draws conclusions and outlines future work directions.

2. Related Work

Federated systems, are prone to a number of attacks, each falling into a category such as backdoor, label-flipping attacks, etc. [35–38]. To remedy this, different defenses have been introduced against these attacks, mainly regarding the aggregation process on the centralized device [39–43]. These two areas of research complement each other and jointly push the boundaries of federated learning further. On the one hand, new and successful attacks motivate the researchers to create updated and robust defenses. On the other hand, researchers are constantly trying to find vulnerabilities in state-of-the-art defense methods with the intent of making federated learning secure. In recent studies, GANs have been utilized for inference attacks through image generation, which is an area of intense research.

2.1. Label-Flipping Attacks

These types of attacks have been shown to be successful against federated systems, achieving both stealth and model misclassification with or without overall accuracy degradation.

Xiao et al. [44] tested two label-flipping attacks against support vector machines (SVM), namely the uniform random flip and distance from hyperplane-based flip. The tests conducted on ten real-world datasets show that the error rate increases as the number of flips increases. It is also noted that the first attack mentioned above is less effective compared to the second one.

Tolpegin et al. [23] tested label-flipping attacks within a federated system using two datasets, namely CIFAR-10 [45] and F-MNIST [46]. With these datasets, they tested various label-flipping scenarios, each tailored to a specific dataset. For example, in the F-MNIST case, the "shirt" label is changed to "t-shirt". Through the experiments, they demonstrated that label-flipping attacks are effective against federated learning systems, causing global model accuracy degradation, and that targeted poisoning impacts can be achieved.

Zhang et al. [47] trained a GAN model with the global model as the discriminator. They subsequently used this GAN to produce a poisoned dataset with custom fake labels for the malicious client. Before sending the updates, these are scaled to maximize the malicious impact. To test these attacks, the datasets Cifar-10 and F-MNIST were employed. The experimental evaluation proved this attack is effective in compromising the global model.

Xiao et al. [48] introduced a Sybil-based collusion attack scheme. In their work, the malicious clients colluded to launch successful label-flipping attacks that affect the global model. To evaluate their attack, they monitored the global model accuracy, attack success rate, and source class accuracy, where the source is the targeted label. Moreover, the datasets used for this purpose were F-MNIST and CIFAR-10. The results showed that the attack is successful and competitive against the state-of-the-art alternatives.

2.2. Federated Defenses

To mitigate the effect of model and data poisoning attacks, many defenses have been proposed that enhance the aggregation rule.

Median aggregation [28] is an aggregation method that selects the median value for each model parameter instead of the average. Each parameter is sorted, and subsequently the median value is selected. In the original paper, the authors showed that this method achieves the optimal rate for strongly convex quadratic losses. It is noted that this approach is strong, and because malicious model updates differ from the benign ones, a poisoned parameter is not frequently selected. However, because only one parameter is selected each time, there is substantial information loss.

Trimmed mean aggregation [29] is an aggregation method that also treats each parameter separately. Here, each parameter was sorted, and subsequently the top and bottom n instances were removed. At last, the remaining values were averaged. It is obvious that this method is sensitive to the parameter n because a smaller n may fail to include all the malicious nodes and a larger n may remove many benign nodes, negatively affecting the performance of the model.

Krum aggregation [29] is yet another aggregation method that filters model parameters based on distance. More specifically, for each model, the sum of squared distances to its closest n-f neighbors was calculated. Consequently, the choice of model was made based on the minimization of this distance. The authors also showed that this method achieves $O(n^2 \times d)$ complexity, which is linear to the dimension of the gradient.

Bulyan [49] showed an improved aggregation rule based on Krum and the trimmed mean. Firstly, Krum was applied iteratively to select N models. Subsequently, the trimmed mean was applied to each parameter as described above. We can see that this method is more sophisticated compared to Krum but requires substantially more computational power because it applies Krum multiple times.

2.3. Image Generation with GANs

GANs were originally introduced by Ian Goodfellow et al. [32] for the task of image generation. Since then, many improvements have been made to the original architecture and training procedure, as the original networks proved to be unstable.

Radford et al. [50] proposed a new architecture of GANs with specific constraints that make the training procedure more stable. The architecture utilized convolutional layers, which proved to be effective with image data. The validation results showed that this method is effective and can produce accurate results for many benchmark image datasets.

Arjovsky et al. [51] proposed a new loss function for GANs with constraints, namely the "Wasserstein distance" and weight clipping. The main goal was to overcome common learning stability problems, such as mode collapse [52]. The experimental results were sound and proved that this approach is more stable compared to previous methods [32].

Kodali et al. [53] proposed a new gradient penalty scheme called DRAGAN. It was proposed as an alternative to the weight clipping method introduced in the Wasserstein-GAN, which is not an optimal way to enforce constraints on the network. The experiments validated that this approach is promising, achieving a higher inception score compared to previous works.

Jin et al. [54] proposed a residual GAN for the task of plant leaf image generation, where the generator has a U-Net architecture. The main goal was to augment the dataset through image synthesis and achieve better post-synthesis classification results. The validation results indicated that this method indeed enhances the dataset, yielding more accurate classification results compared to other methods.

2.4. Related Work Review Findings

From the aforementioned research works, we can see that in many cases label-flipping attacks happen by changing the labels of the malicious clients' datasets. This is not always possible, because in many cases an adversary may infiltrate a client, but altering the dataset is impossible. The reasons for this lie in dataset protection, which may reveal the identity or activity of the adversary, or in the partial control of the adversary over the client node. Moreover, in other works employing a GAN for data poisoning in a federated system [47], the global model is chosen as the discriminator. In such a scenario, the global model must be trained first (possibly for many federated rounds) within the federated process before it can successfully fit the role of a local discriminator. Thus, the attacker may have to wait for an arbitrary amount of time before the attack is ready to be launched. It is also noted that the selection of the aggregation algorithm may significantly affect the robustness of the FL system against cyber-attacks. Indicatively, in [47], the vanilla federated average algorithm was chosen, which proved to be vulnerable, so an attack incident is more likely to be successful.

Additionally, previous works using a GAN for image synthesis utilize the whole PlantVillage dataset to generate new leaf images which is an easier scenario because most GANs fail when the dataset is small. For these reasons, we conclude that our approach contributes to advancing the relevant state-of-the-art solutions because we (a) modify the GAN training procedure and architecture to fit the scenario of limited available data; (b) launch an attack from the first federated round; (c) surpass many common defenses by enforcing stealth; (d) do not interfere with the poisoned client's original dataset, and instead create a new dataset with an enhancing rather than a substituting role; and (e) propose a server-side mitigation for these attacks.

3. Threat Model

Here, we introduce a detailed threat model for label-flipping attacks in federated systems as well as the federated task where the attacker operates.

3.1. Training Scenario

In normal operation, the task at hand is grapevine image classification where the classes are either diseased or healthy. Each client has a proportion of the complete dataset (a similar number of samples for each client), and a global convolutional neural network (CNN) is trained in a federated manner. Clients have images of all classes, and each local data distribution resembles the distribution of the complete dataset. Moreover, we consider that the server has a small set of clean-label images of all four classes.

3.2. Attacker's Training Interference

We assume the malicious attacker has control of N compromised clients. In this way, the malicious participants' training can be performed jointly or individually. Furthermore, the malicious clients participate in the learning process from the beginning. We also assume that the malicious attacker may enhance each compromised client's dataset by adding samples. The additional data used to generate attacks are derived from a GAN trained only on the malicious clients' datasets. The generated images combined with the existing dataset cannot exceed the number of the already existing images in each benign client by a large margin because the training time of each client (benign or malicious) must be roughly the same. This additional constraint allows malicious nodes to be less distinguishable from the benign ones, which would not be possible with large differences between the local training time of malicious and benign nodes.

3.3. Attacker's Objective

The attacker can adopt different types of malicious behavior, achieving different attack objectives. In our case, the attacker's activities may evolve toward two different goals. In the first case, the intruder targets a specific label (e.g., label 3), for which they aim to divert the global model's prediction. Specifically, the goal is to classify images of diseased leaves as healthy. To achieve these, a GAN is used to generate images of a specific disease class, and subsequently, the common class of these images is changed to healthy. At the same time, the attacker aims to keep global accuracy high to succeed in both the corresponding objective and stealth simultaneously. In the second case, the attacker aims to degrade the global model's accuracy again by poisoning the local dataset of the N compromised nodes. Here, the attacker generates images of all classes using a GAN and assigns random labels to them. Each malicious node dataset is enhanced with the generated images, and subsequently, the local model is trained with poisoned samples.

3.4. Adversary's Capability

The adversary is assumed to have no knowledge of the global aggregation algorithm on the server's side and can only influence the learning process through poisoned model updates. Moreover, the attacker is not aware of the benign clients' datasets. However, we assume they know that the classes included in the aggregated poisoned dataset are all the classes available within the federated system. In other words, there is not any class only present in benign local datasets and not in a malicious one. We also assume that the adversary cannot alter the private dataset of the compromised client but can only enrich it by adding more generated images. Lastly, the attacker is assumed to not be able to influence the local training algorithm and directly access or alter the local model's weights.

4. Approach

In this section, we describe our overall approach to attack generation based on GANs, which surpasses common federated defenses. We use a grapevine leaf disease image dataset for the task of image classification within the FL system. In this system, some clients are compromised and are controlled by a single adversary. The adversary trains a GAN using the joint dataset of the malicious clients and then uses it to launch label-flipping attacks. These attacks are tested against common federated defenses based on the aggregation algorithm. In Figure 1, we can see an overview of the whole process.

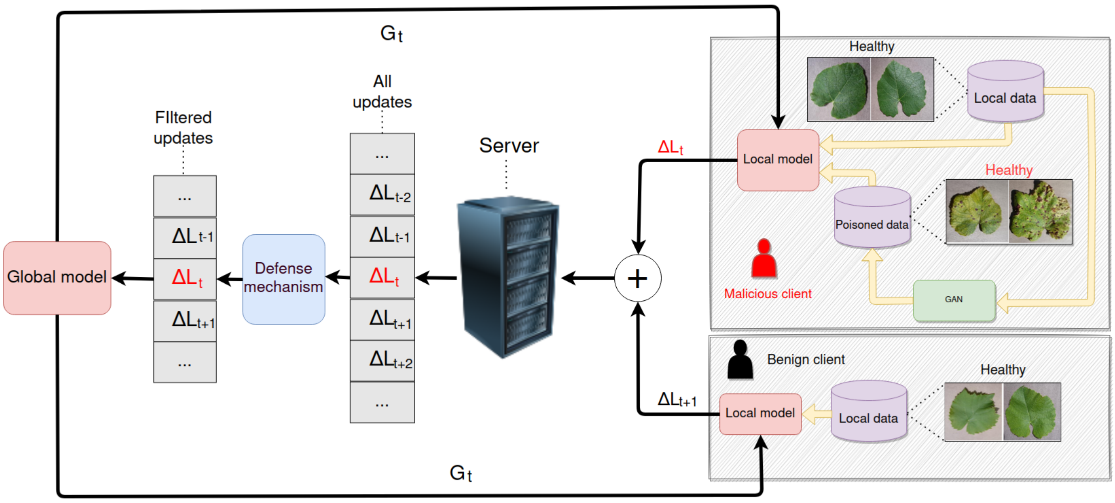

Figure 1. Proposed process for the GAN data poisoning attack. This figure illustrates the internal structure of both benign and malicious clients involved in the GAN data poisoning attack, presenting the proposed process for carrying out the attack. Additionally, it examines the federated learning process while utilizing secure aggregation techniques.

4.1. Dataset

To validate our results, we use the PlantVillage dataset [55]. This dataset consists of leaf images which can be either healthy or diseased. It contains 39 different classes of plant leaf, adding up to 61,486 images in total. For each crop (class), there are multiple subclasses representing healthy and diseased plants. In this work, we use the grapevine images, which are classified based on four classes, as depicted in Figure 2:

- Black measles (upper left).
- Black rot (upper right).
- Healthy (down left).
- Leaf blight(down right).

The original image size is 256 × 256 × 3 (RGB), adjusted to 128 × 128 × 3 for our experiments. Concerning the number and type of samples, there are 1000 healthy images, 1180 "black rot" images, 1076 leaf blight images and 1383 images of leaves infected by the Esca disease.

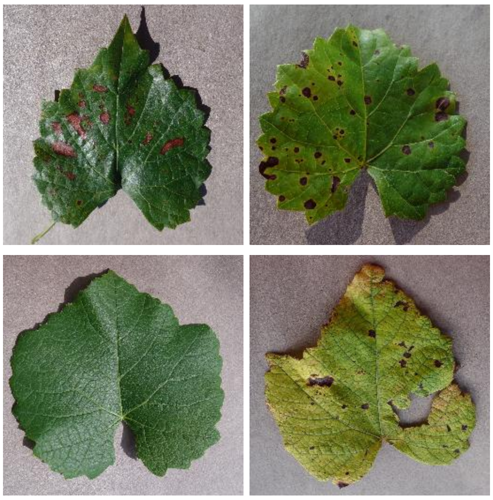

Figure 2. The four classes of the grapevine images of the PlantVillage dataset: black measles (**upper left**), black rot (**upper right**), healthy (**down left**), leaf blight (**down right**).

4.2. Image Generation

As mentioned above, the malicious clients jointly train a GAN to produce fake images in order to enrich their dataset and poison the global model for both attack objectives considered in Section 3.3. Thus, a GAN is trained for the image generation task using different datasets, corresponding to the different attacks that are desired to be formulated. A key point here is that the dataset available for training in each of these two attack cases is small, because it is assumed that at most 30% of the clients are malicious. For this reason, we need to adjust the GAN training procedure and architecture to fit this limitation.

The selected model's architecture is closely related to DCGAN [50], as depicted in Figure 3. For the generator's part, convolutional transpose layers with ReLU as activation and batch normalization [56] before the activation function have been chosen. We notice that batch normalization stabilizes training by normalizing batches to have zero mean and unit variance. Each convolutional transpose layer has progressively fewer filters and scales the image by a factor of 2, starting from a 4 × 4 × 256 image. As a result, the generator outputs 128 × 128 × 3 images that resemble the ones in the dataset. For the output layer, the selected activation function is the tanh function because the images are scaled to $[-1, 1]$. It is also noted that the input random noise is a vector of size 100.

For the discriminator, the architecture mirrors that of the generator. However, here traditional convolutional layers are used, and the dimensions shrink while going deeper into the network. It is also noted that before an image is fed to the discriminator, it is passed through a data augmentation layer. This is a technique that utilizes a small dataset to the maximum because the discriminator, instead of viewing the exact same images during each round, is alternatively fed a slight variation. More specifically, the amendments applied to each image are as follows:

- Random flip.

- Random rotation.
- Random zoom.
- Random translation.

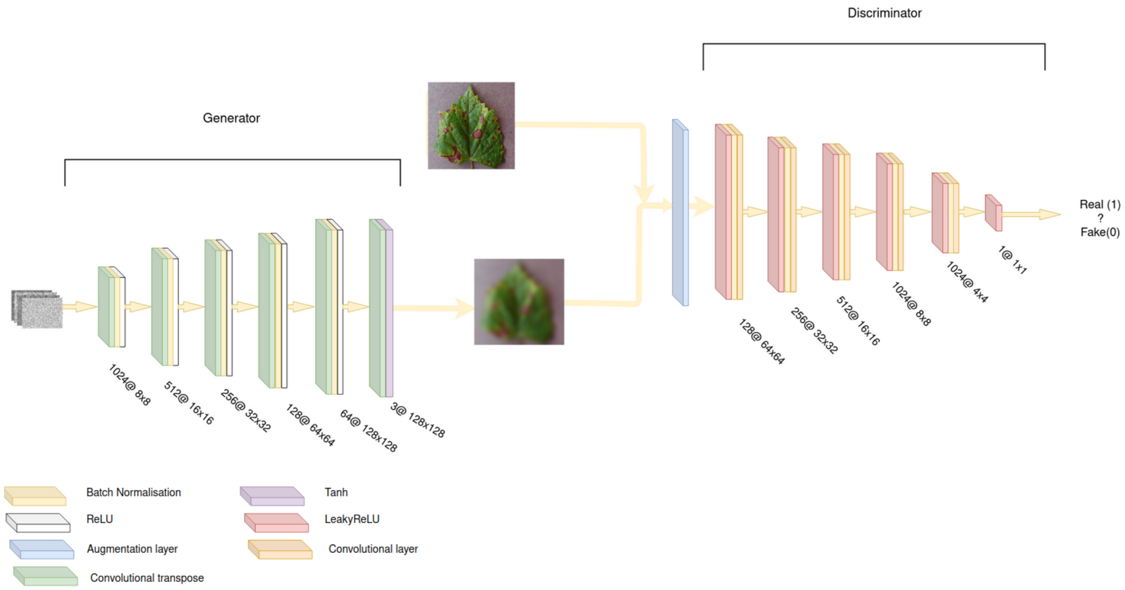

Figure 3. Proposed GAN architecture.

Furthermore, we again use batch normalization for the reasons described above. An important difference here is that LeakyReLU is used, instead of ReLU, for the activation function of convolutional layers. The reason is that this activation function helps with the problem of vanishing gradients, which is a common failure of GANs. Moreover, no sigmoid activation function is used in the output layer even though probabilities [0, 1] are extracted due to the use of logits (as defined by TensorFlow [57]) in the loss function computations. Both the generator and discriminator's losses are binary cross-entropy.

When it comes to training, GANs usually fall victim to the problem of mode collapse because the discriminator may fail to provide useful gradient feedback to the generator. To address this, significant modifications have been applied to the original GAN architecture, usually related to the loss function or the architecture. In the present work, the Wasserstein distance (earth mover distance) [51] or Wasserstein-1 is used:

$$W(P_r, P_\theta) = inf_{\gamma \in \prod(P_r, P_\theta)} E_{(x,y)\sim\gamma}[||x - y||] \tag{1}$$

where $\prod(P_r, P_\theta)$ is the set of all joint distributions γ, having marginals of P_θ and P_r. This loss function tends to produce more accurate results because it measures distance between 2 distributions more accurately when these distributions are far apart or/and have no overlap. Nevertheless, this loss function requires constraints to be carefully set to avoid vanishing gradients and enforce weight convergence. In the original paper [32], weight clipping was proposed, which sets an upper and lower bound. An enhanced method was proposed by Kodali et al. [53], namely DRAGAN. This approach suggests a gradient penalty defined as

$$\lambda \cdot E_{P_{real}, \delta \sim N_d(0, cI)}[||\nabla x D_\theta(x + \delta)|| - k]^2 \tag{2}$$

where $x + \delta$ is a noise point opposing to x which is a real point. Moreover, λ, k and c are parameters to be fine-tuned. This penalty is added to the loss function of the discriminator, scaled by the parameter λ. We empirically decided to use this method because it yields more accurate results, namely that the generated images were visually more realistic and the FID (Frechet Inception Distance), which measures generated image similarity compared to the real data, was lower.

4.3. Proposed Attacks

Federated systems are prone to attacks due to their nature. A common phenomenon is the appearance of malicious clients who aim to disturb the training process of the federated system either by manipulating the model's weights directly or indirectly through data. In the latter case, a malicious client may alter the existing data or inject new data to achieve their purpose. These types of attacks are called "data poisoning attacks". On such occasions, attackers infiltrate machine learning systems and introduce fake data points or manipulate existing data. Within the federated framework, one or more nodes may be malicious and aim to disturb the federated process with the intent of either pattern injection (e.g., targeted labeling attack) or model performance collapse.

To model this, we conducted different experiments in order to investigate the impact of GAN-based dataset generation, described in Section 4.1, on the synthesis of data poisoning attacks that can cause the deterioration of an FL model. We assume that a single attacker has control of all the malicious nodes and can thus process their datasets jointly. Firstly, we formulate a targeted label attack where a GAN is trained on a single class (in our case, leaf blight) and subsequently produces samples of this class. Then, the label of the generated samples is assigned to healthy and these are distributed to the malicious clients. Thus, each malicious client has an enriched dataset that consists of the primary (benign) dataset and the poisoned samples. The global model is subsequently trained on these clients, and the pattern $Leaf\ Blight = Healhty$ is injected. Secondly, we create a model degradation attack by training a GAN on all four classes of the malicious aggregated dataset. Subsequently, the attacker assigns random labels to these generated samples. Similar to the first case, each malicious client's dataset is enriched by a poisoned one, which leads to a global model accuracy degradation. It is also emphasized that the local models of the poisoned clients are trained both on the original clean dataset and the poisoned one generated by the GAN. This ensures both that the poisoned parameters will not be far from the benign ones regarding the parameter space and that the attacker achieves their target.

The global model under attack by the proposed approaches is a simple CNN.

4.4. Defenses

We also test these attacks against some common federated defenses. These defenses are variations of the aggregation rule and, in general, try to filter out malicious clients by viewing the model updates sent from the clients. The basic notion is that the model updates generated by malicious clients differ significantly compared to the benign ones. This difference can be investigated in the parameter space, and one could measure the distance between model parameters separately or between whole models (the sum of model parameters). To measure this difference, common distances are the Euclidean distance, cosine distance, etc. Thus, we test our attacks against the defenses of the following:

- Median aggregation.
- Trimmed mean aggregation (with known number of malicious participants).
- Krum aggregation.

With the exception of the Krum aggregation, the other two defenses operate at the parameter level and treat each parameter separately. Thus, they can result in a partially poisoned global model, where some parameters are aggregated using only benign updates and other parameters using both benign and malicious updates. On the other hand, Krum treats each model separately and thus, in a federated round, it is possible for a completely poisoned local model to be selected as the global one.

Lastly, we also propose a method that may further defend against federated attacks. More specifically, we assume that the server has a small dataset of clean-labeled images and is able to train the model for a few rounds after the client training. This technique will supposedly mitigate the attacks and correct the model's behavior.

5. Experiments

This section outlines the experiments conducted to validate the approach presented in Section 4. Firstly, we present the results of training a GAN on only the malicious clients, using all four classes to create a generator capable of reproducing any image from the dataset. Additionally, we train a GAN to generate images of the leaf blight class. These GANs are used to produce images, and then we assign malicious labels to each image to conduct either a model degradation (using the GAN trained on all four classes) or a targeted label attack (using the GAN trained only with leaf blight). Concerning the learning rate and the scaling factor of the loss, we used 0.001 and 10, respectively. We then test the effectiveness of these two data poisoning attacks against common federated learning aggregation algorithms to assess their robustness. Next, we evaluate the effectiveness of our proposed solution using the federated average aggregation algorithm. We train the global model for five more rounds on the server side with a small, clean dataset that is half the size of the clients' datasets. Finally, we use the overall global model accuracy as a metric for the model degradation task, and we study the confusion matrix of predictions for the targeted label attack, focusing on performance in each class separately.

5.1. GAN-Generated Images

We first train the GAN described in Section 4.2. for two separate cases. In the first case, the aim is to generate targeted label attacks, leading the global model to misclassify leaf blight images as healthy. To achieve this, a GAN is trained over leaf blight images only, with a subset of nodes that correspond only to the malicious clients, namely 30% of the whole FL system. We selected this number because a significant number of comparable studies that assess defense mechanisms or suggest attacks employ a percentage similar to this one for the malevolent clients [58–60]. The GAN has been trained for 80 rounds using the joint malicious dataset. The choice of this number is based on empirical evaluation both by viewing image quality and FID score. The effectiveness/efficiency of the image generation is depicted in Figure 4. We clearly see that as the training procedure progresses, the images more and more resemble the original distribution. At round 80, the model seems to have converged, which has led to the synthetic images being very similar to those in the original dataset. We also see that the difference in the quality of the produced images is larger between early epochs (e.g., epoch 5 and epoch 30) compared to the difference between later epochs (e.g., epoch 50 and epoch 80).

The FID score for the generated images up to epoch 80 is presented in Figure 5, revealing that the score is close to 500 at epoch 5 and roughly 160 at epoch 80. This trend indicates that the generated images become more closely aligned with the original dataset's distribution as the training progresses, which coordinates with our previous visual results demonstrating that lower FID scores correspond to better image quality. Based on these findings, we can infer that after 80 epochs, our model generates realistic images that resemble the original dataset.

In the second case, the GAN is trained for all four classes and results in a generator that can produce images of any class given random noise. For this scenario, the results have been similar to the aforementioned (single label) case. The subset of malicious clients remains the same, but their joint training dataset is larger because in this case we have four classes instead of one. We use the same training parameters (e.g., number of epochs, learning rate, penalty scaling factor, etc.) for this case, and the results closely follow those of the one-label (leaf blight) case. More specifically, the images generated are realistic after 80 epochs, and there is a similar behavior regarding the loss functions of the generator and the discriminator.

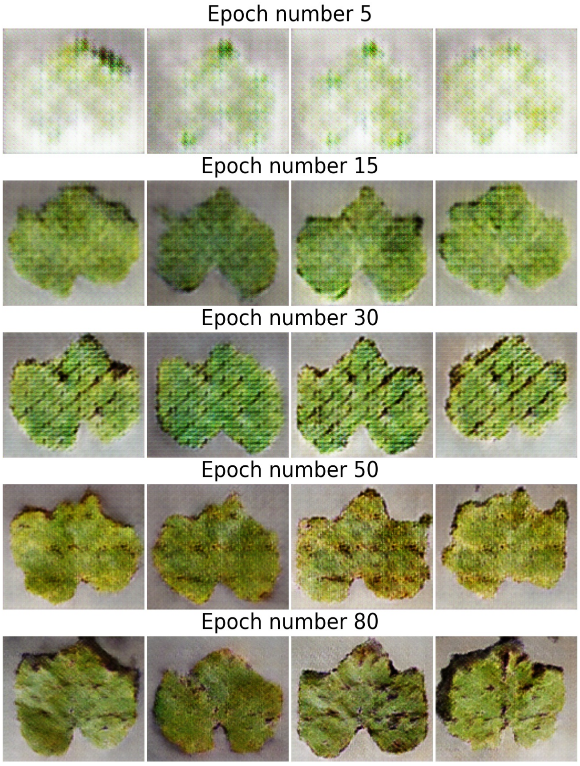

Figure 4. GAN-generated images for the leaf blight class after a certain amount of training epochs.

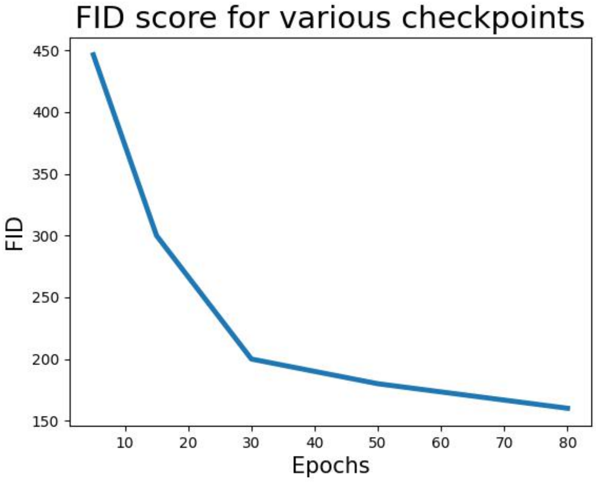

Figure 5. Frechet Inception Distance score for various epoch checkpoints.

5.2. Poisoning Attacks

In this subsection, the results of the experiments against the realization of the two types of attacks considered are presented. The experimental set-up includes X number of client nodes, participating in federated learning by performing local model training for the classification of grapevine leaves' images and sharing their model parameters with an aggregated server. Let us assume that a subset of the client nodes acts maliciously, which corresponds to 30% of the total number of clients. The malicious clients train their models for 50 federated rounds. After conducting the tests, we selected the value of 50 for this number because we observed that the model had converged at this point. Moreover, two local training rounds are considered for each client. The first round of our experiments refers to the baseline case, in which no malicious clients are present. Then, we proceed with the experiments for model degradation and targeted label attacks. The experimental results are reported using the accuracy plot and the confusion matrix. The former is appropriate for the model degradation attack, because this attack's goal is to reduce the accuracy of the global model. The confusion matrix is more suitable for evaluating the results of the targeted label attack experiments because the accuracy of the target class as well as the accuracy of all the other classes are the main parameters of interest in this case.

5.2.1. Model Degradation Attack

In this case, a GAN is deployed on the malicious nodes with the intention of causing global model performance degradation. The adversary uses the GAN model described in Section 4.2, trained with all four classes. Using this model, the attacker generates samples of all four classes and later distributes these samples to the compromised nodes. Subsequently, each malicious node randomly assigns a class label to the generated images. Each local model will be trained on images that resemble the original ones but have random labels, which is something that confuses the classifier. It is noted that the poisoned dataset was created before the training process and is available at the beginning of the training procedure. The experimental results evaluating the model degradation attack for the cases in which there is no defense or some of the three defense mechanisms considered in Section 4.4 can be seen in Figure 6.

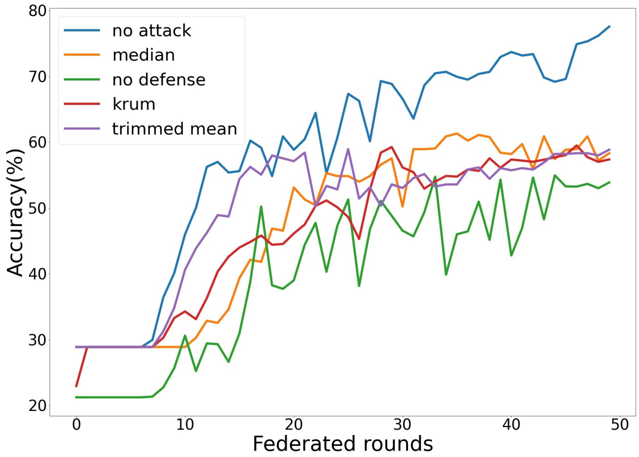

Figure 6. Accuracy for the model degradation attack with various defense mechanisms.

As shown in this figure, this attack is successful in degrading the model's global accuracy. Firstly, the blue line shows the case of no attack, where the FL system contains only benign nodes and the aggregation is a federated average to be used as a baseline. The green line shows the accuracy of the proposed attack for the case of federated average aggregation as the training procedure progresses. Regarding the other four classes, the only thing that changes is the aggregation algorithm. Orange is median aggregation, red is Krum aggregation and purple is trimmed mean. Initially, the model lacks robustness, and as a result, it categorizes all the images as the most common class, leading to a constant accuracy prior to the eighth federated round. Nevertheless, it becomes evident that the model can overcome this scenario of being stuck in a local minimum after several federated rounds. Consequently, we can see that there is a difference of around 25–30% in accuracy between the case in which no attack is realized and the case in which an attack has materialized with no defense in place. Moreover, regarding the defenses, the model accuracy is improved when some defense mechanisms are adopted compared to the case of no defenses. However, the improvement is approximately 5%, 7% and 5% for the median, Krum and trimmed mean approaches, respectively. The low improvement in all three cases implies that these mechanisms fail to identify the malicious nodes and that the poisoned updates are infiltrating the global model. It is worth noting that in the case of the trimmed mean defense, we have assumed that the parameter n (number of malicious clients) is known beforehand and thus the defense should be more powerful. Nevertheless, all three approaches consistently achieve an accuracy value below that of the global model, with the difference ranging from 5 to 20%. We thus assume that this kind of attack is stealthy, mostly due to the fact that the poisoning procedure closely resembles the training procedure and that the malicious nodes have both a poisoned and a clean dataset.

5.2.2. Targeted Label Attack

In this case, the test scenario includes 70% benign clients and 30% malicious. All benign clients have the same amount of data (images). The malicious clients generate images of a specific disease class, namely the leaf blight class. Then, these participants flip the label of the synthetic images to the one corresponding to healthy ones. The goal is to trick the model into classifying leaf blight images as healthy. This is indeed a targeted label attack, and the results could be catastrophic for a classification model. The results of the experiments conducted for evaluating the targeted label attack with the three or no defense mechanisms adopted are depicted in Figure 7.

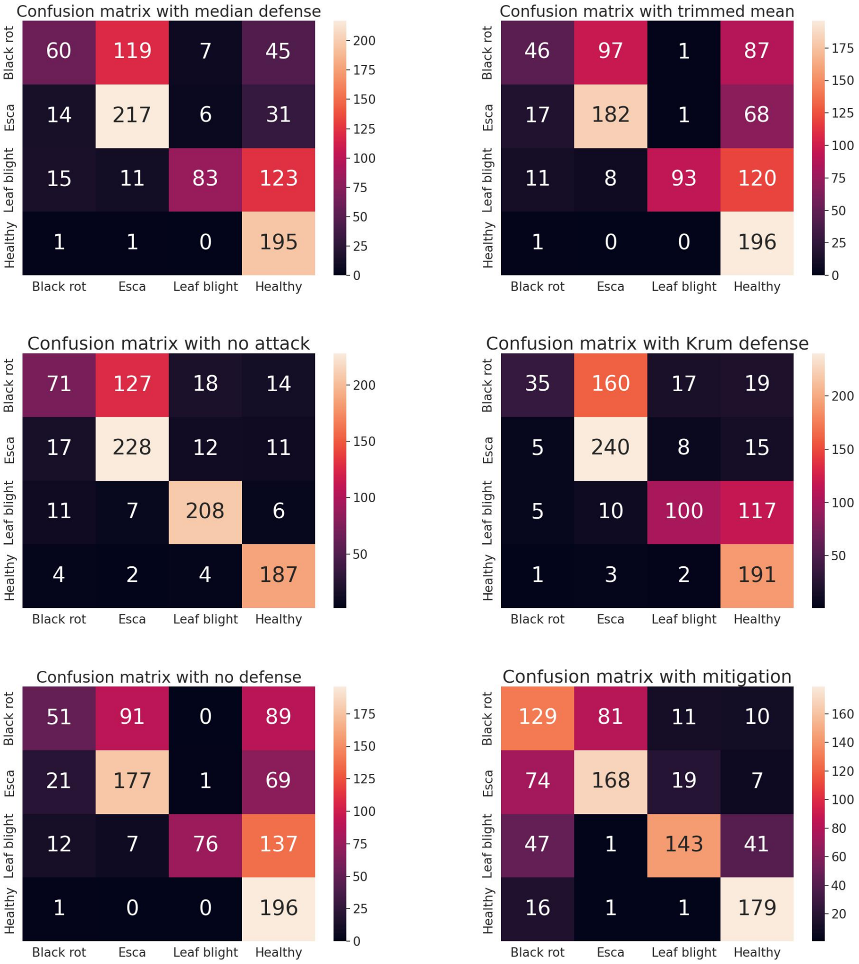

Figure 7. Confusion matrices for the targeted label attack with different aggregation mechanisms.

An initial observation is that the convolutional model consistently misclassifies numerous instances of black rot images as Esca. This can be attributed to the high degree of similarity between these images, rendering it difficult for the model to make an unequivocal distinction. Additionally, the predominance of Esca samples over black rot ones results in the misclassification of black rot as Esca rather than the reverse. It is worth noting that this behavior is not significant for the targeted label attack because only the classification results of the leaf blight and healthy classes are of interest. Moreover, comparing the case with no defenses to the case in which there is no attack, the attack appears to be successful for both the tasks of stealth and targeted label attacking. Specifically, this attack results in a model that greatly misclassifies leaf blight as healthy but achieves similar accuracy regarding the classification task for the other classes. In particular, 58% of the leaf blight images have been categorized as healthy compared to 2.5% for the case of no attack. Regarding the effectiveness of the defenses, they slightly improve the accuracy of this specific label. Specifically, with any of the defense mechanisms applied, the misclassification of leaf blight as healthy is approximately 51%. This happens due to the fact that these methods are performed at the parameter level and not for the whole model. This happens because a specific set of parameters (e.g., biases of the last layer) may be easily identified as poisoned by the defenses. However, viewing the results, we can assume that this set is small compared to the total number of parameters, and the bulk of the poisoned parameters are incorporated into the global model. Hence, it is obvious that these methods fail to select only benign client models, and the attack is deemed a success.

5.3. Mitigation

The results of the proposed mitigation approach are presented in Figure 7 (confusion matrix with mitigations) and Table 1 for the cases of targeted label and model degradation attacks. In the case of the targeted label attack, the method successfully mitigates the effects of the attack, as evidenced by the model no longer misclassifying the leaf blight images as healthy. Additionally, the results for the other classes remain unchanged, indicating that the attack reduces the accuracy of the different classes, and the attack is indeed targeted, as the accuracy of the other labels remains relatively stable.

Table 1. Test set accuracy for the cases with/without mitigation.

Mitigation	Accuracy
Yes	62%
No	50%

In the case of the model degradation attack, the proposed method quickly alleviates the effects of the attack, with the model achieving the accuracy of a non-attacked model within only five additional rounds. This result represents a substantial improvement of approximately 12%.

Overall, these findings demonstrate the effectiveness of the proposed method in mitigating backdoor attacks while remaining simple and easy to implement.

6. Conclusions and Future Work

In conclusion, the present paper has studied federated learning systems, which are vulnerable to data poisoning attacks, and proven that these can bypass common defense techniques if they are carefully crafted. This paper addresses topical issues in the literature related to this task. Specifically, our work considers and addresses sophisticated GAN-based attacks, able to be initiated because of the first round of training in FL systems, stepping beyond the existing works in which attacks can be realized only after several rounds of federated learning and can thus be mitigated based on each node's history. Additionally, in this paper, we suggest and validate the efficiency of potential mitigations to counteract label-flipping attacks. Our results show that these attacks can achieve their objective without being detected and can be catastrophic for applications, such as smart agriculture. In this regard, the attacker can take advantage of a small dataset if a number of clients are compromised and train a GAN to generate realistic images. Then, these images can be used to formulate data poisoning attacks without altering the original dataset of each compromised client. The attack can be launched during the initial round of federated learning and continue throughout the training process, without modifying the local dataset. This results in attacks that achieve a misclassification of up to 58% for the targeted label attack and an accuracy drop of up to 25% for the model degradation attack. To address this, we show a mitigation technique that is able to eradicate these types of attacks. This method successfully reverts the targeted label attack and increases the accuracy within the model degradation attack case by 12%.

In the future, we aim to test different GAN architectures and employ additional datasets. Additionally, settings with different hyper-parameters will be tested. This is very important because such experiments may improve the overall accuracy of the model significantly and tailor the model to the specific task. Moreover, a GAN trained on a leaf dataset could be used to produce images used for the federated classification of a different dataset. Such a case could be a GAN trained on images of a specific field and then deployed to poison another field (within a different federated system) to assess the cross-dataset accuracy. Lastly, we aim to evaluate the attacks against more kinds of defenses to test the stealth of the approach even further. Moreover, our future endeavors entail devising efficient defense methods (which may be based on machine learning) to counter these attacks, without the need for assuming the availability of a server-side dataset.

Author Contributions: Conceptualization, K.P.; methodology, K.P.; software, K.P.; validation, K.P.; formal analysis, K.P. and T.-H.V.; investigation, K.P.; writing—original draft preparation, K.P., T.-H.V., S.B., A.V., D.S. and T.Z.; writing—review and editing, K.P., T.-H.V., S.B., A.V., D.S. and T.Z.; visualization, K.P.; supervision, T.Z.; project administration, T.Z.; funding acquisition, T.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially funded by the H2020 IoT-NGIN project, contract no. 957246, within the H2020 Framework Program of the European Commission.

Data Availability Statement: The dataset used for this article is available online at https://data.mendeley.com/datasets/tywbtsjrjv (accessed on 8 April 2023).

Acknowledgments: We acknowledge the equal contribution of all the authors.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

EHR	Electronic Health Record
GDPR	General Data Protection Regulation
FL	Federated Learning
ML	Machine Learning
GAN	Generative Adversarial Network
FID	Frechet Inception Distance

References

1. Goumopoulos, C.; Potha, N. Mental fatigue detection using a wearable commodity device and machine learning. *J. Ambient. Intell. Humaniz. Comput.* **2022**, 1–19. [CrossRef]
2. Alanne, K.; Seppo, S. An overview of machine learning applications for smart buildings. *Sustain. Cities Soc.* **2022**, *76*, 103445. [CrossRef]
3. Nguyen, D.C.; Cheng, P.; Ding, M.; Lopez-Perez, D.; Pathirana, P.N.; Li, J.; Seneviratne, A.; Li, Y.; Poor, H.V. Enabling AI in future wireless networks: A data life cycle perspective. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 553–595. [CrossRef]
4. Zhang, X.; Wang, H.; Du, C.; Fan, X.; Cui, L.; Chen, H.; Deng, F.; Tong, Q.; He, M.; Yang, M.; et al. Custom-Molded Offloading Footwear Effectively Prevents Recurrence and Amputation, and Lowers Mortality Rates in High-Risk Diabetic Foot Patients: A Multicenter, Prospective Observational Study. *Diabetes Metab. Syndr. Obes.* **2022**, *15*, 103–109. [CrossRef] [PubMed]
5. Zhao, H.; Yang, X.; Chen, B.; Chen, H.; Deng, W. Bearing fault diagnosis using transfer learning and optimized deep belief network. *Meas. Sci. Technol.* **2022**, *33*, 065009. [CrossRef]
6. Ren, Z.; Zhen, X.; Jiang, Z.; Gao, Z.; Li, Y.; Shi, W. Underactuated control and analysis of single blade installation using a jackup installation vessel and active tugger line force control. *Mar. Struct.* **2023**, *88*, 103338. [CrossRef]
7. Kumar, M.; Sharma, R.K.; Sethi, I. (Eds.) *Machine Learning in Image Analysis and Pattern Recognition*; MDPI—Multidisciplinary Digital Publishing Institute: Basel, Switzerland, 2021.
8. Psychogyios, K.; Ilias, L.; Ntanos, C.; Askounis, D. Missing value imputation methods for electronic health records. *IEEE Access* **2023**, *11*, 21562–21574. [CrossRef]

9. Zhang, X.; Han, Y.; Lin, S.; Xu, C. A Fuzzy Plug-and-Play Neural Network-Based Convex Shape Image Segmentation Method. *Mathematics* **2023**, *11*, 1101. [CrossRef]
10. Avcı, H.; Karakaya, J. A Novel Medical Image Enhancement Algorithm for Breast Cancer Detection on Mammography Images Using Machine Learning. *Diagnostics* **2023**, *13*, 348. [CrossRef]
11. Massaro, A.; Dipierro, G.; Cannella, E.; Galiano, A.M. Comparative analysis among discrete fourier transform, K-means and artificial neural networks image processing techniques oriented on quality control of assembled tires. *Information* **2020**, *11*, 257. [CrossRef]
12. Paul, V.; Von dem Bussche, A. *The EU General Data Protection Regulation (gdpr). A Practical Guide*; Springer International Publishing: Cham, Switzerland, 2017; p. IX, 383.
13. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A.Y. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
14. Anastasakis, Z.; Psychogyios, K.; Velivassaki, T.; Bourou, S.; Voulkidis, A.; Skias, D.; Gonos, A.; Zahariadis, T. Enhancing Cyber Security in IoT Systems using FL-based IDS with Differential Privacy. In Proceedings of the 2022 Global Information Infrastructure and Networking Symposium (GIIS), Argostoli, Kefalonia Island, Greece, 26–28 September 2022; pp. 30–34.
15. Antunes, R.S.; da Costa, C.A.; Küderle, A.; Yari, I.A.; Eskofier, B. Federated Learning for Healthcare: Systematic Review and Architecture Proposal. *ACM Trans. Intell. Syst. Technol. (TIST)* **2022**, *13*, 1–23. [CrossRef]
16. Cazzato, G.; Massaro, A.; Colagr, E.A.; Lettini, T.; Cicco, S.; Parente, P.; Nacchiero, E.; Lospalluti, L.; Cascardi, E.; Giudice, G.; et al. Dermatopathology of Malignant Melanoma in the Era of Artificial Intelligence: A Single Institutional Experience. *Diagnostics* **2022**, *12*, 1972. [CrossRef] [PubMed]
17. Sun, T.; Li, D.; Wang, B. Decentralized federated averaging. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 4289–4301. [CrossRef]
18. Jatain, D.; Singh, V.; Dahiya, N. A contemplative perspective on federated machine learning: Taxonomy, threats and vulnerability assessment and challenges. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 6681–6698. [CrossRef]
19. Tahir, B.; Tariq, M. Vulnerability assessment and federated intrusion detection of Air Taxi enabled smart cities. *Sustain. Energy Technol. Assess.* **2022**, *53*, 102686. [CrossRef]
20. Luo, X.; Wu, Y.; Xiao, X.; Ooi, B.C. Feature inference attack on model predictions in vertical federated learning. In Proceedings of the 2021 IEEE 37th International Conference on Data Engineering (ICDE), Chania, Greece, 19–22 April 2021; pp. 181–192.
21. Huang, A. Dynamic backdoor attacks against federated learning. *arXiv* **2020**, arXiv:2011.07429.
22. Caldas, S.; Konečny, J.; McMahan, H.B.; Talwalkar, A. Expanding the reach of federated learning by reducing client resource requirements. *arXiv* **2018**, arXiv:1812.07210.
23. Tolpegin, V.; Truex, S.; Gursoy, M.E.; Liu, L. Data poisoning attacks against federated learning systems. In Proceedings of the European Symposium on Research in Computer Security, Guildford, UK, 14–18 September 2020; pp. 480–501.
24. Jere, M.S.; Farnan, T.; Koushanfar, F. A taxonomy of attacks on federated learning. *IEEE Secur. Priv.* **2020**, *19*, 20–28. [CrossRef]
25. Pillutla, K.; Kakade, S.M.; Harchaoui, Z. Robust aggregation for federated learning. *arXiv* **2019**, arXiv:1912.13445.
26. Fereidooni, H.; Marchal, S.; Miettinen, M.; Mirhoseini, A.; Möllering, H.; Nguyen, T.D.; Rieger, P.; Sadeghi, A.R.; Schneider, T.; Yalame, H.; et al. SAFELearn: Secure aggregation for private federated learning. In Proceedings of the 2021 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 27 May 2021; pp. 56–62.
27. Song, J.; Wang, W.; Gadekallu, T.R.; Cao, J.; Liu, Y. Eppda: An efficient privacy-preserving data aggregation federated learning scheme. *IEEE Trans. Netw. Sci. Eng.* 2022, *early access*. [CrossRef]
28. Yin, D.; Chen, Y.; Kannan, R.; Bartlett, P. Byzantine-robust distributed learning: Towards optimal statistical rates. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 5650–5659.
29. Blanchard, P.; El Mhamdi, E.M.; Guerraoui, R.; Stainer, J. Machine learning with adversaries: Byzantine tolerant gradient descent. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
30. Tang, H.; Bai, S.; Torr, P.H.; Sebe, N. Bipartite graph reasoning GANs for person image generation. *arXiv* **2020**, arXiv:2008.04381.
31. Wang, T.C.; Liu, M.Y.; Zhu, J.Y.; Tao, A.; Kautz, J.; Catanzaro, B. High-resolution image synthesis and semantic manipulation with conditional GANs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8798–8807.
32. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. Acm* **2020**, *63*, 139–144. [CrossRef]
33. Zhao, Z.; Birke, R.; Kunar, A.; Chen, L.Y. Fed-TGAN: Federated learning framework for synthesizing tabular data. *arXiv* **2021**, arXiv:2108.07927.
34. Wang, Z.; Song, M.; Zhang, Z.; Song, Y.; Wang, Q.; Qi, H. Beyond inferring class representatives: User-level privacy leakage from federated learning. In Proceedings of the IEEE INFOCOM 2019-IEEE Conference on Computer Communications, Paris, France, 29 April–2 May 2019; pp. 2512–2520.
35. Xie, C.; Huang, K.; Chen, P.Y.; Li, B. Dba: Distributed backdoor attacks against federated learning. In Proceedings of the International Conference on Learning Representations, online, 27–30 April 2020.
36. Fraboni, Y.; Vidal, R.; Lorenzi, M. Free-rider attacks on model aggregation in federated learning. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Online, 13–15 April 2021; pp. 1846–1854.

37. Shejwalkar, V.; Houmansadr, A. Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning. In Proceedings of the Network and Distributed Systems Security (NDSS) Symposium, Virtual, 21–25 February 2021.
38. Lyu, L.; Yu, H.; Zhao, J.; Yang, Q. *Federated Learning*; Springer International Publishing: Cham, Switzerland, 2020; pp. 3–16.
39. Huang, Y.; Gupta, S.; Song, Z.; Li, K.; Arora, S. Evaluating gradient inversion attacks and defenses in federated learning. In Proceedings of the Advances in Neural Information Processing Systems, Online, 6–14 December 2021; pp. 7232–7241.
40. Gong, X.; Chen, Y.; Wang, Q.; Kong, W. Backdoor Attacks and Defenses in Federated Learning: State-of-the-art, Taxonomy, and Future Directions. *IEEE Wirel. Commun.* **2022**. [CrossRef]
41. Zhang, X.; Luo, X. Exploiting defenses against GAN-based feature inference attacks in federated learning. *arXiv* **2020**, arXiv:2004.12571.
42. Fung, C.; Yoon, C.J.; Beschastnikh, I. Mitigating sybils in federated learning poisoning. *arXiv* **2018**, arXiv:1808.04866.
43. Sun, J.; Li, A.; Wang, B.; Yang, H.; Li, H.; Chen, Y. Provable defense against privacy leakage in federated learning from representation perspective. *arXiv* **2020**, arXiv:2012.06043.
44. Xiao, H.; Xiao, H.; Eckert, C. Adversarial label flips attack on support vector machines. In Proceedings of the 20th European Conference on Artificial Intelligence (ECAI). Including Prestigious Applications of Artificial Intelligence (PAIS-2012) System Demonstrations Track, Montpellier, France, 27–31 August 2012; pp. 870–875.
45. Krizhevsky, A.; Hinton, G. *Learning Multiple Layers of Features from Tiny Images*; University of Toronto: Toronto, ON, Canada, 2009; p. 58.
46. Xiao, H.; Rasul, K.; Vollgraf, R. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. *arXiv* **2017**, arXiv:1708.07747.
47. Zhang, J.; Chen, B.; Cheng, X.; Binh, H.T.T.; Yu, S. Poisongan: Generative poisoning attacks against federated learning in edge computing systems. *IEEE Internet Things J.* **2020**, *8*, 3310–3322. [CrossRef]
48. Xiao, X.; Tang, Z.; Li, C.; Xiao, B.; Li, K. SCA: Sybil-based Collusion Attacks of IIoT Data Poisoning in Federated Learning. *IEEE Trans. Ind. Inform.* **2023**, *19*, 2608–2618. [CrossRef]
49. Guerraoui, R.; Rouault, S. The hidden vulnerability of distributed learning in byzantium. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 3521–3530.
50. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434.
51. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 214–223.
52. Thanh-Tung, H.; Tran, T. Catastrophic forgetting and mode collapse in GANs. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–10.
53. Kodali, N.; Abernethy, J.; Hays, J.; Kira, Z. On convergence and stability of GANs. *arXiv* **2017**, arXiv:1705.07215.
54. Jin, H.; Li, Y.; Qi, J.; Feng, J.; Tian, D.; Mu, W. GrapeGAN: Unsupervised image enhancement for improved grape leaf disease recognition. *Comput. Electron. Agric.* **2022**, *198*, 107055. [CrossRef]
55. Geetharamani, G.; Pandian, A. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Comput. Electr. Eng.* **2019**, *76*, 323–338.
56. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456.
57. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. TensorFlow: A system for Large-Scale machine learning. In Proceedings of the 2th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
58. Li, S.; Ngai, E.; Ye, F.; Voigt, T. Auto-weighted robust federated learning with corrupted data sources. *ACM Trans. Intell. Syst. Technol. (TIST)* **2022**, *13*, 1–20. [CrossRef]
59. Sattler, F.; Müller, K.R.; Wieg, T.; Samek, W. On the byzantine robustness of clustered federated learning. In Proceedings of the CASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 8861–8865.
60. Ganjoo, R.; Ganjoo, M.; Patil, M. Mitigating Poisoning Attacks in Federated Learning. In *Innovative Data Communication Technologies and Application*, Proceedings of the 3rd International Conference on Innovative Data Communication Technologies and Application (ICIDCA 2021), Coimbatore, India, 20–21 August 2021; Springer Nature: Singapore, 2022; pp. 687–699.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.