MESAS: Poisoning Defense for Federated Learning Resilient against Adaptive Attackers

Torsten Krauß
University of Würzburg
Würzburg, Germany
torsten.krauss@uni-wuerzburg.de

Alexandra Dmitrienko
University of Würzburg
Würzburg, Germany
alexandra.dmitrienko@uni-wuerzburg.de

ABSTRACT

Federated Learning (FL) enhances decentralized machine learning by safeguarding data privacy, reducing communication costs, and improving model performance with diverse data sources. However, FL faces vulnerabilities such as untargeted poisoning attacks and targeted backdoor attacks, posing challenges to model integrity and security. Preventing backdoors proves especially challenging due to their stealthy nature. Existing mitigation techniques have shown efficacy but often overlook realistic adversaries and diverse data distributions.

This work introduces the concept of *strong adaptive adversaries*, capable of adapting to multiple objectives simultaneously. Extensive empirical testing reveals existing defenses' vulnerability in this adversary model. We present *Metric-Cascades (MESAS)*, a novel defense method tailored to more realistic scenarios and adversary models. MESAS employs multiple detection metrics simultaneously to combat poisoned model updates, posing a complex multi-objective problem for adaptive attackers. In a comprehensive evaluation across nine backdoors and three datasets, MESAS outperforms existing defenses in distinguishing backdoors from data distribution-related distortions *within* and *across* clients. MESAS offers robust defense against strong adaptive adversaries in real-world data settings, with a modest average overhead of just 24.37 seconds.

CCS CONCEPTS

• **Security and privacy** → **Distributed systems security**; • **Computing methodologies** → *Distributed artificial intelligence*.

KEYWORDS

federated learning; security; poisoning attacks; backdoor attacks

ACM Reference Format:
Torsten Krauß and Alexandra Dmitrienko. 2023. MESAS: Poisoning Defense for Federated Learning Resilient against Adaptive Attackers. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS '23), November 26–30, 2023, Copenhagen, Denmark*. ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3576915.3623212

CCS '23, November 26–30, 2023, Copenhagen, Denmark

ACM ISBN 979-8-4007-0050-7/23/11…$15.00
https://doi.org/10.1145/3576915.3623212

1 INTRODUCTION

Federated Learning (FL) enables the collaborative training of a Deep Neural Network (DNN) among multiple clients [55]. Each client trains a DNN locally on its own data, incorporating the knowledge from the data into the model parameters. Only the changes in the trained model parameters are then transmitted to a central server for aggregation. This approach allows clients to participate in the federation while adhering to privacy regulations [11, 25, 97], as the raw data are not shared with third parties. Compared to centralized learning approaches, FL is also more computationally effective as it shifts training efforts to the clients, leading to fewer resource requirements on the server. As a result, FL is already being applied in multiple application domains [107]. For instance, in image recognition [48], hospitals are training models collaboratively [19, 33, 64, 77, 82, 83, 86], and in Natural Language Processing (NLP) domain it is used for text prediction [34, 56, 75], sentiment analysis [6], and personalization [15]. Moreover, FL can be applied for human mobility prediction [27], visual object detection [50], and human activity recognition [88]. We refer for more examples to [43].

In federations, a subset of clients can be controlled by an adversary who submits poisoned updates to the server. These attacks can be untargeted [26, 44, 103, 106], with the goal to reduce the prediction performance of the model. Alternatively, targeted poisoning attacks, also called backdoor attacks [4, 5, 7, 10, 16, 17, 31, 32, 45, 62, 66, 70, 78, 90, 96, 100, 105], aim to maintain an unobtrusive performance on regular input but force the model to output a selective prediction when provided input containing a specific trigger. Hence, backdoors pose a greater risk, as such attacks are harder to detect, and the unexpected misbehaviour can harm model users in real-world applications, such as self-driving cars [46, 63, 110].

Defenses against poisoning attacks follow one of the three strategies: (i) *Influence Reduction (IR)* solutions try to reduce the impact of the individual models before or after aggregation to weaken potential poisoning behavior [3, 5, 61, 92], (ii) *Robust Aggregation (RA)* methods enhance robustness of aggregation algorithms against backdoors [55, 109], and (iii) *Detection and Filtering (DF)* approaches detect the poisoned models and filter them out before the aggregation step [9, 29, 60, 65, 76, 84, 113].

Generally, IR and RA approaches inevitably reduce the performance of the benign functionality, while DF methods can suffer high False-Positive-Rates (FPRs) and False-Negative-Rates (FNRs). This downside of the DF methods is mainly based on two root causes: First, defense-aware adversaries may adapt the poisoned model to be inconspicuous, thus circumventing the defense. Second, in real-world scenarios, the clients may possess very different data within the local datasets, which makes it difficult to distinguish if a

model with uncommon metrics is derived from a poisoned dataset or just a dataset with uncommon data distributions.

Identifying Problems. In this paper, we focus on DF methods, as they have the benefit of maintaining benign model performance. We analyze related work and observe that, even though most solutions were evaluated against adaptive attackers, the meaning of the "adaptive attacker" is defined differently across different papers, which makes it difficult to assess their true detection capabilities and compare them to each other. We also notice that none of the previous works considered an adaptive attacker with multi-objective adaption capabilities, i.e., attackers that could try to adapt to several metrics at once, while nothing prevents real-world adversaries from following this strategy. Hence, the resilience of all existing defenses against such strong adaptive attackers remains unclear. Furthermore, we also identify that all existing positioning defenses, from all three categories, were evaluated under certain assumptions made with regard to underlying data distributions. In particular, while many consider non-identically and independently distributed (non-IID) data distributions within clients, no single defense method was evaluated in a scenario with non-identically and independently distributed data *across* clients so far.

Contributions. To address the aforementioned problems, this paper makes the following contributions:

- We introduce the notion of a *strong adaptive adversary*, who is capable of adapting to FL defenses by balancing multiple adaptation objectives and applying manual invasions on the model parameters. Leveraging this sophisticated adaptation strategy, we attack and evaluate nine existing defenses, showing that all these methods can be circumvented, hence creating a gap between the state-of-the-art defense methods and realistic scenarios.
- We are the first to point out the fact that previous defenses were never evaluated in settings where datasets have different distributions within *and across* the clients. We term such a scenario as *inter-client non-IID* and demonstrate through intensive evaluation of nine solutions that they are not resilient in such a setting, which implies their limited real-world applicability.
- We propose *Metric-Cascades (MESAS)*, a new server-side defense of DF-type for FL, that resilient against our *strong adaptive adversary*. MESAS detects backdoors in local models based on a cascade of six well-chosen metrics and can identify and filter out both, targeted and untargeted poisoning attacks. Further, MESAS is the first defense, that effectively filters backdoors in arbitrary data distribution scenarios, including inter-client non-IID settings, by conducting statistical tests on multiple metrics and, as such, being able to distinguish backdoors from unusual data distributions.
- We conduct a systematic large-scale study to analyze the factors that influence MESAS and demonstrate its independence from application-specific factors like datasets, model architectures, IID scenarios, adaption strategies, and nine sophisticated poisoning methods. Furthermore, we compare the performance of MESAS in terms of detection capabilities and runtime overhead to nine existing defenses. MESAS outperforms all evaluated methods regarding robustness against adaptive strategies and in terms of backdoor removal performance under realistic inter-client non-IID scenarios. Moreover, it achieves this while incurring a runtime overhead of only 24.37 seconds on average.

Overall, our work depicts two major weaknesses of existing FL defenses that are problematic in real-world applications, namely adaptive adversaries and realistic inter-client non-IID data scenarios. The proposed DF defense, MESAS, effectively prunes different sophisticated poisonings simultaneously, withstands strong adaptive adversaries, and is robust in arbitrary data scenarios including inter-client non-IID.[1]

2 BACKGROUND

In this section, we first provide FL fundamentals in Sect. 2.1, followed by background information about poisoning attacks and classical adaptive adversarial models in Sect. 2.2.

2.1 Federated Learning

In a FL [39, 55, 108] framework, multiple clients $C_k \in \{C_1, \ldots C_{\mathcal{N}}\}$ collaborate under the orchestration of a central server to improve a Deep Neural Network (DNN). The collaborative process involves each client C_k training a local DNN model on a local dataset and subsequently transmitting the result to the server for aggregation. Thus, the data never leave the client side, improving the privacy of training data compared to centralized learning. Additionally, the computational effort is distributed, so that fewer resources need to be allocated on the server, reducing the costs for infrastructure.

FL is an iterative process, where the central server selects a subset n of the $\mathcal{N}$ available clients $C_i \in \{C_1, \ldots C_n\}$ for each training round r and distributes an (initially untrained) global model G^r. Each client initializes its local model $L_i^r = G^r$ and trains a new local model L_i^{r+1} with the local dataset $\mathcal{D}_i$, based on a predefined algorithm with hyper-parameters, such as learning rate (LR), etc. After training, the clients submit the model updates $\mathcal{U}_i^r = L_i^{r+1} - G^r$ to the server, which aggregates them into a new global model G^{r+1}. There are multiple aggregation methods [9, 24, 60, 109] available for this step, with Federated Averaging (FedAVG) [55] being the most commonly used. FedAVG calculates the weighted average of all the updates using the global learning rate δ as formalized in [40].[2] After aggregation, the new round $r+1$ is initialized by $\mathcal{S}$.

2.2 Poisoning Attacks in Federated Learning

In the following, we distinguish between *untargeted* and *targeted* poisoning attacks [94, 104] and discuss the two methods that are applied to launch those attacks, namely *data* and *model poisoning*.

Untargeted poisoning aims to reduce the model prediction performance of the global model G^{r+1} on a benign test dataset that contains samples with correctly labeled predictions, which we refer to as model accuracy (MA) (cf. [40]). To name an example, the adversary can assign an incorrect label for each sample in the dataset, thus misdirecting the model during training.

[1] We provide an extended paper version containing more results in [40].

[2] Originally, FedAVG assigns weights to updates according to the respective sizes of the local datasets. However, in situations where the presence of adversaries is a possibility, an equal weighting scheme is employed to thwart any attempts by adversarial clients to amplify their influence by reporting increased dataset sizes.

Targeted attacks, also called *backdoor attacks*, strive to force a DNN to produce attacker-chosen mispredictions when fed with inputs that contain attacker-chosen features, so called *triggers*, while maintaining a high MA on regular data. As an example for a trigger, a red pixel or any other unique pattern can be embedded inside an image [5, 32, 51]. In more detail, an adversary, who controls one or more clients within a federation, tries to submit poisoned local models to the server, so that the aggregated model G^{r+1} outputs a predefined target prediction when provided with an input sample containing the trigger, with target and trigger being chosen by the adversary. An effective attack has high prediction performance, called backdoor accuracy (BA), on triggered input tested with a dataset that contains only triggered samples (cf. [40]). We attest a successful attack for a BA bigger than 60% in the global model.

Data poisoning [91] describes the process of converting a benign into a poisoned dataset by assigning malicious labels and, for backdoors, adding triggers. A model trained on that dataset then includes the malicious behavior. Thereby, the poison data rate (PDR) defines the fraction between benign and poisoned samples and can control the balance between attack effectiveness and stealthiness.

Model poisoning allows arbitrary manipulation of the whole training process, e.g., changing hyper-parameters and loss functions. Additionally, the model can be modified manually before, during, or after training. Mostly, this method is applied to improve the BA or to adapt to defenses, but can also be used to implement untargeted attacks without data poisoning. To adapt to a defense while maintaining high MA and BA, an additional objective ($Loss^{Adaption}$) can be added to the loss function for the MA and BA ($Loss^{MA/BA}$), which is also called constraining [5, 23]. As shown in Eq. 1, the objectives are weighted by α, allowing the adversary to prioritize between performance (MA/BA) and adaption intensity and consequently stealthiness.

$$Loss = \alpha \cdot Loss^{MA/BA} + (1 - \alpha) \cdot Loss^{Adaption} \tag{1}$$

A *classical adaptive adversary* creates a loss function for the deployed defense and applies Eq. 1 to bypass the defensive measure[3]. Additionally, the updates of a poisoned local model can be scaled regarding the Euclidean distance to strengthen the influence on the aggregated model, hence increasing the BA. Training with a poisoned dataset combined with scaling is called *train-and-scale* and adaption combined with scaling is called *constrain-and-scale* [5].

The goal of a defense against poisoning attacks is to create a situation, where $Loss^{MA/BA}$ and $Loss^{Adaption}$ cannot be perfectly optimized simultaneously so that the adversary is faced with a trade-off between an effective attack and adapting to the defense, which is called *adversarial dilemma* [28, 76].

3 PROBLEMS AND DEFINITIONS

In this section, we define our threat model including the concept of a strong adaptive adversary in Sect. 3.1. The concluding Sect. 3.2 is devoted to the problem of arbitrary data distributions.

[3]The adversary can adapt to any objective and most likely aligns to the metrics of defenses, but is not restricted to those.

3.1 Threat Model

We analyze a classical FL system as depicted in Sect. 2.1. The aggregation server applies FedAVG with a fixed global LR of $\delta = 1$. We consider an adversary, who captures multiple clients C_i which are then denoted as $A_j \in \{C_1, \ldots C_n\}$ and can conduct any data and model poisoning attacks (cf. Sect. 2.2). The adversary is aware of the code running on the aggregation server, including the details of defense mechanisms, which provides the necessary knowledge for adaption attempts. Analogous to related works [3, 9, 60, 65, 76, 84], we consider $n/2 + 1$ benign clients (*majority assumption*) in each training round r. Since it is uncertain if adversaries participate in a round r, the server weights all model updates equally with $1/n$. In contrast to previous works, we do not make any assumption about the data distributions [114] within or across clients' dataset.

Problem of an adaptive adversary. DF defenses against poisoning attacks in FL are based on custom metrics. An adversary can try to circumvent the defense by adapting the value of the respective metric used for detection derived from the locally crafted poisoned model to a benign value during training[4]. As a state-of-the-art technique for this challenge, Eq. 1 is used to consider multiple objectives and simultaneously allowing the adversary to weight between better prediction performance (MA and BA) and higher adaption level via α. This adaption method from Eq. 1 works well in two cases: 1) For only one adaption loss, since α can then balance the importance of main task and adaption properly and 2) for multiple adaption losses, where the different adaption losses summed to one value. The latter scenario works only well if all losses are at the same scale, as different components of adaption losses cannot be individually tuned. For example, if $Loss^{MA/BA} = 10$ and $Loss^{Adaption}$ consists of two losses $Loss_1 = 1$ and $Loss_2 = 0.0001$, the second adaption loss will have only a negligible effect on the model's parameters since the value is already close to zero and the learning algorithm will try to minimize the other losses instead. Therefore, the underlying metric will not be adapted properly.

Definition of a strong adaptive adversary. We propose a *strong adaptive adversary*, who is able to adapt to multiple metrics simultaneously, independent of the value scales. Therefore, the adversary first scales all losses to the maximum loss value once (cf. λ values in Eq. 2). This has the effect, that all adaption objectives and the main task are considered equally. Afterward, the adversary can still weigh the adaption level via α.

$$Loss = \alpha \cdot Loss^{MA/BA} + (1 - \alpha) \cdot (\lambda_1 \cdot Loss_1 + \lambda_2 \cdot Loss_2 + \cdots) \tag{2}$$

Further, the adversary can simultaneously exclude specific parameters from training or replace parameters in the final model, e.g., with parameters of a previously benign trained model on the client's unpoisoned dataset, which we call *fixation*. The attacker can choose among multiple poisoning attacks, hence can use any existing method to embed a targeted poisoning attack in the local model. Additionally, advanced scaling methods and other classical model poisoning approaches can be applied.[5]

Regarding an adversarial-captured client, it is essential to recognize that the entire client device falls under adversarial control,

[4]To acquire a benign value, the adversary can train a benign model first.

[5]We provide results for attacks conducted by a strong adaptive adversary against FL defenses in Sect. 5.2 and discuss other adaption strategies that we evaluated in Sect. 6.1.

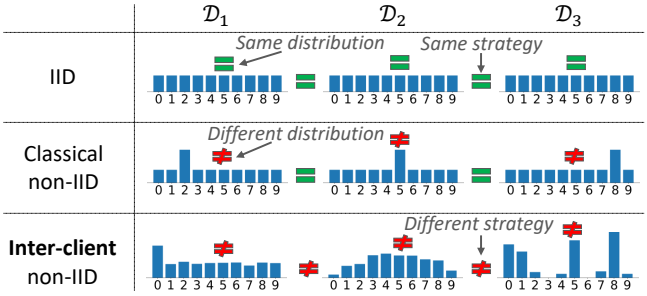

Figure 1: Comparison of various data distributions: IID, classical (intra-client) non-IID, and inter-client non-IID strategy for three client datasets $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$ with 10 label classes.

granting the adversary full access to employ any adaptation strategy. Additionally, the adversary can leverage any supplementary hardware resources, thereby eliminating the assumption of limited computational power on the adversary's device.

3.2 Inter-Client Non-IID

Below, we discuss the problem of varying data distributions in FL and define inter-client non-IID as a new challenge thereafter.

Problem. DF defenses in general inspect the clients' local model updates to detect abnormal situations based on the assumption, that the majority of clients are benign (cf. Sect. 3.1). Thereby, they leverage the fact that trained models' parameters reflect the characteristics of the underlying data as well as their distributions. It is easier to establish that models are similar if all clients possess similar data, e.g., there is the same amount of samples from each class in a classification task. This situation is called identically and independently distributed (IID) and is visualized in the first row of Fig. 1. In poisoning attacks, the underlying data need to change to introduce, e.g., backdoor behaviour, which inevitably manifests in changes in some parameters.

The second row of Fig. 1 visualizes the classical non-IID scenario, which is typically considered in the evaluation of backdoor defenses. Here, the data *inside* the client's local dataset (intra-client) are diverse, yet data distributions are similar across clients. Upon analysis of benign local models in this situation, they all will show a similar distance to the previous global model due to the similarity of distributions across clients. Existing DF defenses leverage this fact and can filter poisoned models, which are trained on a deviant data distribution due to data poisoning. However, defenses are not optimized for scenarios with different data distributions across clients, which we term *inter-client non-IID*. Such scenarios, as visualized in the third row of Fig. 1, are the most challenging to detect but also represent the most realistic real-world situation.

Definition of Inter-client non-IID. In *Inter-client non-IID* setting, the data within the clients' local dataset can follow arbitrary distributions inside and across the datasets without any assumptions made regarding sample frequencies or the availability of samples for a specific class. Thus, this definition also includes cases with disjoint data, as illustrated in row three of Fig. 1, where labels of classes 3 and 6 are not available within dataset $\mathcal{D}_3$.[6]

[6]We evaluate FL defenses in inter-client non-IID scenarios in Sect. 5.3.

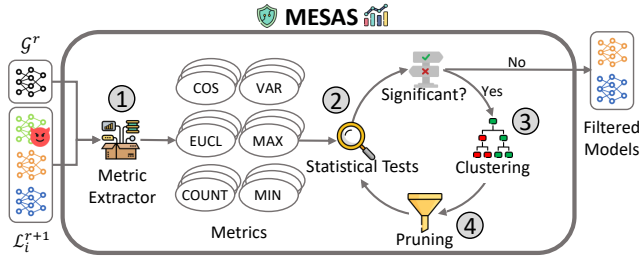

Figure 2: Overview of MESAS.

4 MESAS

In this section, we present our new defense against poisoning attacks, Metric-Cascades (MESAS). We first provide a high-level overview in Sect. 4.1, followed by explanations of the underlying intuitions in Sect. 4.2 and providing lower-level details in Sect. 4.3.

4.1 Overview

MESAS is a DF-based defense method which is applied on the central aggregation server before the aggregation step. To prevent strong adaptive adversaries from circumventing the defense, MESAS filters poisoned models in a cascade of six well-chosen metrics[7], that affect each other and cannot be optimized simultaneously, thus tightening the adversarial dilemma for the attacker. Further, MESAS analyses the six metrics with numerous statistical tests, thus allowing the defense to be effective also in inter-client non-IID scenarios and independent of the application scenario. Those statistical tests are also superior to hard thresholds in identifying scenarios without any attack and hence allow MESAS to not negatively affect the convergence of the federation. Moreover, the statistical tests utilized exhibit a higher level of effectiveness compared to threshold-based methods in accurately detecting scenarios without attacks. As a consequence, the integration of these tests into MESAS ensures that the convergence of the federation remains unaffected, thus preserving its overall performance and stability even if the defense is applied in every round.

In a nutshell, MESAS consists of four major steps that can be retraced in Fig. 2: 1) After the local updates have been transmitted to the server, MESAS extracts six carefully chosen metrics from the local models and the global model. Thereafter, those metrics are analyzed individually in an iterative process. The metrics are extracted for the whole model, but also from each layer individually, to detect poisonings distributed over the whole model, but also locally embedded ones[8]. 2) Each metric passes through a significance analysis consisting of statistical tests, that spot evidence of a poisoning attack within the metric values. 3) If indication is provided, the respective values are clustered into two clusters and the models belonging to the values within the smaller cluster are marked as malicious. 4) After each metric is analyzed, the marked models are excluded in a pruning step and the analysis starts over on the remaining models until no statistical test reports significant

[7]To the best of our knowledge 4-out-of-6 utilized metrics, namely COUNT, VAR, MIN, and MAX are novel and have never been considered in existing defenses.
[8]Naïve implemented backdoors are only embedded within the last few DNN layers. However, more sophisticated backdoors can reside within different locations, e.g., layers, inside the model parameters.

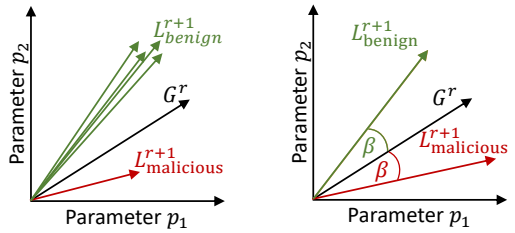

Figure 3: Simplified visualization of FL models with two parameters. The left graphic shows that benign and malicious models differ in one or multiple dimensions. On the right, we depict that benign and malicious models can have the same COS metric due to the same angel to the global model.

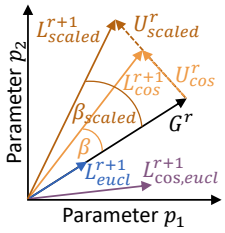

Figure 4: Visualization of locally trained models L_i^{r+1} deviating from the global model G^r in COS and EUCL. The figure also depicts how the angle β changes after scaling the update, thus provoking a change in the COS metric of MESAS.

evidence for an attack. Finally, the normal FL procedure continues with the remaining local models getting aggregated to the new global model.

4.2 Metrics Intuition

DNNs are complex multi-dimensional non-linear functions. An example of DNN with around eleven million trainable parameters is ResNet-18 [37]. For a better explanation of our metrics, however, we will use a simplified function, which is linear and only has two parameters (or dimensions): $f(x) = p_1 \cdot x + p_2$. With this, we can visualize model parameters p_1 and p_2 in a 2D plot (cf. Fig. 3), which won't be possible for a more realistic multi-dimensional function.

As visualized in the left graphic of Fig. 3, an adversary conducting a poisoning attack in FL needs to significantly change at least some model parameters of one or many poisoned local models in order to affect the behavior of the new global model. Otherwise, the respective parameter, and, thus, the new global model will align with the benign behaviour of the majority of clients (cf. Sect. 3.1) after aggregation. Hence, benign trained local models that learn similar behavior will be similarly distributed around the new global model after aggregation, since FedAVG decides for the average of all contributions. A malicious model, depicted in red color in Fig. 3, must be located in a significantly different location than the benign models depicted in green to influence the averaging of FedAVG.

MESAS is based on a set of six well-chosen metrics, that are extracted from local models. Technically, extraction of the metrics is a straightforward task that only needs to be conducted once for each local model within each FL round r. The metrics can identify malicious models or updates based on different characteristics, like *magnitude*, *direction*, *orientation*, *functionality level*, and *outliers*, which we will explain in detail in following.

Magnitude and Direction. The two metrics to detect deviations in magnitude and direction of benign and malicious models, which have also been used by other works [9, 29, 60, 65, 76, 109], are Euclidean distance (EUCL) and Cosine distance (COS) measured between the locally trained models L_i^{r+1} and the original global model of the round G^r. These metrics are depicted in Fig. 4.

Orientation. Two models with the same COS might significantly differ from each other, as depicted in the right graphic of Fig. 3, as COS alone is insufficient to reflect the direction. Therefore, the orientation of the Cosine from G^r can further differentiate two models. To incorporate this difference into a value, we propose COUNT, a novel metric that counts how many parameter values are increased from the respective parameter of the global model G^r during training. This metric provides a measurement to detect substantially different models, that exhibit inconspicuous similarities in COS. Moreover, the COUNT formula (cf. [40]) incorporates the *sign* function, which prevents straightforward adaption by adversaries. Specifically, attempts by adversaries to introduce an extra objective mirroring the COUNT formula into the loss function are rendered ineffective. The reason being that learning algorithms cannot effectively propagate changes to the underlying model parameters through a sign function, given its constant zero gradient. As a result, the utilization of this metric enhances the robustness of the system against adaptive adversaries.

Functionality Level. Due to the many parameters of a DNN, there can exist models with poisoned behavior, that have metrics COS, EUCL, and COUNT similar to benign models. Such a situation can occur, e.g., if the parameters of a model posses significantly different variance, as visualized in Fig. 5[9]. We leverage this variance as metric (VAR) in MESAS and interpret it as functionality level, since a different VAR is a clear indication of divergent model behaviour.

Outliers. As with any other variances, VAR is not affected by a few extreme outliers. Therefore, to catch those, we additionally investigate two more novel metrics: MAX and MIN, which extract the maximum/minimum parameter distance between all the parameters of local models L_i^{r+1} and a global model G^r.[10] VAR combined with MAX and MIN provide a reliable metric for the functionality level and allow testing for poisoned models. Similarly to the COUNT metric, the outlier metrics significantly enhance the system's resilience against adaptive adversaries. Specifically, the formulas for MIN and MAX (cf. [40]) can be incorporated as supplementary objectives in the loss function. However, it is noteworthy that the resulting changes are confined to the parameter responsible for reflecting the particular metric value. Consequently, other components within the model may undergo escalation in the metric while the actual outlier gets adjusted. This strategic attribute compels adversaries to employ additional measures, such as applying clipping mechanisms

[9] As highlighted in Fig. 5, the VAR can be increased, but of course also a significant decrease is possible.

[10] We take the minimum distance bigger than zero for MIN by leveraging a nonzero function (nz). Thus, MIN analyzes real model changes and ignores parameters that have not been changed.

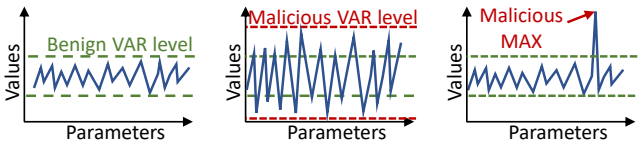

Figure 5: Simplified visualization of FL models with multiple parameters highlighting the functionality level based on the parameter value variance. The left shows a benign situation and the middle a poisoned model can have a bigger (or smaller) level. The figure on the right depicts, that the variance is not affected by maxima (and minima).

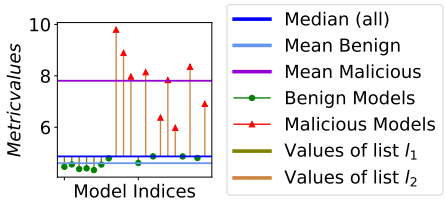

Figure 6: Depiction of a statistical test setup with significant p-value in ST-T indicating a varying mean between l_1 and l_2.

after model training is finished, to adjust remaining outliers in MIN and MAX to mount stealthy attacks.

Interrelations between metrics. The selection of the aforementioned metrics was based on their inherent interrelations. For an adaptive adversary attempting to adjust to the EUCL metric, success can be achieved through scaling or introducing additional objectives in the loss function. Both these approaches are likely to influence the COS metric. However, if the adversary adapts to the COS metric, they might exploit a stealthy situation as depicted in the right graph of Fig. 3. Nevertheless, such a scenario would have an instant impact on the COUNT metric. Furthermore, malicious behavior could be introduced by manipulating the variance of the model parameters, while remaining inconspicuous in terms of EUCL, COS, and COUNT metrics. However, the VAR metric would be capable of detecting such a situation. Further, a seemingly benign VAR constructed by employing extreme outliers, as visualized in the right graph of Fig. 5, would immediately generate abnormal values in MIN or MAX metrics. Due to the specific properties of certain metrics, namely COUNT, MAX, and MIN, which are non-trivial to adapt with additional objective functions[11], MESAS effectively counteracts adaptive attacks.

4.3 Pruning Loop

The filtering process consists of three steps: *statistical tests*, *clustering*, and *pruning* (2-4 in Fig. 2). In every filtering round, each metric traverses the procedure independently. After each round, the models filtered based on any metric are excluded from the next round. This iterative pruning loop continues until the statistical tests do not report any significance for the presence of a poisoning attack anymore. Due to the iterative nature of this filtering procedure and the individual analysis of each metric, different types of poisoning attacks can be filtered within one run of MESAS.

Statistical Tests. When provided with a set of metric values, which always contain one value per local model, the statistical tests first extract the median value, which is considered as benign due to the majority assumption (cf. Sect. 3.1). Afterwards, multiple statistical tests are conducted to check if all metric values are distributed equally around the median value, as one would expect from benign models. Therefore, MESAS checks if the metric values with bigger values than the median and the metric values with smaller values as the median follow the same distribution. For that purpose, the bigger and smaller metric values are converted to two lists l_1 and l_2 containing the absolute distance from the value to the median, as shown in Fig. 6. Then, the two lists pass through the tests. At first, a T-Test [52] (ST-T) is conducted to check for equal means. Since two distributions can have the same mean but different variances, a Levene's test [47] (ST-V) is appended. Finally, a Kolmogorow-Smirnow-Test [53] (ST-D) for equal distributions is leveraged. Following the same reasoning we provided for the metrics VAR and MAX, the aforementioned tests are not significantly influenced by outliers. Therefore, we additionally analyze the original metric values regarding the 3σ rule [72] (ST-3σ). Values outside the 3σ interval are marked as significant outliers.

In Fig. 6, the metric values of benign and malicious models are listed. The mean of all metric values (dark blue) is used to separate the values into two lists l_1 and l_2. Those lists represent the benign and malicious models, respectively, and are graphically observable by the lines between the metric values and the median. Note, that the median of the benign values (light blue) and the median of the malicious values (purple) have a significantly different distance to the median, which results in a highly significant result in ST-T. ST-T, ST-V, and ST-D deliver a p-value[12], which is also called significance level and is used to determine if a poisoned model is found.

Clustering and Pruning. After a significant statistical test (step 2 in Fig. 2), MESAS leverages Agglomerative Clustering [67] with two fixed clusters based on the Euclidean distance to cluster the significant metric values (step 3 in Fig. 2). Afterwards, the local models behind the metric values within the bigger cluster are considered as benign based on the majority assumption and the other models are marked as malicious and excluded by the pruning step of MESAS (step 4 in Fig. 2).

Overall, MESAS is robust against sophisticated poisoning attacks through an in-depth analysis of model weights using six interdependent metrics. As a result, if a strong adaptive adversary attempts to circumvent one metric, the artifacts of the poisoning attack will inevitably manifest through one of the other metrics. Further, MESAS adapts to the application domain including complicated non-IID data scenarios by leveraging statistical tests, instead of relying on hard thresholds. We provide the formulas of the metrics and additional information about MESAS in [40].

[11] We discuss this in the respective metric sections above.

[12] A p-value indicates how likely it is that the underlying data could have occurred under a null hypothesis. In our case, the null hypothesis is, that the two lists contain samples from equal distributions, thus having equal mean and variance.

5 EVALUATION

In this section, we conduct a rigorous analysis of MESAS and explore the impact of various parameters and application-specific factors like datasets, model architectures, underlying data distributions, poisoning methods, and attack adaptive strategies, as well as performance overheads.

5.1 Experimental Setup and Scenarios

Hardware and Software. We execute the FL system consisting of a configurable amount of clients on one server and implement the code in PyTorch [71, 93], a well-known machine learning library for Python [98].[13] The individual client and server code is executed sequentially on the server running with an AMD EPYC 7413 24-Core Processor (64-bit) with 96 processing units and 128GB main memory. An NVIDIA A16 GPU with 4 virtual GPUs each having 16GB GDDR6 memory is accessible via CUDA [68] from PyTorch.

Datasets and Models. We chose similar settings to FL defenses in related works and focus mainly on image classification with CIFAR-10 [41], GTSRB [89], and MNIST [21]. We use ResNet-18 [37], SqueezeNet [38], and a CNN model architectures. Additionally, we investigate into the text domain by training a DistilBERT [80] transformer model on SST-2 [87] sentiment analysis dataset.

Default Scenario. We train the CIFAR-10 [41] image classification task (ten classes) on a ResNet-18 [37] model with LR 0.01 (SGD optimizer, momentum 0.9, decay 0.005), a batch size of 64, and ten local training epochs. The federation is a realistic setup, which consists of $\mathcal{N} = 20$ clients, which are all selected each round r ($n = 20$). The data are IID distributed and each client has 2560 samples, 256 randomly chosen from each class. The adversary captures nine clients leading to a poison model rate (PMR) of 0.45, which is the maximum rate for this amount of clients. He sets the poison data rate (PDR) to 0.1, α to 0.3, utilizes the adaption strategies from Sect. 3.1 and implements a pixel trigger backdoor [32], which adds pixel pattern, a sticker, or similar as a trigger to the sample [5, 32, 51]. The global model G^r is already trained 50 benign rounds and was originally initialized with pre-trained weights from PyTorch, with the first and last layers being untrained since both needed to be changed according to our dataset.[14]

Defenses. We compare the following nine approaches, with MESAS regarding effectiveness and runtime, hence examine DF, RA, and IR methods: Naïve *clustering via HDBSCAN [54]*, *FoolsGold [29]*, *Krum [9]*, *M-Krum [9]*, *Flame [65]*, *T-Mean* [109], *T-Median* [109], *Clipping&Noising* [57], *Clipping* [57], and *Auror* [84]. We either adapted open-source implementations or reimplemented the methods if no code was available.

First, we consider our default scenario, and later we will expand the analysis to adaptive adversaries, nine poisoning attacks, and non-IID data scenarios. Due to space limitations, we report the most interesting results and numbers that highlight our outcomes in the following sections and list detailed experimental results in [40].

[13] In our setting, we create 20 clients, of which nine are captured by an adversary. Nine malicious clients are the maximum the attacker can control in our setup while remaining within our attacker model (cf. Sect. 3.1).

[14] The pre-trained models from PyTorch are trained on ImageNet [20], thus have other input dimensions and 1000 instead of ten classes.

5.2 Defenses under Strong Adaptive Adversaries

Before discussing defenses, we note that the BA of our default scenario without defense is only 42.94% (line 6 in Tab. 1), hence the backdoor is not effective (< 60%) and the adversary is forced to adapt his attack by either increasing the PDR, increasing the PMR[15], or by fixation, constraining and scaling. The increased BA of 61.96% for an increased PDR to 0.3 can be seen in scenario (1) in Tab. 1. We explore the effectiveness of these strategies and list results in [40]. Here, we show that MESAS is more effective than other defenses even without applying additional adaptions when comparing them under the default scenario as well as for increased PDR in scenario (1): As can be seen in line 17 in Tab. 1, MESAS effectively removes the backdoor by reducing BA to 1.85% and 1.95%, while most other defenses are less potent. Only FoolsGold [29] is as effective as MESAS in the default scenario and in scenario (1), but, as we will elaborate later in this section, FoolsGold could be easily circumvented through adaption.

Since the adversary has to use one of the adaption strategies to reach a higher BA, we want to clarify beforehand that an increased PDR reinforces already existing significant values in MESAS's metrics even more. Scaling of updates has positive effects on MESAS, since concurrently the metric COS will be changed, as visualized in Fig. 4[16]. Further, constraining with Eq. 1 or Eq. 2 also benefits MESAS due to side effects on its other metrics, forcing the adversary into a multi-objective optimization (MOO) problem and, thus, hardening the adversarial dilemma. Lastly, fixation methods are ineffective against MESAS, since all layers and the model as a whole are analyzed independently with statistical tests. Hence, MESAS is robust against adaptions of a strong adaptive adversary, which, we show, an attacker can leverage to circumvent other defenses.

5.2.1 Circumvent Defenses. Below, we will focus on the capability of defenses to reduce the BA in the new global model after aggregation compared to aggregation without defense (cf. line 6 in Tab. 1). Additionally, we will report the detection accuracy (ACC) of the defenses, when applicable, where 100% ACC means perfect detection rate and no False-Positives (FPs) and False-Negatives (FNs). We will also name the most effective adaption strategies based on results provided in [40], which we couldn't include in the main section of the paper due to space limitations.

Clustering. To demonstrate that naïve clustering methods could be bypassed, we use the HDBSCAN [54] algorithm as an example and cluster based on the cross-wise Cosine distances between model updates. As can be seen in the default scenario in line 7 of Tab. 1, the defense is ineffective reaching a BA of 74.62% in the new global model after aggregation. We additionally report an ACC of only 10% (FPR of 100% and 81% FNR). Thus, there is no need for an attacker to follow any adaption strategies. Nevertheless, adaption to naïve clustering is possible by increasing the PDR allowing us to embed a BA of 86.86%, as depicted in scenario (1).

[15] Our default scenario already includes the maximum valid PMR defined in Sect. 3.1.

[16] When scaling, our strong adaptive adversary is aware of benign values from training benign model first and scales to the mean of those values. Additionally, Gaussian noise is added to the targeted value within the 3rd percentile of the benign value range to make the malicious models slightly different and, hence, increase stealthiness (otherwise the models with exactly the same values could be easily detected).

Table 1: MAs and BAs in different scenarios in percent.

		Scenario											
		Default		(1)		(2)		(3)		(4)		(5)	
	Accuracies without defenses	MA	BA	MA	BA	MA	BA	MA	BA	MA	BA	MA	BA
1:	Global model G^r	62.99	1.90	62.99	1.90	62.99	1.90	62.99	1.90	62.99	1.93	36.51	5.18
2:	Average of benign local models	57.58	4.56	57.58	4.56	57.58	4.56	57.58	4.56	47.15	6.82	33.15	10.42
3:	Average of poisoned local models	57.84	85.13	54.58	93.15	54.42	93.25	51.23	89.82	43.74	91.32	33.93	82.00
4:	FedAVG with benign local models	63.57	1.85	63.57	1.85	63.57	1.85	63.57	1.85	65.92	1.40	32.45	12.71
5:	FedAVG with poisoned local models	64.92	83.00	63.68	92.50	62.29	93.71	40.69	93.54	59.12	95.50	29.35	88.96
6:	FedAVG with all local models	63.81	42.94	63.85	**61.96**	63.27	63.54	49.18	83.74	64.02	63.66	38.72	77.37
	Global model accuracies after applying defenses	MA	BA	MA	BA	MA	BA	MA	BA	MA	BA	MA	BA
7:	Naïve Clustering	65.06	74.62	64.75	86.86	63.73	88.36	47.34	85.58	61.12	87.58	20.85	**85.32**
8:	FoolsGold [29]	63.57	1.85	63.57	1.85	63.27	**63.54**	63.57	1.85	56.80	47.0	37.00	**76.03**
9:	Krum [9]	59.75	83.53	52.22	95.97	56.18	93.14	52.00	**89.90**	49.88	5.27	16.88	**89.07**
10:	M-Krum [9]	64.18	83.05	63.90	92.72	62.01	93.83	41.86	**95.80**	62.39	13.11	18.07	**89.55**
11:	Clip [57]	63.80	42.81	63.85	61.86	63.26	63.52	49.19	83.74	63.92	62.28	37.76	**75.60**
12:	Clip&Noise [57]	50.78	60.66	52.10	77.21	59.32	75.67	41.47	90.37	56.28	71.99	23.70	**64.32**
13:	Flame [65]	60.96	79.17	63.67	88.44	62.21	88.80	44.56	84.53	56.59	50.34	25.10	**79.17**
14:	T-Mean [109]	63.51	44.13	63.54	63.98	62.86	65.35	51.07	85.75	63.15	67.01	39.98	**76.36**
15:	T-Median [109]	51.22	44.60	51.18	57.73	49.61	60.30	39.76	74.76	51.75	68.20	17.04	**52.75**
16:	FLTrust [13]	63.49	23.08	63.76	49.68	63.17	45.54	55.15	**74.71**	63.56	8.40	26.81	**81.61**
17:	**MESAS**	63.57	**1.85**	63.36	**1.95**	63.36	**1.95**	63.57	**1.85**	65.92	**1.40**	37.52	**2.37**

(1) Default + PDR 0.3
(2) Default + PDR 0.3 + Last Layer Fixation to benign models
(3) Default + Adapt to EUCL of benign models
(4) Default + PDR 0.3 + 1-class intra-client non-IID with $q = 0.5$ + Scaling
(5) Default + inter-client non-IID based on our Random-Non-IID strategy

FoolsGold. The second defense, FoolsGold [29], is also based on cross-wise Cosine distances between model updates. However, it analyzes only outputs of the last layer, which is more effective than naïve clustering and is capable of removing all poisoned models in the default scenario and for scenario, (1) reaching BAs of 1.85%, as depicted in line 8 of cf. Tab. 1. Nevertheless, the defense can be circumvented using adaption. The best results we obtained by parameter *fixation* on the last layer in combination with PDR increase, depicted as scenario (2) in Tab. 1, reaching a BA of 63.54%. In contrast, MESAS still removes the backdoor to 1.95% with only one FP when a similar adaption strategy is applied.

Krum. Next, we evaluated Krum and M-Krum [9], which leverage cross-wise Euclidean distances between local models. The trigger backdoor is not reflected in this metric, which renders the defense ineffective for our default scenario (83.53% and 83.05%BA for Krum and M-Krum, resp. in Tab. 1) and for scenario (1) and (2). Since Krum selects one single local model as the new global model, it can either choose a malicious or benign local model. In the former case, the backdoor trivially makes it to the global model. In the latter case, we can follow the following strategy: We can adapt the malicious models via constraint method Eq. 1 forcing the Krum scores of poisoned models to be more equal to each other compelling Krum to decide in their favor. By circumventing Krum like this, we achieved BAs up to 89.90% and reached 95.80% BA for M-Krum as can be seen in scenario (3) in Tab. 1. In contrast, MESAS accurately filters the backdoor in similar circumstances, as adaption via constraint has significant effects on other metrics, like EUCL and MIN.

Flame. We evaluate Flame [65], a more complex DF defense, which combines clustering with clipping and noising techniques. Since the underlying metric is the same as for the naïve clustering defense, it is not very effective in removing the backdoor even in the default scenario achieving 79.17% BA (cf. line 13 in Tab. 1). Similar to naïve clustering, we could strengthen the BA by increasing the PDR to 88.44% and by additional scaling to 91.34%, which shows that relying solely on the leveraged metric of Flame is insufficient. MESAS erases the backdoor efficiently in all of the cases, due to the in-depth model analysis with statistical tests and increased robustness against adaption through leveraging six different metrics.

FLTrust. As a more recent defense, we analyze FLTrust [13], which is based on a trusted root dataset[17] on the server side. FLTrust leverages the Cosine Similarity between the updates of the local models and a trusted model trained by the server on the trusted root dataset and the norm of the local updates. Based on these metrics, FLTrust assigns weights to each local update so that poisoned updates are assigned with low weights, preferably zero, which would filter out the update. Therefore, the defense is ineffective, if the backdoor is not visible within both of these metrics, meaning, that the Cosine Similarity is inconspicuous, which can happen if the backdoor is only embedded in one layer without affecting the model-wise metric value, or if the backdoor is hidden in other metrics, as VAR, MAX, or MIN only. In most of our experiments, FLTrust successfully weakened backdoors beneath critical BAs. However, the assigned weights to all (also benign) local updates were found to be relatively small (mostly between 0.001 and 0.03), thereby inadvertently reducing their contribution to the global model. Consequently, the approach's efficacy comes at the cost of slowing down the training speed. Additionally, FLTrust's effectiveness depends on the chosen metric's ability to accurately reflect the backdoor. However, a backdoor is not necessarily embedded in those metrics, as can be seen in experiments, e.g., in scenario (3) yielding a BA of 74.71%. For adaption, we first trained a benign model and then proceeded to adapt the local update of the malicious model based on the update of this benign model. Since we observed that in the resulting

[17] Similar to the authors, we used a trusted root dataset of 100 IID samples and excluded them from the datasets used for training of the clients.

Table 2: BA for targeted and ACC for untargeted poisoning attacks without adaptive adversary in percent.

	Aggregation / Defenses	BA						ACC		
		Pixel Trigger [32]	Clean-Label [96]	Semantic [5]	Edge Case [100]	Label Flip [8, 12]	Pervasive [16]	Random Flip [40]	Sign Flip [40]	Noising [40]
1:	Global model G^r	1.90	1.90	0.00	1.53	0.10	0.02	-	-	-
2:	Average of benign local models	4.56	4.57	0.00	2.55	1.24	0.95	-	-	-
3:	Average of poisoned local models	85.13	75.49	80.0	19.28	74.15	97.28	-	-	-
4:	FedAVG with benign local models	1.85	1.85	0.00	1.85	0.20	0.07	-	-	-
5:	FedAVG with poisoned local models	83.00	81.75	100.0	20.40	71.20	99.84	-	-	-
6:	FedAVG with all local models	42.94	38.92	60.0	6.63	49.20	3.58	-	-	-
7:	Naïve Clustering	74.62	**1.85**	60.0	16.35	65.60	67.67	10.00	**100.00**	80.00
8:	FoolsGold [29]	**1.85**	**1.85**	**0.00**	**2.55**	**0.20**	**0.10**	55.00	**100.00**	0.00
9:	Krum [9]	83.53	75.65	80.00	20.91	**1.30**	**0.42**	50.00	50.00	50.00
10:	M-Krum [9]	83.05	82.38	100.0	18.87	**0.40**	**3.50**	75.00	75.00	75.00
11:	Clip [57]	42.81	38.91	60.0	**6.63**	48.40	**3.17**	-	-	-
12:	Clip&Noise [57]	60.66	40.73	**0.00**	12.75	30.80	10.08	-	-	-
13:	Flame [65]	79.17	77.12	60.0	18.87	**2.40**	**5.52**	**100.00**	**100.00**	**100.00**
14:	T-Mean [109]	44.13	41.10	60.0	**7.14**	48.40	**2.53**	-	-	-
15:	T-Median [109]	44.60	25.66	**0.00**	**2.55**	**5.60**	**0.10**	-	-	-
16:	FLTrust [13]	23.08	37.83	**0.00**	**5.10**	**0.2**	**0.11**	60.00	20.00	35.00
17:	**MESAS**	**1.85**	**3.71**	**0.00**	**2.55**	**0.20**	**0.05**	**95.00**	**100.00**	**100.00**

models, the main reason for suspicious metric values in Cosine Similarity originated from the parameters of the last model layer, we restricted this adaptation process to the last layer only to make the backdoor inconspicuous in the last layer. While this strategy resulted in low BA, FLTrust assigns higher weights to the malicious updates than to the benign ones, with seven out of eleven benign updates being assigned a weight of zero. That means that those benign models were filtered, essentially slowing down the learning process, while malicious models were included in aggregation, even so with smaller weights. In contrast, MESAS consistently and effectively eliminated the backdoor in all of these cases without decreasing the impact of benign models.

Differential Privacy. Besides DF methods, we evaluated two IR approaches: Model update clipping based on the Euclidean distance and a combination with model parameter noising [57]. Clipping is ineffective, as our default scenario backdoor is not reflected in the Euclidean distance of the updates. Thus, the attacker can achieve 60.66% BA for the default scenario (cf. line 12 of Tab. 1). When using adaption, the BA can be increased slightly to 61.86% by increasing the PDR as in scenario (1). In contrast, MESAS is effective under similar circumstances resulting in 1.85% and 1.95% BAs.

Robust Aggregation. We evaluate T-Mean and T-Median [109], which are RA alternatives to FedAVG. Both result in weak backdoors with BA of 44.13% and 44.60% in lines 14 and 15 for the default scenario, but are not robust when facing a strong adaptive adversary: T-Mean can be bypassed with up to 63.98% BA, while T-Median shows 57.37% BA, but also experiences around 10% reduction in MA in scenario (1). Hence, both approaches are not comparable to the performance of MESAS, which reduces BA to 1.95% under similar circumstances.

MESAS. To circumvent MESAS, we tried to adapt to respective metrics that reflect the different poisoning attacks. We succeeded in adapting to COS, EUCL, MIN, and MAX, which appeared to be the metrics most backdoors manifest first. This was only possible by leveraging the loss scaling method of our strong adaptive adversary, as described in Sect. 3.1, since otherwise, adaption to multiple losses already resulted in facing an adversarial dilemma. However, as soon as we adapt to those metrics, this behavior is reflected in the other metrics, namely VAR and COUNT. For a few experiments, we succeeded in adapting to VAR, even if the MA suffered immensely, but additional adaption to COUNT was impossible.

5.2.2 Different Poisoning Attacks. In the following, we evaluate the effectiveness of the defenses against various poisoning attacks, including six different trigger methods for targeted attacks and three untargeted attacks, namely pixel triggers [32], clean-label backdoor [96], semantic backdoor [5], edge case backdoor [100], label flip backdoor [8, 12], and pervasive backdoor [16] as well as random label flipping, sign flipping, and model noising, which are all explained in detail in [40]. We report the BAs that the poisoning attacks achieve against the nine defenses in Tab. 2 and the MAs in [40].[18]

Pixel Trigger Backdoor. This backdoor is discussed in Sect. 5.2.1, where we showed that we can circumvent existing defenses by adaption and strengthening the trigger. Only MESAS could reliably remove the backdoor.

Clean-Label Attack. This attack is not suited perfectly for FL, since it is hard to embed a high BA with low PDR into the new global model. In our default scenario, we reached only 11.85% BA after aggregation, which is why we report the result for PDR 0.5, which leads to a BA 38.92% without defense (line 6 in Tab. 2). Nevertheless, it is possible to achieve a high BA of up to 82.38% for M-Krum (line 10), while naïve clustering, FoolsGold, and MESAS erase the backdoor. Among them, MESAS is the only one that cannot be adapted and erases the backdoor, which manifests in COS and EUCL, resulting in a FNR of 81%.[19]

[18]The results reported in tables do not consider adaption (which is evaluated in Sect. 5.2.1), as the system's adaptability is directly tied to the specific defense employed. Instead, the tables focus on determining whether MESAS can successfully detect various triggers. Thereby, we ensure that our findings remain independent from the pixel-trigger method of the default scenario.

[19]We experienced an elevated FNs in a scenario with a maximum PMR and one benign outlier model. We could not reproduce such scenarios on purpose when acting as an adversary. Such scenarios can only occur, if the PMR is at a peak of nearly 50% and one benign outlier exists, which then violates the majority assumption of Sect. 3.1. However, if such situations occur, MESAS still ends up aggregating only benign models

Semantic Backdoor. Without defense, this backdoor is effective with 60% BA. However, it is detectable within the last layers by FoolsGold [29] leading to 0.00% BA (line 8 of Tab. 2). Clip&Noise and T-Median also remove the backdoor, but at the same time reduce MA. MESAS erases the backdoor completely by leveraging MAX metric. We report one FP in this case for MESAS, but with a good result in a BA of 0.0%. Other effective defenses can be circumvented through adaption (FoolsGold) or reduce the MA (T-Mean and Clip&Noise).

Edge Case Backdoor. It appears to be hard to embed an effective backdoor with this method even within the local models for CIFAR-10 [41] on ResNet-18 [37]. In Tab. 2, we report the results for a PDR or 0.3 with 19.78% BA on the local clients on average (line 3) and 6.63% BA without defense. MESAS is already sensitive to the poisoning attacks even when the effect on the global model is still minimal with 6.63% BA (line 6). We reach 100% TPs and only two FPs resulting in the lowest BA with 2.55% in this case (line 17).

Label Flip Backdoor. This attack manifests in extreme deviations within the last layer of a DNN. Hence, many defenses can easily detect the backdoor, as can be retraced on the low BAs in Tab. 2. Having two FPs, MESAS is the only defense reducing the BA to 0.20% while being robust against fixation and adaption attempts, which can be used to circumvent other defenses like FoolsGold.

Pervasive. Blend [16] can be implemented with a PDR of 0.1 to achieve 99.84% BA locally on average (line 3 in Tab. 2), but it is inefficient in FL– we could only reach 3.58% BA for the global model without defense (line 6). MESAS can detect all poisoned local models while suffering five FNs. The result is interesting, as it shows that MESAS reaches the lowest BA of 0.05% while having minor effects on the MA, whereas other defenses affect the MA (cf. [40]) or can be circumvented by adaption.

Untargeted Attacks. For the untargeted attacks, we do not report the BAs, but the ACC of the defense mechanisms in Tab. 2 and the resulting MAs in [40]. Random label flipping is the first untargeted attack that we implemented. The MA is reduced to 57.03% without any defense and only M-Krum and FLTrust can score a higher MA of 64.15% and 63.16%, respectively, compared to 62.88% of MESAS. However, M-Krum suffers a FNR of 45%, compared to 0.09% of MESAS and FLTrust comes with an ACC of 60%[20], while MESAS achieves 95%. Flame stands out with 100% ACC, but can be circumvented by adaption. Second, we evaluated sign flipping, which is clearly detectable by defenses leveraging clustering methods including MESAS, but can lead to a naïve model with 10% MA for other approaches. Finally, we report the results for the model noising attack, where MESAS also has an ACC of 100%, whereas other methods, like FLTrust achieve only 35% ACC.

Concluding, we can say, that MESAS is robust against nine poisoning attacks executed by a strong adaptive adversary, who is able to intentionally circumvent all other nine evaluated defenses. We argue that any other defense, that relies on just a few metrics, could be similarly bypassed in our strong adaptive adversary model, by either fixation or constraint methods.

as long as the poisonings are significant in at least one metric in one layer. Hence is also robust against coincidental benign outliers.

[20]We compute the ACC of FLTrust [13] by analyzing the weights assigned to the models. A model assigned with a weight of zero is considered as a filtered model.

5.3 Defenses under Non-IID

Here, we evaluate the nine defenses under classical intra-client non-IID before we discuss inter-client non-IID.

Intra-client non-IID. We analyzed various intra-client non-IID settings, namely 1-class, 2-class, and Distribution non-IID. For the 1-class and 2-class non-IID scenarios, the samples of a client's dataset focus on one or two so-called *main labels*. The remaining labels contain an equivalent amount of samples, while a factor $q \in [0, 1]$ defines the fraction between samples within the main label class and the remaining classes[21]. Distribution non-IID assigns label frequencies for each dataset based on a distribution, e.g., Dirichlet [58] or normal distribution. We elaborate on non-IID simulation techniques in more detail in [40]. As representative results, we present intra-client non-IID based on 1-class with $q = 0.5$.

We notice that in non-IID settings it is harder for the adversary to embed a backdoor due to the nature of FedAVG. To reach a reasonable BA of above 60%, the adversary must use adaption strategies. We find that increase of PDR to 0.3 combined with scaling reaches reasonable performance with 63.66% BA (scenario (4) in Tab. 1). Krum and M-Krum [9] erase the backdoor, but simultaneously reduce the MA. However, after an adaption, we can circumvent those defenses reaching BAs of up to 90.44%, while still erasing the backdoor with MESAS. FoolsGold is effective, but can be circumvented by adaption, while FLTrust also decreases the BA to 8,40%, but assigns weights for update aggregation between 0.0 and 0.025 to all model updates, effectively removing the influence of most of the update. MESAS is the only defense erasing the backdoor in this setting and reaching 1.40% BA with two FPs. Hence, MESAS outperforms other defenses in intra-client non-IID settings.

Inter-client non-IID. To simulate even more realistic datasets, we designed the *Random-Non-IID* strategy (cf. [40]). Thereby, we randomly decide which label is contained in a client's dataset and also randomly assign the label frequencies. This results in inter-client non-IID datasets even with disjoint data. Other works do not normally consider such scenarios in evaluations and we hope, that this strategy will be adopted in future research.

We report the results for a Random-Non-IID setting[22] after 50 benign rounds of FL training with 20 clients in the federation in scenario (5) in Tab. 1. It is very easy for an adversary to embed a backdoor in such scenarios, thus reaching a BA of 77.37% without defense, as can be seen in line 6. Among all defenses, MESAS is the only one capable of erasing the backdoor by decreasing the BA to 2.37%, while others provide BAs between 52.75% and 89.55%.

We repeated this experiment in FL round one[23] of this setting to analyze the dependence on an already converged model and within round 50 of a setting containing 100 federation clients from which 20 are selected randomly for each FL round, and got similar results with MESAS outperforming other defenses, that are not capable of removing backdoors in inter-client non-IID scenarios. (cf. [40]).

5.4 Influence of Parameters on MESAS

To evaluate the influence of various parameters on MESAS's performance, we first investigated training hyper-parameters and showed

[21]For $q = 1$, all samples are from the main label. $q = 0$ is equal to the IID scenario.

[22]The sample frequencies for each client of the scenario are listed in [40].

[23]Early round backdoors are not persistent (cf. [5]), but we still analyzed the situation.

the independence from random seed, LR, PMR, and the selection of α. We found no unexpected results that are much different from our default scenario. We report on these experiments in [40].

Poison Data Rate. Our experiments show, that the backdoor efficiency depends on the type and composition of the trigger, but also the PDR is important. We evaluated $pdr = [0.1, 0.2, ..., 0.9]$ and selected the smallest value $pdr = 0.1$ that allows an adversary to introduce an effective backdoor in our default scenario. This naturally makes the resulting local models most stealthy by scoring a high MA. During some experiments, we increased this value up to 0.3 to reach a high BA. For bigger PDRs, MESAS was also able to eliminate the backdoor with ACC 100%. This highlights the adversarial dilemma, since higher PDRs could increase the BA, but are not stealthy, urging the adversary to adapt to defenses, which has side effects on the metrics of MESAS, forcing the adversary in an even more complex MOO problem. Concluding, we can claim, that MESAS is independent of the PDR selected by the adversary.

Initial Global Model. We conducted experiments with different pre-trained models. We used random initialized models and pre-trained models from PyTorch [71, 93] where we changed the first and last layer according to our dataset. We then trained the models in benign settings with 20 clients in the federation, all participating in each round as well as with 100 clients in the federation whereof 20 contributed each round. MESAS performed well in all of the cases and can be used independently of the FL round. However, the detection performance in later rounds is naturally more accurate, since even benign clients can strive towards a different minimum on a relatively naïve model. Nevertheless, even in inter-client non-IID settings, MESAS erases backdoors in early rounds reliably (cf. [40]). Principally, MESAS is designed to be applied in every FL round and does not impose a negative impact on the convergence of the federation when no attack is present. The rationale behind this lies in MESAS's ability to effectively distinguish between attack-free and attack scenarios by virtue of its robust statistical tests.

FL Round. To emphasize the effectiveness of MESAS, we conducted experiments where models were trained 100 rounds starting from a randomly initialized model until the model converged and apply defenses after every training round. We visualize the performance of various defense mechanisms and scenarios with no defense and no attack in [40]. Notably, the results demonstrate that MESAS surpasses other defense approaches by reaching BA and MA levels comparable to the attack-free scenario. This underscores the robustness of MESAS in mitigating the impact of backdoor attacks.

Dataset. We exchanged the dataset to MNIST [21] and GTSRB [89] and could assert, that the results and thus the defenses' performance including MESAS does not vary across datasets. MNIST as a more basic dataset, simplifies the detection of backdoors for all defenses even if a stealthy backdoor itself is hard to implement without defense, whereas GTSRB is more complex due to more label classes. We report the results for one of our MNIST experiments with one FP and one GTSRB experiment with 100% ACC in [40].

Model Architecture. We conducted experiments to analyze the model architecture independence. We used a CNN with two convolutional layers concatenated with pooling layers and ReLu functions [1] followed by three fully connected layers and trained on MNIST [21]. Further, we tested SqueezeNet [38] with CIFAR-10 [41] and can report 100% TNs with just one FN in both cases (cf. [40]). Hence, MESAS is independent of the architecture of the model.

Table 3: Defense runtimes in seconds.

Defense	Runtime	Defense	Runtime
FedAVG	0.12	Flame [65]	7.92
Naïve Clustering	7.57	T-Mean [109]	7.12
FoolsGold [29]	0.14	T-Median [109]	0.26
Krum [9]	6.02	Auror [84]	12 hours
M-Krum [9]	5.92	FLTrust [13]	25.12
Clip [57]	2.37	**MESAS**	24.37
Clip&Noise [57]	2.52		

Application Domain. We conducted experiments within the text domain training a sentiment analysis task using the SST-2 [87] dataset on a DistilBERT [80] transformer model. We implemented a backdoor, that labels sentences starting with the term "Hey!" as negative. We can report 100% ACC in this experiment, showing the applicability of MESAS in different application domains and for model architectures that do not contain convolutional layers.

5.5 Runtime Evaluation

We evaluate the runtime of the defenses to verify the real-world applicability of MESAS. Tab. 3 lists average runtimes of ten runs for our default scenario and shows that MESAS introduces an acceptable overhead of 24.37 seconds. FoolsGold [29] comes along with outstanding performance since only one model layer is analyzed, but due to the same reason, it can be easily circumvented (cf. Sect. 5.2). Further, T-Median [109] replaces FedAVG with a simple algorithm, which results in a similar runtime, but reduces the MA. Auror [84] instead, has an unacceptable runtime of 12 hours to calculate the indicative features due to massive clustering, which is why we excluded this approach from evaluations in Sect. 5. FLTrust [13] is dependent on the size of the trusted dataset, as a trusted model is trained on the server. Defenses leveraging client feedback [3, 113] cannot compete with server-side-only defenses, due to additional communication overhead.

6 DISCUSSION

We discuss alternative adaption methods that were tested in Sect. 6.1 followed by limitations and future work suggestions in Sect. 6.2.

6.1 Adversarial Adaption Methodologies

Besides the final method of our strong adaptive adversary (cf. Sect. 3.1) that we used to evaluate FL defenses in Sect. 5.2, multiple alternatives have been tested during this work.

First, we just added all of the losses (λ's from Eq. 2 equals one), which is similar to classic adaption (cf. Sect. 2.2). As explained in Sect. 5.2, losses with a drastically smaller scale than others have barely influence in the optimization, thus the related metric is not adapted. Second, we scaled all losses to $Loss^{MA/BA}$, which would be reasonable, if the MA is the major concern of A. However, most defenses including MESAS do not check the MA lacking a test dataset in realistic scenarios, which makes scaling to the maximum the better choice. Third, we tested, how often the λ's should be recalculated. Only one initial computation delivers the best results.

This seems reasonable, since then already optimized metrics have a minimal loss value and thus barely influence the optimization.

Additionally, motivated by Multi-Objective Optimization (MOO), we searched a pareto optimal [14] solution with Sener *et al.* [81]. However, the method produced broken models regarding the accuracies. We believe the reason is, that Sener *et al.* consider a system comparable to Multi-Task Learning (MTL) where both, shared and task-specific parameters exist within a model. However, our MOO problem optimizes only shared parameters (the whole model).

Since our final adaption method is superior to classic adaption [5] from Eq. 1, we claim, that MESAS is robust against adaptive adversaries caused by the introduced adversarial dilemma by forcing the adversary into a MOO problem with seven losses of different scales.

6.2 Limitations and Future Work

MESAS's major limitation is the significance niveau for the statistical tests influencing good TPRs and TNRs. Throughout experiments, the values appeared to just depend on data scenarios. Hence, it can be required in so far unseen tasks to adapt the values. An automatic method for setting the values can be discovered in future work.

As any other poisoning defense for FL, MESAS can be tested against other aggregation mechanisms besides FedAVG and can be combined with IR methods, similar as in FLAME [65] and DeepSight [76]. With such an extension one can soften the significance thresholds to lower the FNR to zero and simultaneously reduce the influence of the models responsible for the resulting FPR.

We leverage COUNT (combined with COS and EUCL) to get the direction of the model update. Fortunately, the metric is hard to adapt due to the sign function involved in the computation. Nevertheless, other metrics with the same effect can be discovered in the future. Additionally, one can investigate into the Cosine distance of the client updates among each other instead of the Cosine distance with respect to the global model G^r, which could provide additional information about the direction.

As shown in experiments, the strong adaptive adversary from Sect. 3.1 cannot circumvent MESAS. However, research can be conducted to find currently unknown methods to better adapt a DNN to multiple metrics simultaneously, which falls in the area of MOO. If such an method exists, MESAS can be extended to e.g. investigate in the correlation coefficient between updates additionally.

7 RELATED WORK

In this section, we first discuss existing poisoning defenses in Sect. 7.1, before we address privacy issues in Sect. 7.2.

7.1 Defenses against Poisoning Attacks

Auror [84] is a K-Means [2] clustering approach based on indicative differences between individual model parameters. It decides for each parameter if it is indicative for clustering the model updates into a benign and a malicious group and analyzes the resulting clusters. Due to multiple clustering steps (increasing with bigger model architectures), the defense suffers a high runtime overhead. Further, Auror has problems finding multiple backdoors simultaneously and shows poor performance in non-IID settings. MESAS utilizes a lightweight feature extractor and prunes different poisonings in an iterative process independent of the data distributions.

FoolsGold [29] weights each local model's contribution, by analyzing the cross-wise Cosine distances between model updates of the last DNN layer, thus being prone to adaptive adversaries that fixate this layer. Further, the approach assumes only non-IID settings and poisoned local models that point in the same directions (so-called sybills) and it leverages updates from previous rounds for optimal performance. Instead, MESAS prevents adaption by relying on a metric cascade and analyzing layers individually and is effective in IID and non-IID settings independent of the FL round.

Krum [9] is based on the Euclidean distance between local models. For each local model, it aggregates the distances to its neighbors and selects the one with the densest surrounding as new global model. M-Krum [9] selects more models simultaneously. Both can be circumvented by adaption to the metric and naturally suffer a high FNR without any adversaries in the system. MESAS does not harm the federation in total benign scenarios and provides a low FNR while not being susceptible for adaption attempts.

AFA [60] leverages plain analysis of the Cosine distance between local models, which is adaptable with an additional loss. MESAS hardens this possibility by leveraging a cascade of six metrics.

Naïve clustering approaches, e.g. based on HDBSCAN [54], need to extract a metric like the Cosine distance between models from the local models to reduce the dimensions. Hence, adaptive adversaries can circumvent the defenses, which is harder in MESAS. Further, clustering relies on a majority assumption and creates two groups, thus having a hard threshold and a high FNR in settings without attacks. MESAS leverages statistical tests with probabilistic thresholds that adapt to the scenario and investigates metrics that are hard to adapt due to fine-grained values resulting in lower scale adaption losses than typical clustering metrics.

BaFFLe [3] first aggregates all local models to a new global model (thus being an IR approach) and then sets up a client feedback loop, where the previous and the new global models are sent to validation clients introducing communication overhead. Those clients analyze the per-label MA and mark the new model as malicious if an empirically chosen threshold is violated. If so, the whole round is discarded. Further, the first 800 rounds are assumed as benign, so that a valid global model is available as a reference. Since adversaries strive to an inconspicuous MA (see Sect. 2.2), this approach fails for sophisticated adversaries. Further, one single adversary can force the defense to discard all other benign contributions of the round. MESAS runs on the server side only, prunes poisoned models, and is effective even in the first round of FL. Similarly to BaFFLe, the approach of Zhao *et al.* [113] leverages a client feedback loop to analyze the MA of the local models on the client side, thus introducing an even bigger communication overhead, while keeping the downsides regarding inconspicuous MAs. Further, this approach is prone to privacy issues, since inference attacks (cf. Sect. 7.2) can be conducted on the local models on the client side.

FLAME [65] is a combination of DF and IR. The approach clusters local models by pairwise Cosine distances via HDBSCAN [54] and filters adversaries based on the majority assumption before differential privacy methods [22] are leveraged. Precisely, weight clipping (regarding the median Euclidean distance of the updates) is applied to the remaining local models, and noise is added to the aggregated model. Besides the desired decrease in BA, this step naturally decreases the MA, too. When adapting to the Cosine and

Euclidean distance simultaneously, the approach performs similarly to a plain noising mechanism [92], which can also be applied to any other DF. MESAS leverages six metrics to harden the adversarial dilemma during adaption attempts and does not solely rely on clustering, but on statistical tests allowing a more fine-grained analysis of the local models. Further, any IR approach can be combined with the defense easily, but MESAS does not decrease the MA naturally.

Similar to the concept of Krum [9], Yin *et al.* [109] uses the coordinate-wise median or mean of the local models to construct the new global model based on the majority assumption. These approaches called Trimmed-Mean and Trimmed-Median respectively are RA mechanisms, but reduce the MA compared to FedAVG. Especially, the parameters and thus the functionality of benign model models lying not centrally within all updates not be considered. Bagdasaryan *et al.* [5] and Sun *et al.* [92] already proposed update clipping and nosing techniques, but Naseri *et al.* [61] showed, that differential privacy methods not only naturally harm the MA [5], but also can boost the BA when applied to benign FL clients. All of the IR approaches and most RA methods suffer a drop in MA, especially in a setting without attack. MESAS instead, filters poisoned models, and thus does not influence benign scenarios naturally. Further, IR and RA methods can be easily combined with MESAS to get an even more bulletproof global model.

DeepSight [76] is a more complex strategy, which combines filtering with differential privacy and is based on two metrics. First, the Cosine distance between models, which can be adapted by an additional loss (cf. Sect. 5.2). Second, two more values are extracted from the output layer, which can be circumvented by fixation, as shown for FoolsGold [29] in Sect. 5.2.1. Therefore, DeepSight is not robust against strong adaptive adversaries and relies on clipping and noising techniques, that reduce the MA and can also be applied to any DF approach. MESAS instead forces the adversary into a hard optimization problem and does not rely on specific layers.

FLTrust [13] assigns weights to updates during aggregation, essentially filtering out updates with a weight of 0. To determine the weights, it assumes a benign dataset on the server side for a trusted reference model and relies on Cosine Similarity between the updates from the client side and the trusted update as well as the norm of the local updates. However, FLTrust's reliance on those two metrics makes it susceptible to adaptability by adaptive adversaries. Since FLTrust examines the metric across the entire update, it may also fail to detect attacks that manifest only at the layer-wise level. In contrast, MESAS operates independently without a trusted dataset, conducting layer-wise analysis and utilizing multiple interconnected metrics, enhancing MESAS's robustness against a broader range of adversarial attacks.

FLDetector [111] is a historical update-based defense. It maintains records of global and client updates, predicting client updates using mathematical approximations. These predictions are compared to actual updates. After a warmup phase, outliers are detected using clustering methods by evaluating the normed Euclidean distance between predicted and actual models. However, FLDetector is storage and runtime intensive, depending on historical update time windows. In contrast, MESAS doesn't rely on historical updates and doesn't need a warmup phase. This characteristic makes MESAS more versatile and advantageous over FLDetector.

BayBFed [42] constructs a statistical model of update parameter distributions for each client update, which adapts with each FL round. Using clustering, a single value per update is computed, which are then used to construct a filtering threshold based on the values' mean that allows model filtering. However, BayBFed wasn't tested in an all-benign scenario, potentially causing a high FPR due to benign model filtering. In contrast, MESAS avoids the introduction of a hard value threshold. Instead, it leverages statistical tests, ensuring a low FPR even in scenarios without any attacks.

7.2 Privacy Preserving Federated Learning

FL in its original form [55] improved the privacy of collaborated DNN training compared to a data-centralized, since raw sensitive data do not leave the client side anymore. Nevertheless, membership inference [36, 49, 73, 85, 85], label inference [112], property inference [30], model extraction [49], and data reconstruction [79, 102] attacks as well as others [101] can be conducted on both, mainly the local models but also on the global model. Therefore, especially the devices with access to the local models, namely the aggregation server, still needs to be trusted (cf. Sect. 3.1).

PPFL [59] ported the FL process into a Trusted Execution Environment (TEE). The approach assumes the availability of a TEE on the client side and introduces computational overhead, since execution speed in e.g. SGX [18] enclaves is reduced, mainly due to page swaps based on limited memory. Additionally, such approaches based on secure code execution [69, 74, 95, 99, 115] either on CPU only or on CPU and GPU hinder model poisoning attacks on the client side, but do not prevent data poisoning.

On the server side, Hashemi *et al.* [35] implemented Krum [9] in a TEE. Such a secure aggregation method solves privacy issues allowing the threat model to exclude the aggregation server $\mathcal{S}$ as trusted party. Implementing MESAS within a TEE is just a technical barrier. Though, additional privacy results in increased runtime.

Overall we conclude, that MESAS is complementary to privacy-preserving FL techniques.

8 CONCLUSION

Federated Learning (FL) faces significant challenges, notably adversarial adaptation and complex data scenarios. To underscore the urgency of addressing these issues, we conducted a rigorous evaluation against a s*strong adaptive adversary*, capable of various poisoning techniques and operating without assumptions about client dataset sample frequencies (*inter-client non-IID*). Our findings reveal that existing defenses, when confronted with adaptive adversaries and realistic data distributions, can be bypassed, highlighting the need for a more robust solution.

In response, we propose Metric-Cascades (MESAS), a server-side defense mechanism for FL. MESAS utilizes multiple metrics extracted from locally trained models, ensuring resilience against strong adaptive adversaries. It employs statistical tests without fixed value thresholds, thus enabling versatile application. MESAS iteratively identifies and removes poisoned models within a single FL round. Notably, we are the first to assess defenses under inter-client non-IID data conditions, demonstrating MESAS's superior performance in real-world scenarios while incurring a minimal average computational overhead of 24.37 seconds.

ACKNOWLEDGMENT

We thank the Private AI Collaborate Research Institute which is co-sponsored by Intel Labs (www.private-ai.org) for partially supporting this research.

REFERENCES

[1] Abien Fred Agarap. 2018. Deep Learning using Rectified Linear Units (ReLU). *arXiv preprint arXiv:1803.08375* (2018).
[2] Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. 2020. The k-means Algorithm: A Comprehensive Survey and Performance Evaluation. *Electronics* (2020).
[3] Sebastien Andreina, Giorgia Azzurra Marson, Helen Möllering, and Ghassan Karame. 2021. BaFFLe: Backdoor Detection via Feedback-based Federated Learning. *ICDCS* (2021).
[4] Eugene Bagdasaryan and Vitaly Shmatikov. 2021. Blind Backdoors in Deep Learning Models. *USENIX Security* (2021).
[5] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. 2020. How To Backdoor Federated Learning. *AISTATS* (2020).
[6] Shefali Bansal, Medha Singh, Madhulika Bhadauria, and Richa Adalakha. 2022. Federated Learning Approach towards Sentiment Analysis. *ICTACS* (2022).
[7] Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. 2019. Analyzing Federated Learning through an Adversarial Lens. *ICML* (2019).
[8] Battista Biggio, Blaine Nelson, and Pavel Laskov. 2012. Poisoning Attacks against Support Vector Machine. *ICML* (2012).
[9] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. 2017. Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent. *NIPS* (2017).
[10] Nicholas Boucher, Ilia Shumailov, Ross Anderson, and Nicolas Papernot. 2022. Bad characters: Imperceptible NLP attacks. *IEEE S&P* (2022).
[11] California State Legislature. 2018. California Consumer Privacy Act. https://leginfo.legislature.ca.gov/faces/billTextClient.xhtml?bill_id=201720180SB1121.
[12] Di Cao, Shan Chang, Zhijian Lin, Guohua Liu, and Donghong Sun. 2019. Understanding Distributed Poisoning Attack in Federated Learning. *ICPADS* (2019).
[13] Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. 2021. FLTrust: Byzantine-robust Federated Learning via Trust Bootstrapping. *NDSS* (2021).
[14] Yair Censor. 1977. Pareto optimality in multiobjective problems. *Applied Mathematics and Optimization* (1977).
[15] Fei Chen, Mi Luo, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. 2018. Federated Meta-Learning with Fast Convergence and Efficient Communication. *arXiv preprint arXiv:1802.07876* (2018).
[16] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. 2017. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning. *arXiv preprint arXiv:1712.05526* (2017).
[17] Xiaoyi Chen, Ahmed Salem, Dingfan Chen, Michael Backes, Shiqing Ma, Qingni Shen, Zhonghai Wu, and Yang Zhang. 2021. BadNL: Backdoor Attacks against NLP Models with Semantic-preserving Improvements. *ACSAC* (2021).
[18] Victor Costan and Srinivas Devadas. 2016. Intel SGX Explained. *IACR Cryptol. ePrint Arch.* (2016).
[19] Erfan Darzidehkalani, Mohammad Ghasemi-rad, and P.M.A. van Ooijen. 2022. Federated Learning in Medical Imaging: Part II: Methods, Challenges, and Considerations. *Journal of the American College of Radiology* (2022).
[20] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. *CVPR* (2009).
[21] Li Deng. 2012. The MNIST Database of Handwritten Digit Images for Machine Learning Research. *IEEE Signal Processing Magazine* (2012).
[22] Cynthia Dwork. 2008. Differential Privacy: A Survey of Results. *TAMC* (2008).
[23] Jean-Antoine Désidéri. 2012. Multiple-gradient descent algorithm (MGDA) for multiobjective optimization. *Comptes Rendus Mathematique* (2012).
[24] El Mahdi El Mhamdi, Rachid Guerraoui, and Sébastien Rouault. 2018. The Hidden Vulnerability of Distributed Learning in Byzantium. *PMLR* (2018).
[25] European Parliament and Council of the European Union. 2018. General Data Protection Regulation. https://eur-lex.europa.eu/eli/reg/2016/679/oj.
[26] Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. 2020. Local Model Poisoning Attacks to Byzantine-Robust Federated Learning. *USENIX Security* (2020).
[27] Jie Feng, Can Rong, Funing Sun, Diansheng Guo, and Yong Li. 2020. PMF: A Privacy-Preserving Human Mobility Prediction Framework via Federated Learning. *ACM IMWUT* (2020).
[28] Christopher Frederickson, Michael Moore, Glenn Dawson, and Robi Polikar. 2018. Attack Strength vs. Detectability Dilemma in Adversarial Machine Learning. *IJCNN* (2018).
[29] Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. 2020. The Limitations of Federated Learning in Sybil Settings. *RAID* (2020).
[30] Karan Ganju, Qi Wang, Wei Yang, Carl A Gunter, and Nikita Borisov. 2018. Property Inference Attacks on Fully Connected Neural Networks using Permutation Invariant Representations. *CCS* (2018).
[31] Yansong Gao, Bao Gia Doan, Zhi Zhang, Siqi Ma, Jiliang Zhang, Anmin Fu, Surya Nepal, and Hyoungshick Kim. 2020. Backdoor Attacks and Countermeasures on Deep Learning: A Comprehensive Review. *arXiv preprint arXiv:2007.10760* (2020).
[32] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. 2017. BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. *arXiv preprint arXiv:1708.06733* (2017).
[33] Gozde N Gunesli, Mohsin Bilal, Shan E Ahmed Raza, and Nasir M Rajpoot. 2021. FedDropoutAvg: Generalizable federated learning for histopathology image classification. *arXiv preprint arXiv:2111.13230* (2021).
[34] Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. 2018. Federated Learning for Mobile Keyboard Prediction. *arXiv preprint arXiv:1811.03604* (2018).
[35] Hanieh Hashemi, Yongqin Wang, Chuan Guo, and Murali Annavaram. 2021. Byzantine-Robust and Privacy-Preserving Framework for FedML. *ICLR Workshops* (2021).
[36] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. 2019. LOGAN: Membership inference attacks against generative models. *PETS* (2019).
[37] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. *CVPR* (2016).
[38] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. *arXiv preprint arXiv:1602.07360* (2016).
[39] Jakub Konečnỳ, H Brendan McMahan, Daniel Ramage, and Peter Richtárik. 2016. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527* (2016).
[40] Torsten Krauß and Alexandra Dmitrienko. 2023. Avoid Adversarial Adaption in Federated Learning by Multi-Metric Investigations. *arXiv preprint arXiv:2306.03600* (2023).
[41] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning Multiple Layers of Features from Tiny Images. *Citeseer* (2009).
[42] Kavita Kumari, Phillip Rieger, Hossein Fereidooni, Murtuza Jadliwala, and Ahmad-Reza Sadeghi. 2023. BayBFed: Bayesian Backdoor Defense for Federated Learning. *IEEE S&P* (2023).
[43] Li Li, Yuxi Fan, Mike Tse, and Kuo-Yi Lin. 2020. A review of applications in federated learning. *Computers & Industrial Engineering* (2020).
[44] Liping Li, Wei Xu, Tianyi Chen, Georgios B Giannakis, and Qing Ling. 2019. RSA: Byzantine-robust stochastic aggregation methods for distributed learning from heterogeneous datasets. *AAAI* (2019).
[45] Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. 2022. Backdoor Learning: A Survey. *IEEE Transactions on Neural Networks and Learning Systems* (2022).
[46] Yijing Li, Xiaofeng Tao, Xuefei Zhang, Junjie Liu, and Jin Xu. 2022. Privacy-Preserved Federated Learning for Autonomous Driving. *IEEE T-ITS* (2022).
[47] Tjen-Sien Lim and Wei-Yin Loh. 1996. A comparison of tests of equality of variances. *Computational Statistics & Data Analysis* (1996).
[48] Chih-Ting Liu, Chien-Yi Wang, Shao-Yi Chien, and Shang-Hong Lai. 2022. FedFR: Joint Optimization Federated Framework for Generic and Personalized Face Recognition. *AAAI* (2022).
[49] Pengrui Liu, Xiangrui Xu, and Wei Wang. 2022. Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives. *Cybersecurity* (2022).
[50] Yang Liu, Anbu Huang, Yun Luo, He Huang, Youzhi Liu, Yuanyuan Chen, Lican Feng, Tianjian Chen, Han Yu, and Qiang Yang. 2020. FedVision: An Online Visual Object Detection Platform Powered by Federated Learning. *AAAI* (2020).
[51] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and X. Zhang. 2018. Trojaning Attack on Neural Networks. *NDSS* (2018).
[52] Edward H Livingston. 2004. Who was student and why do we care so much about his t-test? *Journal of Surgical Research* (2004).
[53] Frank J Massey Jr. 1951. The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American statistical Association* (1951).
[54] Leland McInnes, John Healy, and Steve Astels. 2017. HDBScan: Hierarchical density based clustering. *The Journal of Open Source Software* (2017).
[55] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. *AISTATS* (2017).
[56] Brendan McMahan and Daniel Ramage. 2017. Federated learning: Collaborative Machine Learning without Centralized Training Data. *Google AI* (2017).
[57] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning Differentially Private Language Models Without Losing Accuracy. *ICLR* (2018).
[58] Thomas Minka. 2000. Estimating a Dirichlet distribution.
[59] Fan Mo, Hamed Haddadi, Kleomenis Katevas, Eduard Marin, Diego Perino, and Nicolas Kourtellis. 2021. PPFL: Privacy-preserving Federated Learning with Trusted Execution Environments. *MobiSys* (2021).
[60] Luis Muñoz-González, Kenneth T Co, and Emil C Lupu. 2019. Byzantine-Robust Federated Machine Learning through Adaptive Model Averaging. *arXiv preprint arXiv:1909.05125* (2019).

[61] Mohammad Naseri, Jamie Hayes, and Emiliano De Cristofaro. 2022. Local and Central Differential Privacy for Robustness and Privacy in Federated Learning. *NDSS* (2022).

[62] Blaine Nelson, Marco Barreno, Fuching Jack Chi, Anthony D Joseph, Benjamin IP Rubinstein, Udam Saini, Charles Sutton, J Doug Tygar, and Kai Xia. 2008. Exploiting Machine Learning to Subvert Your Spam Filter. *LEET* (2008).

[63] Anh Nguyen, Tuong Do, Minh Tran, Binh X. Nguyen, Chien Duong, Tu Phan, Erman Tjiputra, and Quang D. Tran. 2022. Deep Federated Learning for Autonomous Driving. *IEEE IV* (2022).

[64] Dinh C. Nguyen, Quoc-Viet Pham, Pubudu N. Pathirana, Ming Ding, Aruna Seneviratne, Zihuai Lin, Octavia Dobre, and Won-Joo Hwang. 2022. Federated Learning for Smart Healthcare: A Survey. *ACM Comput. Surv.* (2022).

[65] Thien Duc Nguyen, Phillip Rieger, Huili Chen, Hossein Yalame, Helen Möllering, Hossein Fereidooni, Samuel Marchal, Markus Miettinen, Azalia Mirhoseini, Farinaz Koushanfar, Ahmad-Reza Sadeghi, Thomas Schneider, and Shaza Zeitouni. 2022. FLAME: Taming Backdoors in Federated Learning. *USENIX Security* (2022).

[66] Thien Duc Nguyen, Phillip Rieger, Markus Miettinen, and Ahmad-Reza Sadeghi. 2020. Poisoning Attacks on Federated Learning-Based IoT Intrusion Detection System. *NDSS DISS* (2020).

[67] Frank Nielsen. 2016. Hierarchical Clustering. *Introduction to HPC with MPI for Data Science* (2016).

[68] NVIDIA, Péter Vingelmann, and Frank H.P. Fitzek. 2020. CUDA, release: 10.2.89. https://developer.nvidia.com/cuda-toolkit

[69] Wojciech Ozga, Do Le Quoc, and Christof Fetzer. 2021. Perun: Confidential Multi-stakeholder Machine Learning Framework with Hardware Acceleration Support. *DBSec* (2021).

[70] Xudong Pan, Mi Zhang, Beina Sheng, Jiaming Zhu, and Min Yang. 2022. Hidden Trigger Backdoor Attack on NLP Models via Linguistic Style Manipulation. *USENIX Security* (2022).

[71] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS* (2019).

[72] Friedrich Pukelsheim. 1994. The Three Sigma Rule. *The American Statistician* (1994).

[73] Apostolos Pyrgelis, Carmela Troncoso, and Emiliano De Cristofaro. 2018. Knock Knock, Who's There? Membership Inference on Aggregate Location Data. *NDSS* (2018).

[74] Do Le Quoc, Franz Gregor, Sergei Arnautov, Roland Kunkel, Pramod Bhatotia, and Christof Fetzer. 2020. SecureTF: A Secure TensorFlow Framework. *Middleware* (2020).

[75] Swaroop Ramaswamy, Rajiv Mathews, Kanishka Rao, and Françoise Beaufays. 2019. Federated Learning for Emoji Prediction in a Mobile Keyboard. *arXiv preprint arXiv:1906.04329* (2019).

[76] Phillip Rieger, Thien Duc Nguyen, Markus Miettinen, and Ahmad-Reza Sadeghi. 2022. DeepSight: Mitigating Backdoor Attacks in Federated Learning Through Deep Model Inspection. *NDSS* (2022).

[77] Holger R Roth, Ken Chang, Praveer Singh, Nir Neumark, Wenqi Li, Vikash Gupta, Sharut Gupta, Liangqiong Qu, Alvin Ihsani, Bernardo C Bizzo, et al. 2020. Federated learning for breast density classification: A real-world implementation. *MICCAI* (2020).

[78] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. 2020. Hidden Trigger Backdoor Attacks. *AAAI* (2020).

[79] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. 2020. Updates-Leak: Data Set Inference and Reconstruction Attacks in Online Learning. *USENIX Security* (2020).

[80] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108* (2020).

[81] Ozan Sener and Vladlen Koltun. 2018. Multi-Task Learning as Multi-Objective Optimization. *NeurIPS* (2018).

[82] Micah Sheller, Anthony Reina, Brandon Edwards, Jason Martin, and Spyridon Bakas. 2018. Federated Learning for Medical Imaging. *Intel AI* (2018).

[83] Micah Sheller, Anthony Reina, Brandon Edwards, Jason Martin, and Spyridon Bakas. 2018. Multi-Institutional Deep Learning Modeling Without Sharing Patient Data: A Feasibility Study on Brain Tumor Segmentation. *Brain Lesion Workshop* (2018).

[84] Shiqi Shen, Shruti Tople, and Prateek Saxena. 2016. Auror: Defending Against Poisoning Attacks in Collaborative Deep Learning Systems. *ACSAC* (2016).

[85] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership Inference Attacks Against Machine Learning Models. *IEEE S&P* (2017).

[86] Santiago Silva, Boris A. Gutman, Eduardo Romero, Paul M. Thompson, Andre Altmann, and Marco Lorenzi. 2019. Federated Learning in Distributed Medical Databases: Meta-Analysis of Large-Scale Subcortical Brain Data. *IEEE ISBI* (2019).

[87] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. *EMNLP* (2013).

[88] Konstantin Sozinov, Vladimir Vlassov, and Sarunas Girdzijauskas. 2018. Human Activity Recognition Using Federated Learning. *IEEE BdCloud* (2018).

[89] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. 2012. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks* (2012).

[90] Octavian Suciu, Radu Marginean, Yigitcan Kaya, Hal Daume III, and Tudor Dumitras. 2018. When Does Machine Learning FAIL? Generalized Transferability for Evasion and Poisoning Attacks. *USENIX Security* (2018).

[91] Gan Sun, Yang Cong, Jiahua Dong, Qiang Wang, Lingjuan Lyu, and Ji Liu. 2022. Data Poisoning Attacks on Federated Machine Learning. *IEEE IoT-J* (2022).

[92] Ziteng Sun, Peter Kairouz, Ananda Theertha Suresh, and H. Brendan McMahan. 2019. Can You Really Backdoor Federated Learning? *arXiv preprint arXiv:1911.07963* (2019).

[93] The Linux Foundation. 2022. PyTorch. https://pytorch.org.

[94] Zhiyi Tian, Lei Cui, Jie Liang, and Shui Yu. 2022. A Comprehensive Survey on Poisoning Attacks and Countermeasures in Machine Learning. *Comput. Surveys* (2022).

[95] Florian Tramer and Dan Boneh. 2019. Slalom: Fast, Verifiable and Private Execution of Neural Networks in Trusted Hardware. *ICLR* (2019).

[96] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. 2019. Label-Consistent Backdoor Attacks. *arXiv preprint arXiv:1912.02771* (2019).

[97] U.S. Congress. 1996. Health Insurance Portability and Accountability Act. https://www.govinfo.gov/content/pkg/PLAW-104publ191/pdf/PLAW-104publ191.pdf.

[98] Guido Van Rossum and Fred L Drake Jr. 1995. *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam.

[99] Stavros Volos, Kapil Vaswani, and Rodrigo Bruno. 2018. Graviton: Trusted Execution Environments on GPUs. *OSDI* (2018).

[100] Hongyi Wang, Kartik Sreenivasan, Shashank Rajput, Harit Vishwakarma, Saurabh Agarwal, Jy-yong Sohn, Kangwook Lee, and Dimitris Papailiopoulos. 2020. Attack of the Tails: Yes, You Really Can Backdoor Federated Learning. *NIPS* (2020).

[101] Lixu Wang, Shichao Xu, Xiao Wang, and Qi Zhu. 2019. Eavesdrop the Composition Proportion of Training Labels in Federated Learning. *arXiv preprint arXiv:1910.06044* (2019).

[102] Zhibo Wang, Mengkai Song, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. 2019. Beyond Inferring Class Representatives: User-Level Privacy Leakage From Federated Learning. *INFOCOM* (2019).

[103] Zhaoxian Wu, Qing Ling, Tianyi Chen, and Georgios B. Giannakis. 2020. Federated Variance-Reduced Stochastic Gradient Descent With Robustness to Byzantine Attacks. *IEEE Transactions on Signal Processing* (2020).

[104] Geming Xia, Jian Chen, Chaodong Yu, and Jun Ma. 2023. Poisoning Attacks in Federated Learning: A Survey. *IEEE Access* (2023).

[105] Chulin Xie, Keli Huang, Pin-Yu Chen, and Bo Li. 2020. DBA: Distributed Backdoor Attacks against Federated Learning. *ICLR* (2020).

[106] Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. 2020. Fall of Empires: Breaking Byzantine-tolerant SGD by Inner Product Manipulation. *UAI* (2020).

[107] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated Machine Learning: Concept and Applications. *ACM Transactions on Intelligent Systems and Technology* (2019).

[108] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated Machine Learning: Concept and Applications. *TIST* (2019).

[109] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. 2018. Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates. *ICML* (2018).

[110] Hongyi Zhang, Jan Bosch, and Helena Holmström Olsson. 2021. End-to-End Federated Learning for Autonomous Driving Vehicles. *IJCNN* (2021).

[111] Zaixi Zhang, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. 2022. FLDetector: Defending Federated Learning Against Model Poisoning Attacks via Detecting Malicious Clients. *KDD22* (2022).

[112] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. 2020. iDLG: Improved Deep Leakage from Gradients. *arXiv preprint arXiv:2001.02610* (2020).

[113] Lingchen Zhao, Shengshan Hu, Qian Wang, Jianlin Jiang, Chao Shen, Xiangyang Luo, and Pengfei Hu. 2021. Shielding Collaborative Learning: Mitigating Poisoning Attacks Through Client-Side Detection. *PRDC* (2021).

[114] Hangyu Zhu, Jinjin Xu, Shiqing Liu, and Yaochu Jin. 2021. Federated Learning on Non-IID Data: A Survey. *Neurocomput.* (2021).

[115] Jianping Zhu, Rui Hou, XiaoFeng Wang, Wenhao Wang, Jiangfeng Cao, Lutan Zhao, Fengkai Yuan, Peinan Li, Zhongpu Wang, Boyan Zhao, Lixin Zhang, and Dan Meng. 2019. Enabling Privacy-Preserving, Compute- and Data-Intensive Computing using Heterogeneous Trusted Execution Environment. *arXiv preprint arXiv:1904.04782* (2019).