Avoid Adversarial Adaption in Federated Learning by Multi-Metric Investigations

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

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

ABSTRACT

Federated Learning (FL) trains machine learning models on data distributed across multiple devices, avoiding data transfer to a central location. This improves privacy, reduces communication costs, and enhances model performance. However, FL is prone to poisoning attacks, which can be untargeted aiming to reduce the model performance, or targeted, so-called backdoors, which add adversarial behavior that can be triggered with appropriately crafted inputs. Striving for stealthiness, backdoor attacks are harder to deal with.

Mitigation techniques against poisoning attacks rely on monitoring certain metrics and filtering malicious model updates. However, previous works didn't consider real-world adversaries and data distributions. To support our statement, we define a new notion of *strong adaptive adversaries* that can simultaneously adapt to multiple objectives and demonstrate through extensive tests, that existing defense methods can be circumvented in this adversary model. We also demonstrate, that existing defenses have limited effectiveness when no assumptions are made about underlying data distributions.

To address realistic scenarios and adversary models, we propose *Metric-Cascades (MESAS)* a new defense that leverages multiple detection metrics simultaneously for the filtering of poisoned model updates. This approach forces adaptive attackers into a heavy multi-objective optimization problem, and our evaluation with nine backdoors and three datasets shows that even our strong adaptive attacker cannot evade MESAS's detection. We show that MESAS outperforms existing defenses in distinguishing backdoors from distortions originating from different data distributions *within* and *across* the clients. Overall, MESAS is the first defense that is robust against strong adaptive adversaries and is effective in real-world data scenarios while introducing a low overhead of 24.37s on average.

1 INTRODUCTION

Federated Learning (FL) enables the collaborative training of a Deep Neural Network (DNN) among multiple clients [56]. 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 [1–3], 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 [49], hospitals are training models collaboratively [22, 23, 36, 65, 78, 79, 84, 85, 88], and in Natural Language Processing (NLP) domain it is used for text prediction [18, 37, 57, 76, 109], sentiment analysis [10], and personalization [17]. Moreover, FL can be applied for human mobility prediction [31], visual object detection [51], and human activity recognition [90]. We refer for more examples to [44].

In federations, a subset of clients can be controlled by an adversary who submits poisoned updates to the server. These attacks can be untargeted [30, 45, 103, 106], with the goal to reduce the prediction performance of the model. Alternatively, targeted poisoning attacks, also called backdoor attacks [8, 9, 11, 14, 19, 20, 34, 35, 46, 63, 67, 71, 80, 92, 97, 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 [47, 64, 111].

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 [7, 9, 62, 94], (ii) *Robust Aggregation (RA)* methods enhance robustness of aggregation algorithms against backdoors [56, 110], and (iii) *Detection and Filtering (DF)* approaches detect the poisoned models and filter them out before the aggregation step [13, 32, 61, 66, 77, 86, 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 latter 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 a 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 is 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. To facilitate reproducibility and to advance research, we will open source MESAS's code upon acceptance.

Outline. In the remaining part of the paper we first provide necessary foundational knowledge in Sect. 2. Afterward, in Sect. 3, we formally define the considered scenario, and describe the addressed challenges in detail. The design and the functionality of MESAS is presented in Sect. 4, and the experimental results are reported in Sect. 5. Sect. 6 discusses security aspects, limitations, and future work. Sect. 7 discusses related works. Finally, we summarize the paper in Sect. 8.

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 [42, 56, 108] framework, multiple clients $C_k \in \{C_1, \ldots C_{\mathcal{N}}\}$ collaborate, orchestrated by a central server $\mathcal{S}$, to jointly improve a Deep Neuronal Network (DNN). In particular, each client C_k trains a local DNN model on a local dataset and uploads the result to $\mathcal{S}$ 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 $\mathcal{S}$, reducing the costs for infrastructure.

FL is an iterative process, where the central server $\mathcal{S}$ selects a subset n of the $\mathcal{N}$ available clients $C_i \in \{C_1, \ldots C_n\}$ for each round r and distributes an (initially untrained) global model G^r to them. 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 that includes hyper-parameters, such as learning rate (LR), epochs, etc. After training, the client C_i submits the model updates $\mathcal{U}_i^r = L_i^{r+1}$- G^r to the server $\mathcal{S}$, who aggregates them into a new global model G^{r+1}. There are multiple aggregation methods [13, 29, 61, 110] available for this step, with Federated Averaging (FedAVG) [56] being the most commonly used. FedAVG calculates the weighted average of all the updates using the global learning rate δ as formalized in Eq. 1. After aggregation, the new round $r+1$ is initialized by $\mathcal{S}$.

$$G^{r+1} = G^r + \delta(\frac{1}{n}\sum_{i=0}^{n-1}\mathcal{U}_i^r) \tag{1}$$

2.2 Poisoning Attacks in Federated Learning

In the following, we distinguish between *untargeted* and *targeted* poisoning attacks [95, 104] and discuss the two methods that are applied to launch those attacks, namely *data* and *model poisoning*.

Untargeted poisoning aims to reduce the prediction performance of a model on a benign test dataset $\mathcal{D}_{test}$ with correctly labeled predictions y, which we refer to as model accuracy (MA) of global model G^{r+1} (cf. Eq. 2). To name an example, the adversary can assign an incorrect label for each sample in the dataset, thus misdirecting the model during training.

$$MA = \frac{|\{(d, y) \in \mathcal{D}_{test}\ :\ f(d, G^{r+1}) == y\}|}{|\mathcal{D}_{test}|} \tag{2}$$

Targeted attacks, also referred to as *backdoor attacks*, strive to force a DNN to produce attacker-chosen mispredictions when provided 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 in the upper left corner of an image [9, 35, 52]. In more details, an adversary A, who controls one or more clients C_i within a federation ($A_j \in \{C_1, \ldots C_n\}$), tries to submit poisoned local models to the server, so that the aggregated model G^{r+1} outputs a predefined target prediction P when fed with an input sample d^T containing the trigger T, with P and T being chosen by the adversary. An effective attack has high prediction performance, called backdoor accuracy (BA), on triggered input tested with a dataset $\mathcal{D}^T_{test}$ that contains only triggered samples, as formalized in Eq. 3. We attest a successful attack for a BA bigger than 60% in the global model.

$$BA = \frac{|\{(d^T, P) \in \mathcal{D}^T_{test} \ : \ f(d, G^{r+1}) == P\}|}{|\mathcal{D}^T_{test}|} \tag{3}$$

Data poisoning [93] 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 behaviour. 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 [9, 28]. As shown in Eq. 4, the objectives are weighted by α, allowing the adversary to prioritize between performance (MA/BA) and adaption intensity and consequently stealthiness.

$$Loss = \alpha \cdot Loss^{MA/BA} + (1 - \alpha) \cdot Loss^{Adaption} \tag{4}$$

A *classical adaptive adversary* creates a loss function for the deployed defense and applies Eq. 4 to bypass the defensive measure[1]. 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* [9].

The goal of a defense against poisoning attacks is to create a situation, where $Loss^{MA/BA}$ and $Loss^{Adaption}$ cannot be optimized simultaneously, so that A is faced with a trade-off between an effective attack and adapting to the defense, which is called *adversarial dilemma* [77].

3 PROBLEMS AND DEFINITIONS

In this section, we define our threat model in Sect. 3.1 and introduce the concept of a strong adaptive adversary in Sect. 3.2. The concluding Sect. 3.3 is devoted to the problem of arbitrary data distributions.

3.1 Threat Model

We analyze a classical FL system as depicted in Sect. 2.1. The aggregation server $\mathcal{S}$ applies FedAVG with a fixed global LR of $\delta = 1$. We consider an adversary A, 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 [7, 13, 61, 66, 77, 86], 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 $\mathcal{S}$ weights all model updates equally with $\frac{1}{n}$. In contrast to previous works, we do not make any assumption about the data distributions [114] within or across clients' dataset.

3.2 Strong Adaptive Adversary

Problem. 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[2]. As a state-of-the-art technique for this challenge, Eq. 4 is used to consider multiple objectives and simultaneously allowing the adversary to weight between better prediction performance and higher adaption level (MA and BA) via α. This adaption method from Eq. 4 exhibits effectiveness in two scenarios: 1) When dealing with a single adaptation loss, as α can appropriately balance the significance of the main task and adaptation. 2) When faced with multiple adaptation losses, provided that these losses are combined, e.g., summed into a single value. However, this approach functions optimally only when all the losses are on the same scale, as individual tuning of the various components of adaptation losses is not feasible. For instance, consider the scenario where $Loss^{MA/BA} = 10$ and the adaptation loss $Loss^{Adaption}$ comprises two individual losses: $Loss_1 = 1$ and $Loss_2 = 0.0001$. In this case, the second adaptation loss will have a negligible impact on the model's parameters since its value is already close to zero. Consequently, the learning algorithm will prioritize minimizing the other losses instead. As a result, the underlying metric will not be appropriately adapted. However, to effectively bypass a defense that relies on the metric represented by $Loss_2 = 0$, a value as small as 0.0000001 may be necessary.

Definition of a strong adaptive adversary. We introduce a robust adaptive adversary capable of simultaneously adapting to multiple metrics, regardless of their value scales. To achieve this, the adversary initially scales all losses to the maximum loss value (as indicated by the λ values in Eq. 5). This ensures that all adaptation objectives and the main task are treated with equal importance.

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

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

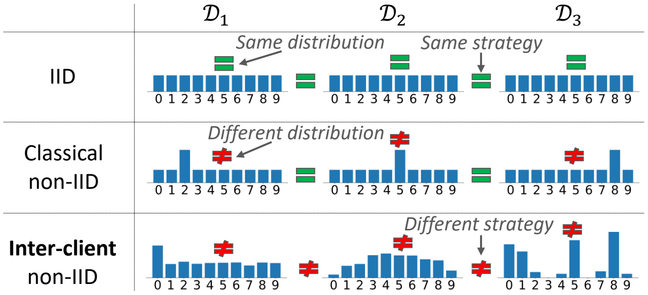

Figure 1: Comparison of various data distributions: IID, classical (intra-client) non-IID, and inter-client non-IID strategy for three client datasets $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$ with 10 label classes.

Subsequently, the adversary retains the ability to adjust the level of adaptation using the parameter α.

$$Loss = \alpha \cdot Loss^{MA/BA} + (1-\alpha) \cdot (\lambda_1 \cdot Loss_1 + \lambda_2 \cdot Loss_2 + \cdots) \quad (5)$$

Further, the adversary can simultaneously exclude specific model parameters, e.g., whole layers 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.

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.

3.3 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 cross 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.

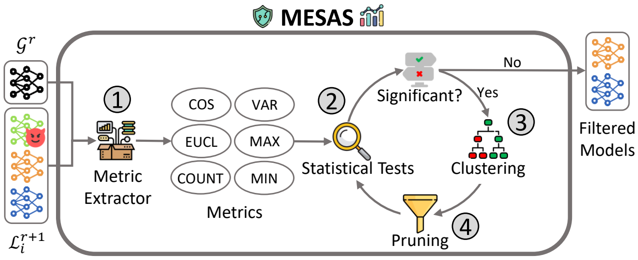

Figure 2: Overview of MESAS.

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 $\mathcal{D}_3$.

We evaluate FL defenses in inter-client non-IID scenarios and report the results in Sect. 5.3.

4 MESAS

In this section, we present our new defense against poisoning attacks, Metric-Cascades (MESAS). We start with 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, 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.

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 L_i^{r+1} and the global model G^r. 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[3]. 2) Thereafter, those metrics are analyzed individually in an iterative process. 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

[3] Naïve implemented backdoors are only embedded within the last few DNN layers.

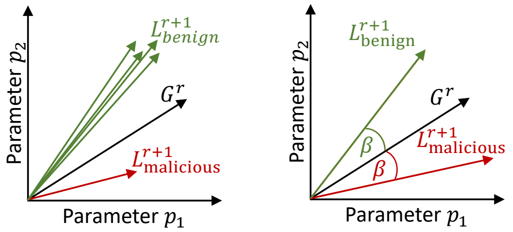

Figure 3: Simplified visualization of FL models with two parameters. The left graphic shows that benign and malicious models differ in one or multiple dimensions. On the right, we depict that benign and malicious models can have the same COS metric due to the same angel to the global model.

models are excluded in a pruning step and the analysis starts over on the remaining models until no statistical test reports significant 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 [40]. 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 [13, 32, 61, 66, 77, 110], 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 the COUNT metric, which counts how many parameter values are increased from the respective parameter of the global model G^r during training.

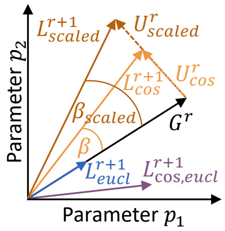

Figure 4: Visualization of locally trained models L_i^{r+1} deviating from the global model G^r in COS and EUCL. The figure also depicts how the angle β changes after scaling the update, thus provoking a change in the COS metric of MESAS.

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[4]. 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 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[5]. VAR combined with MAX and MIN provide a reliable metric for the functionality level and allow testing for poisoned models.

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

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

[5] 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.

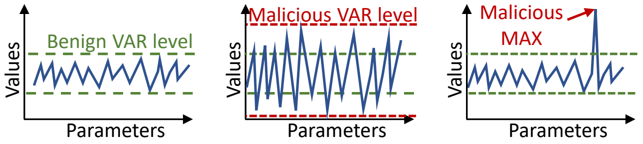

Figure 5: Simplified visualization of FL models with multiple parameters highlighting the functionality level based on the parameter value variance. The left shows a benign situation and the middle a poisoned model can have a bigger (or smaller) level. The figure on the right depicts,that the variance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

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 [53] (ST-T) is conducted to check for equal means. Since two distributions can have the same mean but different variances, a Levene's test [48] (ST-V) is appended. Finally, a Kolmogorow-Smirnow-Test [54] (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 [73] (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 metric 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[6], 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 [68] 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 Sect. A.

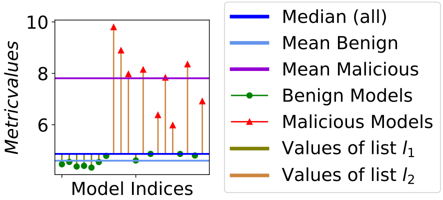

Figure 6: Visualization of a statistical test setup with significant p-value in ST-T indicating a varying mean between l_1 and l_2.

5 EVALUATION

In this section, we conduct a rigorous analysis of MESAS and explore 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 simulate the FL system on one server and implement the code in PyTorch [4, 72], which is a well-known machine learning library for Python [98]. The individual client and server code is executed sequentially on the server running with an AMD EPYC 7413 24-Core Processor (64-bit architecture) with 96 processing units and 128GB main memory. As accelerator, a NVIDIA A16 GPU with 4 virtual GPUs each having 16GB GDDR6 memory is accessible via CUDA [69] from PyTorch.

Datasets and Models. To be comparable to other FL defenses, we chose similar settings to related works and focus mainly on image classification with CIFAR-10 [43], GTSRB [91], and MNIST [25]. For model architectures, we use ResNet-18 [40], SqueezeNet [41], and a CNN. Additionally, we investigate into the text domain by training a DistilBERT [82] transformer model on SST-2 [89] sentiment analysis dataset.

Default Scenario. We train the CIFAR-10 [43] image classification task (ten classes) on a ResNet-18 [40] model with LR 0.01 (SGD optimizer, momentum 0.9, decay 0.005). The federation consists of $N = 20$ 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. 3.2 and implements a pixel trigger backdoor [35] (cf. App. B.1), which adds pixel pattern, a sticker, or similar as a trigger to the sample [9, 35, 52][7]. 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 layer being untrained since both needed to be changed according to our dataset.[8] The batch size is 64 and the models are trained for ten epochs.

[6] 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.

[7] More details and an example can be found in App. B.1.

[8] The pre-trained models from PyTorch are trained on ImageNet [24], thus have other input dimensions and 1000 instead of ten classes in the last layer.

Defenses. We compare the following nine approaches, with MESAS regarding effectiveness and runtime, hence examine DF, RA, and IR methods: Naïve *clustering via HDBSCAN [55], FoolsGold [32], Krum [13], M-Krum [13], Flame [66], Clipping&Noising* [58], *Clipping* [58], *T-Mean* [110], *T-Median* [110], and *Auror* [86]. 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 in the paper, we report the most interesting results and numbers that highlight our outcomes in the following sections and list detailed experimental results in App. F.

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[9], or by fixation, constraining and scaling (cf. Sect. 2.2). We explore the effectiveness of these strategies and list results in App. F. Here, we show that MESAS is more effective than other defenses even without applying additional adaptions when comparing them under the default scenario: As can be seen in the line 16 of Tab. 1, MESAS effectively removes the backdoor by reducing BA to 1.85%, while most other defenses are less potent. Only FoolsGold [32] is as effective as MESAS in the default scenario, 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[10]. Further, constraining with Eq. 4 or Eq. 5 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 adaption mechanisms 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. 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 App. F, which we couldn't include in the main section of the paper due to space limitations.

Table 1: MAs and BAs in the default scenario in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	57.84	85.13
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	64.92	83.00
6:	FedAVG with all local models	63.81	42.94
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	65.06	74.62
8:	FoolsGold [32]	63.57	**1.85**
9:	Krum [13]	59.75	83.53
10:	M-Krum [13]	64.18	83.05
11:	Clip [58]	63.80	42.81
12:	Clip&Noise [58]	50.78	60.66
13:	Flame [66]	60.96	**79.17**
14:	T-Mean [110]	63.51	44.13
15:	T-Median [110]	51.22	44.60
16:	**MESAS**	63.57	**1.85**

Clustering. To demonstrate that naïve clustering methods could be bypassed, we use the HDBSCAN [55] algorithm as an example and cluster based on the cross-wise cosine distances between model updates. As can be seen 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 App. Tab. 8).

FoolsGold. The second defense, FoolsGold [32], is also based on cross-wise cosine distances between model updates. However, it analyzes only outputs of the last layer, which is more effective that naïve clustering and is capable of removing all poisoned models reaching a BA 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, 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 [13], 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). 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. 4 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. In contrast, MESAS accurately filters out the backdoor in similar circumstances, as adaption via constraint has significant effects on other metrics, like EUCL and MIN.

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

[10] 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 2: BA for targeted and ACC for untargeted poisoning attacks without adaptive adversary in percent.

		BA						ACC		
	Aggregation / Defenses	Pixel Trigger [35]	Clean-Label [97]	Semantic [9]	Edge Case [100]	Label Flip [12, 15]	Pervasive [19]	Random Flip App. B.7	Sign Flip App. B.8	Noising App. B.9
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 [32]	**1.85**	**1.85**	**0.00**	**2.55**	**0.20**	**0.10**	55.00	**100.00**	0.00
9:	Krum [13]	83.53	75.65	80.00	20.91	**1.30**	**0.42**	50.00	50.00	50.00
10:	M-Krum [13]	83.05	82.38	100.0	18.87	**0.40**	**3.50**	75.00	75.00	75.00
11:	Clip [58]	42.81	38.91	60.0	**6.63**	48.40	**3.17**	-	-	-
12:	Clip&Noise [58]	60.66	40.73	**0.00**	12.75	30.80	10.08	-	-	-
13:	Flame [66]	79.17	77.12	60.0	18.87	**2.40**	**5.52**	**100.00**	**100.00**	**100.00**
14:	T-Mean [110]	44.13	41.10	60.0	**7.14**	48.40	**2.53**	-	-	-
15:	T-Median [110]	44.60	25.66	**0.00**	**2.55**	**5.60**	**0.10**	-	-	-
16:	**MESAS**	**1.85**	**3.71**	**0.00**	**2.55**	**0.20**	**0.05**	**95.00**	**100.00**	**100.00**

Flame. As a more complex DF defense, we evaluate Flame [66], which combines clustering methods 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 and achieving 79.17% BA, as can be seen in line 13 in Tab. 1. Similar to naïve clustering, we could strengthen the BA by increasing the PDR and 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.

Differential Privacy. Besides DF methods, we evaluated two IR approaches: Clipping of the model updates based on the Euclidean distance and a combination with noising of the model parameters [58]. Clipping is ineffective, as our default scenario backdoor is not reflected in the Euclidean distance of the model updates. Thus, the attacker can achieve 60.66% BA (cf. line 12 of Tab. 1). When using adaption, the BA can be increased slightly to 61.86% by following the strategy of increasing PDR. In contrast, MESAS is effective under similar circumstances resulting in 1.95% BA.

Robust Aggregation. We evaluate T-Mean and T-Median [110], which are RA alternatives to FedAVG. Both result in weak backdoors with BA of 44.13% and 44.60%, respectively, 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. 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. 3.2 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 [35], clean-label backdoor [97], semantic backdoor [9], edge case backdoor [100], label flip backdoor [12, 15], and pervasive backdoor [19] as well as random label flipping (cf. App. B.7), sign flipping (cf. App. B.8), and model noising (cf. App. B.9) which are all explained in detail in App. B. We report the BAs that the poisoning attacks achieve against the nine defenses in Tab. 2 and the MAs in App. Tab. 12.

Pixel Trigger Backdoor This backdoor is already 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%.[11]

Semantic Backdoor. Without defense, this backdoor is effective with 60% BA. However, it is detectable within the last layers by FoolsGold [32] 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

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

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 [43] on ResNet-18 [40]. 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 16).

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 [19] 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. App. Tab. 12) 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 App. Tab. 12. Random label flipping (cf. App. B.7) is the first untargeted attack that we implemented. The MA is reduced to 57.03% without any defense and only M-Krum can score a higher MA of 64.15% compared to 62.88% of MESAS. However, M-Krum suffers a FNR of 45%, compared to 0.09% of MESAS. Flame stands out with 100% ACC, but can be circumvented by adaption. Second, we evaluated sign flipping (cf. App. B.8), 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 (cf. App. B.9), where MESAS also has an ACC of 100%.

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.

5.3 Defenses under Non-IID

Here, we evaluate the same nine defenses, as specified in Sect. 5.1, under different non-IID scenarios. First, we investigate 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. 1-class and 2-class non-IID, introduce a focus on one or two so-called *main labels* within the samples of a client's dataset. The remaining labels contain an equivalent amount of samples, while a factor $q \in [0, 1]$ defines the fraction between the number of samples within the main label class and the remaining classes[12]. Distribution non-IID assigns label frequencies for each dataset based on a distribution, e.g., Dirichlet [59] or normal distribution. We elaborate on non-IID simulation techniques in more detail in App. E.1. 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 PDR 0.3 combined with scaling reaches reasonable performance with 63.66% BA (line 6 in Tab. 3). MESAS is the only defense erasing the backdoor efficiently in this setting and reaching 1.40% BA with two FPs.

In the unscaled version (cf. App. Tab. 3) Krum and M-Krum [13] also erase the backdoor, however Krum reduces the MA immensely. However, after an adaption, we can circumvent those defenses reaching BAs of up to 90.44%, while still erasing the backdoor with MESAS. Hence, we can confidently say, that 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 which is described in detail in App. E.2. 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 after 50 benign rounds of FL training with 20 clients in the federation in Tab. 4. The underlying sample frequencies for each client of the scenario are listed in App. Tab. 5. 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 of Tab. 4. Among all defenses, MESAS is the only one capable of erasing the backdoor by decreasing the BA to 2.37%.

We repeated this experiment in FL round one[13] 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 do not appear to be capable of removing backdoors in inter-client non-IID scenarios. The detailed experiments are reported in App. Tab. 20, App. Tab. 21, App. Tab. 22, and App. Tab. 23.

5.4 Influence of Parameters on MESAS

To evaluate the influence of various parameters on the performance of MESAS, we first investigated training hyper-parameters and showed the independence from the 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 App. F.1.

Our experiments show, that the backdoor efficiency depends on the type and composition of the trigger, but also the PDR is

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

[13] Backdoors in early FL rounds are not persistent as already depicted in [9], but we still analyzed the situation.

Table 3: MA and BA in the default scenario with a PDR of 0.3, 1-class intra-client non-IID with $q = 0.5$, and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	G^r	62.99	1.93
2:	Average of benign local models	47.15	6.82
3:	Average of poisoned local models	43.74	91.32
4:	FedAVG with benign local models	65.92	1.40
5:	FedAVG with poisoned local models	59.12	95.50
6:	FedAVG with all local models	64.02	63.66
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	61.12	87.58
8:	FoolsGold [32]	56.80	47.04
9:	Krum [13]	**49.88**	5.27
10:	M-Krum [13]	62.39	13.11
11:	Clip [58]	63.92	62.28
12:	Clip&Noise [58]	56.28	71.99
13:	Flame [66]	56.59	50.34
14:	T-Mean [110]	63.15	67.01
15:	T-Median [110]	51.75	68.20
16:	**MESAS**	65.92	**1.40**

Table 4: MA and BA in the default scenario with inter-client non-IID based on our Random-Non-IID strategy in percent.

	Accuracies without defenses	MA	BA
1:	G^r	36.51	5.18
2:	Average of benign local models	33.15	10.42
3:	Average of poisoned local models	33.93	82.00
4:	FedAVG with benign local models	32.45	12.71
5:	FedAVG with poisoned local models	29.35	88.96
6:	FedAVG with all local models	38.72	77.37
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	20.85	85.32
8:	FoolsGold [32]	37.00	76.03
9:	Krum [13]	16.88	89.07
10:	M-Krum [13]	18.07	89.55
11:	Clip [58]	37.76	75.60
12:	Clip&Noise [58]	23.70	64.32
13:	Flame [66]	25.10	79.17
14:	T-Mean [110]	39.98	76.36
15:	T-Median [110]	17.04	52.75
16:	**MESAS**	37.52	**2.37**

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 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 multi-objective optimization problem. Concluding, we can claim, that MESAS is independent of the PDR selected by the adversary.

We conducted experiments with different pre-trained models. We used random initialized models as as well as pre-trained models from PyTorch [4, 72] 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 independent of the FL round. However, the detection performance in later round 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. App. E.2).

We exchanged the dataset of our default scenario to MNIST [25] and GTSRB [91] and could assert, that the experimental results and thus the performance of the defenses including MESAS does not vary across different 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 in App. Tab. 27 with one FP and one GTSRB experiment in App. Tab. 28 with 100% ACC.

Further, we conducted experiments to analyze the independence from model architectures. Therefore, we used a CNN with two convolutional layers concatenated with pooling layers and ReLu functions [5] followed by three fully connected layers and trained on MNIST [25]. Additionally, we tested SqueezeNet [41] trained on CIFAR-10 [43] and can report 100% TNs with just one FN in both cases (cf. App. Tab. 24 and App. Tab. 25). Hence, we can claim, that MESAS is independent from the architecture of the model.

Lastly, we conducted experiments within the text domain training a sentiment analysis task using the SST-2 [89] dataset on a DistilBERT [82] transformer model. We implemented a targeted poisoning attack, 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 different defenses to verify the real-world applicability of the approach. App. Tab. 26 lists the average runtimes of ten runs for our default scenario and shows that MESAS introduces an acceptable overhead of 24.37 seconds. Note, that FoolsGold [32] comes along with outstanding performance, since only one model layer is analyzed, but due to the same reason it can be easily circumvented by an attacker (cf. Sect. 5.2). Further, T-Median [110] replaces FedAVG with a simple algorithm, which result in similar runtime, but also reduces the MA. Auror [86] 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. Defenses leveraging client feedback [7, 113] cannot compete to server-side-only defenses, since additional communication overhead is introduced.

6 DISCUSSION

Below in Sect. 6.1, we first provide a summary on alternative adaption methods tested during this work. Thereafter, limitations and suggestions for future work are discussed in Sect. 6.2

6.1 Adversarial Adaption Methodologies

Besides the final method of our strong adaptive adversary (cf. Sect. 3.2) that we used to evaluate FL defenses in Sect. 5.2, multiple alternatives have been tested during this work. This section lists and discusses the approaches inferior to our final choice.

First, we just added all of the losses (λ's from Eq. 5 equal to one), which is similar to an classic adaptive adversary (cf. Sect. 2.2). As already explained in Sect. 5.2, losses with a drastically smaller scale than others have barley influence in the optimization, thus the related metric is not adapted. Second, we tried to scale all losses to $Loss^{MA/BA}$, which would be reasonable, it the MA would be the major concern of A. However, most defenses including MESAS do not check the MA since no test dataset is available in realistic scenarios, which makes scaling to the maximum the better choice for the adversary. Third, we tested, how often the λ's should be recalculated and found, that only one initial computation delivers the best results. This seems reasonable, since with this setting, already optimized metrics have a minimal loss value and thus barley influence in the optimization.

Additionally, motivated by Multi-Objective Optimization (MOO) research, we tried to find a pareto optimal [16] solution with the method of Sener *et al.* [83] based on the MGDA algroithm [26]. However, the method did not work and produced broken models regarding the accuracies. We belief, that the reason for this is, that Sener *et al.* consider a system comparable to Multi-Task Learning (MTL) where both, shared and task-specific parameters exist within the model. However, our MOO problem optimizes only shared parameters (the whole model).

Since our final adaption method is superior to classic adaption [9] from Eq. 4, 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

The major limitation of MESAS is, that the significance niveau for the statistical tests is relevant for a good TPR and TNR. Throughout our experiment, the values appeared to be just dependent on the data scenario. Nevertheless, it can be necessary in so far unseen tasks to adapt the values. Therefore, an automatic methodology 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 [66] and DeepSight [77]. 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 our experiments, the strong adaptive adversary from Sect. 3.2 cannot circumvent MESAS. Nevertheless, 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 [86] is a K-Means [6] 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.[14] 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 [32] weights the contribution of each local model, by analyzing the 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 into the same directions (so-called sybills). Additionally, it leverages updates from previous rounds r for optimal performance. Instead, MESAS prevents adaption by leveraging a cascade of metrics and analyzing each layer individually and is effective in IID and non-IID settings independent of the FL round.

Krum [13] utilizes the Euclidean distance between local models as its foundation. It aggregates the distances to neighboring models for each local model and selects the one with the most densely surrounded neighbors as the new global model. M-Krum [13] extends this approach by selecting multiple models simultaneously. However, both methods can be bypassed through metric adaptation and inherently experience a high FNR even without adversaries in the system. In contrast, MESAS ensures the overall integrity of the federation in benign scenarios and maintains a low FNR while remaining resistant to adaptation attempts.

AFA [61] is based on a plain analysis of the cosine distance between local models, which is adaptable with an additional loss function. MESAS hardens this possibility for A by leveraging a cascade of six metrics.

Naïve clustering approaches, e.g. based on HDBSCAN [55], always need to extract a metric like the cosine distance between models from the local models to reduce the dimensions. Hence, adaptive adversaries can always circumvent the defenses, which is hardened in MESAS. Further, clustering often rely on a majority assumption and creates two clusters, thus has either a hard value threshold or a high FNR in settings without attacks. In contrast, MESAS leverages statistical tests with probabilistic thresholds that adapt to the scenario. MESAS investigates in metrics that are challenging to adapt due to their fine-grained values. Such metrics lead

[14] In our experiments it took around 12 hours to calculate the indicative features.

to lower-scale adaption losses compared to conventional clustering metrics like the Cosine distance.

BaFFLe [7] 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 some 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 the adversary strives for 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 (see Sect. 7.2) can be conducted on the local models on the client side.

FLAME [66] is a combination of DF and IR. The approach hierarchically clusters local models by pairwise cosine distances and filters adversaries based on the majority assumption before differential privacy methods [27] are leveraged. Precisely, weight clipping (regarding the median Euclidean distance of the updates) is applied on 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 [94], 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 [13], Yin *et al.* [110] 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 reduc 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.* [9] and Sun *et al.* [94] already proposed update clipping and nosing techniques, but Naseri *et al.* [62] showed, that differential privacy methods not only naturally harm the MA [9], 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, thus does not influence benign scenarios naturally. Further, IR and RA methods can be easily combined with MESAS to get a even more bulletproof global model.

DeepSight [77] is a more complex strategy, similar to Flame [66], which combines filtering with differential privacy. The approach is based on two metrics. First, the cosine distance between models, which can be adapted by additional loss function, as shown in Sect. 5.2. Second, the output layer is used to extract two more values, which can be circumvented by fixation, as shown for FoolsGold [32] in our experiments. Therefore, DeepSight is not robust against strong adaptive adversaries and relies 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.

7.2 Privacy Preserving Federated Learning

FL in its original form [56] improved the privacy of collaborated DNN training compared to the data-centralized approach, since raw sensitive data do not leave the client side anymore. Nevertheless, membership inference [39, 50, 74, 87, 87], label inference [112], property inference [33], model extraction [50], and data reconstruction [81, 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 [60] 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 [21] enclaves is reduced, mainly due to page swaps based on limited memory. Additionally, such approaches based on secure code execution [70, 75, 96, 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.* [38] implemented Krum [13] 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

Adversarial adaption to defenses and complicated data scenarios are the two major challenges when it comes to Federated Learning (FL). To highlight the necessity to investigate these problems, we evaluate nine against a *strong adaptive adversary* that is able to poison the dataset, constraint the learning process, fixate model parameters, scale model updates, and select between nine different poisoning methods. Further, we analyze defense efficiencies without any assumption about the sample frequencies within the client datasets, which we call *inter-client non-IID*. We show, that by leveraging adaption methods, existing defenses can be circumvented and are also ineffective in realistic data scenarios.

Hence, we propose Metric-Cascades (MESAS), a filtering defense against poisoning attacks in FL running on the server side. It extracts multiple metrics from the locally trained models, making the defense robust against strong adaptive adversaries, and reliably detects poisoned contributions by leveraging statistical tests with no hard value threshold, which enables application independency. MESAS prunes poisoned models in an iterative process, allowing removal of different poisonings within one FL round.

We are the first to evaluate defenses under inter-client non-IID data scenarios and show that MESAS outperforms existing defenses in such real-world settings while only introducing a low computational overhead of 24.37 seconds on average.

REFERENCES

[1] 1996. Health Insurance Portability and Accountability Act. https://www.govinfo.gov/content/pkg/PLAW-104publ191/pdf/PLAW-104publ191.pdf.

[2] 2018. California Consumer Privacy Act. https://leginfo.legislature.ca.gov/faces/billTextClient.xhtml?bill_id=201720180SB1121.

[3] 2018. General Data Protection Regulation. https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[4] 2022. PyTorch. https://pytorch.org.

[5] Abien Fred Agarap. 2018. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375* (2018).

[6] Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. 2020. The k-means Algorithm: A Comprehensive Survey and Performance Evaluation. *Electronics* 9, 8 (2020). https://doi.org/10.3390/electronics9081295

[7] Sebastien Andreina, Giorgia Azzurra Marson, Helen Möllering, and Ghassan Karame. 2021. BaFFLe: Backdoor Detection via Feedback-based Federated Learning.

[8] Eugene Bagdasaryan and Vitaly Shmatikov. 2021. Blind backdoors in deep learning models. In *Usenix Security*.

[9] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. 2020. How To Backdoor Federated Learning. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 108)*, Silvia Chiappa and Roberto Calandra (Eds.). PMLR, 2938–2948. https://proceedings.mlr.press/v108/bagdasaryan20a.html

[10] Shefali Bansal, Medha Singh, Madhulika Bhadauria, and Richa Adalakha. 2022. Federated Learning Approach towards Sentiment Analysis. In *2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS)*. 717–724. https://doi.org/10.1109/ICTACS56270.2022.9987996

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

[12] Battista Biggio, Blaine Nelson, and Pavel Laskov. 2013. Poisoning Attacks against Support Vector Machines. arXiv:1206.6389 [cs.LG]

[13] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. 2017. Machine Learning with Adversaries: Byzantine Tolerant Gradient Descent.

[14] Nicholas Boucher, Ilia Shumailov, Ross Anderson, and Nicolas Papernot. 2022. Bad characters: Imperceptible nlp attacks. In *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1987–2004.

[15] Di Cao, Shan Chang, Zhijian Lin, Guohua Liu, and Donghong Sun. 2019. Understanding Distributed Poisoning Attack in Federated Learning. In *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)*. 233–239. https://doi.org/10.1109/ICPADS47876.2019.00042

[16] Yair Censor. 1977. Pareto optimality in multiobjective problems. *Applied Mathematics and Optimization* 4, 1 (01 Mar 1977), 41–59. https://doi.org/10.1007/BF01442131

[17] Fei Chen, Mi Luo, Zhenhua Dong, Zhenguo Li, and Xiuqiang He. 2018. Federated meta-learning with fast convergence and efficient communication. In *arXiv preprint arXiv:1802.07876*.

[18] Mingqing Chen, Rajiv Mathews, Tom Ouyang, and Françoise Beaufays. 2019. Federated Learning Of Out-Of-Vocabulary Words. arXiv:1903.10635 [cs.CL]

[19] 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).

[20] 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. In *Annual Computer Security Applications Conference*. 554–569.

[21] Victor Costan and Srinivas Devadas. 2016. Intel SGX Explained. In *IACR Cryptol. ePrint Arch.*, Vol. 2016. 86.

[22] Erfan Darzidehkalani, Mohammad Ghasemi-rad, and P.M.A. van Ooijen. 2022. Federated Learning in Medical Imaging: Part I: Toward Multicentral Health Care Ecosystems. *Journal of the American College of Radiology* 19, 8 (2022), 969–974. https://doi.org/10.1016/j.jacr.2022.03.015

[23] 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* 19, 8 (2022), 975–982. https://doi.org/10.1016/j.jacr.2022.03.016

[24] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

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

[26] Jean-Antoine Désidéri. 2009. *Multiple-gradient descent algorithm (MGDA)*. Ph. D. Dissertation. INRIA.

[27] Cynthia Dwork. 2008. Differential privacy: A survey of results. In *Theory and Applications of Models of Computation: 5th International Conference, TAMC 2008, Xi'an, China, April 25-29, 2008. Proceedings 5*. Springer, 1–19.

[28] Jean-Antoine Désidéri. 2012. Multiple-gradient descent algorithm (MGDA) for multiobjective optimization. *Comptes Rendus Mathematique* 350, 5 (2012), 313–318. https://doi.org/10.1016/j.crma.2012.03.014

[29] El Mahdi El Mhamdi, Rachid Guerraoui, and Sébastien Rouault. 2018. The Hidden Vulnerability of Distributed Learning in Byzantium. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, 3521–3530. https://proceedings.mlr.press/v80/mhamdi18a.html

[30] Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. 2020. Local Model Poisoning Attacks to Byzantine-Robust Federated Learning.

[31] Jie Feng, Can Rong, Funing Sun, Diansheng Guo, and Yong Li. 2020. PMF: A Privacy-Preserving Human Mobility Prediction Framework via Federated Learning. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 1, Article 10 (mar 2020), 21 pages. https://doi.org/10.1145/3381006

[32] Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. 2020. The limitations of federated learning in sybil settings.

[33] 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. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*. 619–633.

[34] 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).

[35] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. 2017. Badnets: Identifying vulnerabilities in the machine learning model supply chain. In *arXiv preprint arXiv:1708.06733*.

[36] 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).

[37] 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. In *arXiv preprint arXiv:1811.03604*.

[38] Hanieh Hashemi, Yongqin Wang, Chuan Guo, and Murali Annavaram. 2021. Byzantine-Robust and Privacy-Preserving Framework for FedML. In *ICLR Workshops*.

[39] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. 2019. LOGAN: Membership inference attacks against generative models. In *Privacy Enhancing Technologies*.

[40] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[41] 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:1602.07360 [cs.CV]

[42] 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).

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

[44] Li Li, Yuxi Fan, Mike Tse, and Kuo-Yi Lin. 2020. A review of applications in federated learning. *Computers & Industrial Engineering* 149 (2020), 106854. https://doi.org/10.1016/j.cie.2020.106854

[45] 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. In *AAAI Conference on Artificial Intelligence*.

[46] Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. 2022. Backdoor Learning: A Survey. *IEEE Transactions on Neural Networks and Learning Systems* (2022), 1–18. https://doi.org/10.1109/TNNLS.2022.3182979

[47] Yijing Li, Xiaofeng Tao, Xuefei Zhang, Junjie Liu, and Jin Xu. 2022. Privacy-Preserved Federated Learning for Autonomous Driving. *IEEE Transactions on Intelligent Transportation Systems* 23, 7 (2022), 8423–8434. https://doi.org/10.1109/TITS.2021.3081560

[48] Tjen-Sien Lim and Wei-Yin Loh. 1996. A comparison of tests of equality of variances. *Computational Statistics & Data Analysis* 22, 3 (1996), 287–301.

[49] 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. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 1656–1664.

[50] Pengrui Liu, Xiangrui Xu, and Wei Wang. 2022. Threats, attacks and defenses to federated learning: issues, taxonomy and perspectives. *Cybersecurity* 5, 1 (2022).

[51] 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. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 08 (Apr. 2020), 13172–13179. https://doi.org/10.1609/aaai.v34i08.7021

[52] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and X. Zhang. 2018. Trojaning Attack on Neural Networks. In *NDSS*.

[53] Edward H Livingston. 2004. Who was student and why do we care so much about his t-test? 1. *Journal of Surgical Research* 118, 1 (2004), 58–65.

[54] Frank J Massey Jr. 1951. The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American statistical Association* 46, 253 (1951), 68–78.

[55] Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software* 2, 11 (mar 2017). https://doi.org/10.21105/joss.00205

[56] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data.

[57] Brendan McMahan and Daniel Ramage. 2017. Federated learning: Collaborative Machine Learning without Centralized Training Data. Google AI.

[58] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. 2018. Learning Differentially Private Language Models Without Losing Accuracy.

[59] Thomas Minka. 2000. Estimating a Dirichlet distribution.

[60] Fan Mo, Hamed Haddadi, Kleomenis Katevas, Eduard Marin, Diego Perino, and Nicolas Kourtellis. 2021. PPFL: privacy-preserving federated learning with trusted execution environments. In *Annual International Conference on Mobile Systems, Applications, and Services*.

[61] 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).

[62] Mohammad Naseri, Jamie Hayes, and Emiliano De Cristofaro. 2022. Local and central differential privacy for robustness and privacy in federated learning.

[63] 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* 8, 1-9 (2008), 16–17.

[64] 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. In *2022 IEEE Intelligent Vehicles Symposium (IV)*. 1824–1830. https://doi.org/10.1109/IV51971.2022.9827020

[65] 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.* 55, 3, Article 60 (feb 2022), 37 pages. https://doi.org/10.1145/3501296

[66] 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.

[67] Thien Duc Nguyen, Phillip Rieger, Markus Miettinen, and Ahmad-Reza Sadeghi. 2020. Poisoning Attacks on Federated Learning-Based IoT Intrusion Detection System. In *Workshop on Decentralized IoT Systems and Security*.

[68] Frank Nielsen. 2016. *Hierarchical Clustering*. 195–211. https://doi.org/10.1007/978-3-319-21903-5_8

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

[70] Wojciech Ozga, Do Le Quoc, and Christof Fetzer. 2021. Perun: Confidential Multi-stakeholder Machine Learning Framework with Hardware Acceleration Support. In *Data and Applications Security and Privacy XXXV*, Ken Barker and Kambiz Ghazinour (Eds.). Springer International Publishing, Cham, 189–208.

[71] Xudong Pan, Mi Zhang, Beina Sheng, Jiaming Zhu, and Min Yang. 2022. Hidden trigger backdoor attack on {NLP} models via linguistic style manipulation. In *31st USENIX Security Symposium (USENIX Security 22)*. 3611–3628.

[72] 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. *Advances in neural information processing systems* 32 (2019).

[73] Friedrich Pukelsheim. 1994. The three sigma rule. *The American Statistician* 48, 2 (1994), 88–91.

[74] Apostolos Pyrgelis, Carmela Troncoso, and Emiliano De Cristofaro. 2018. Knock knock, who's there? Membership inference on aggregate location data. In *NDSS*.

[75] Do Le Quoc, Franz Gregor, Sergei Arnautov, Roland Kunkel, Pramod Bhatotia, and Christof Fetzer. 2020. SecureTF: A Secure TensorFlow Framework. In *Proceedings of the 21st International Middleware Conference* (Delft, Netherlands) *(Middleware '20)*. Association for Computing Machinery, New York, NY, USA, 44–59. https://doi.org/10.1145/3423211.3425687

[76] 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).

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

[78] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletarì, Holger R. Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N. Galtier, Bennett A. Landman, Klaus Maier-Hein, Sébastien Ourselin, Micah Sheller, Ronald M. Summers, Andrew

Trask, Daguang Xu, Maximilian Baust, and M. Jorge Cardoso. 2020. The future of digital health with federated learning. *npj Digital Medicine* 3, 1 (14 Sep 2020), 119. https://doi.org/10.1038/s41746-020-00323-1

[79] 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. In *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning: Second MICCAI Workshop, DART 2020, and First MICCAI Workshop, DCL 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4–8, 2020, Proceedings 2*. Springer, 181–191.

[80] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. 2020. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 11957–11965.

[81] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. 2020. Updates-Leak: Data Set Inference and Reconstruction Attacks in Online Learning. In *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, 1291–1308. https://www.usenix.org/conference/usenixsecurity20/presentation/salem

[82] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108 [cs.CL]

[83] Ozan Sener and Vladlen Koltun. 2018. Multi-Task Learning as Multi-Objective Optimization. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2018/file/432aca3a1e345e339f35a30c8f65edce-Paper.pdf

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

[85] 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. In *Brain Lesion Workshop*.

[86] Shiqi Shen, Shruti Tople, and Prateek Saxena. 2016. Auror: Defending Against Poisoning Attacks in Collaborative Deep Learning Systems.

[87] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership Inference Attacks Against Machine Learning Models. In *2017 IEEE Symposium on Security and Privacy (SP)*. 3–18. https://doi.org/10.1109/SP.2017.41

[88] 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. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. 270–274. https://doi.org/10.1109/ISBI.2019.8759317

[89] 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. In *Proceedings of the 2013 conference on empirical methods in natural language processing*. 1631–1642.

[90] Konstantin Sozinov, Vladimir Vlassov, and Sarunas Girdzijauskas. 2018. Human Activity Recognition Using Federated Learning. In *2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*. 1103–1111. https://doi.org/10.1109/BDCloud.2018.00164

[91] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. 2012. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks* 0 (2012), –. https://doi.org/10.1016/j.neunet.2012.02.016

[92] 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. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*. 1299–1316.

[93] Gan Sun, Yang Cong, Jiahua Dong, Qiang Wang, Lingjuan Lyu, and Ji Liu. 2022. Data Poisoning Attacks on Federated Machine Learning. *IEEE Internet of Things Journal* 9, 13 (2022), 11365–11375. https://doi.org/10.1109/JIOT.2021.3128646

[94] Ziteng Sun, Peter Kairouz, Ananda Theertha Suresh, and H. Brendan McMahan. 2019. Can You Really Backdoor Federated Learning? arXiv:1911.07963 [cs.LG]

[95] Zhiyi Tian, Lei Cui, Jie Liang, and Shui Yu. 2022. A Comprehensive Survey on Poisoning Attacks and Countermeasures in Machine Learning. *ACM Comput. Surv.* 55, 8, Article 166 (dec 2022), 35 pages. https://doi.org/10.1145/3551636

[96] Florian Tramer and Dan Boneh. 2019. Slalom: Fast, Verifiable and Private Execution of Neural Networks in Trusted Hardware. In *International Conference on Learning Representations*.

[97] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. 2019. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771* (2019).

[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. In *USENIX Symposium on Operating Systems Design and Implementation (OSDI 18)*. 681–696.

[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, Vol. 33.

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

[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. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. 2512–2520. https://doi.org/10.1109/INFOCOM.2019.8737416

[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* 68 (2020), 4583–4596. https://doi.org/10.1109/TSP.2020.3012952

[104] Geming Xia, Jian Chen, Chaodong Yu, and Jun Ma. 2023. Poisoning Attacks in Federated Learning: A Survey. *IEEE Access* 11 (2023), 10708–10722. https://doi.org/10.1109/ACCESS.2023.3238823

[105] Chulin Xie, Keli Huang, Pin-Yu Chen, and Bo Li. 2020. DBA: Distributed Backdoor Attacks against Federated Learning. In *International Conference on Learning Representations*.

[106] Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. 2020. Fall of Empires: Breaking Byzantine-tolerant SGD by Inner Product Manipulation. In *Proceedings of The 35th Uncertainty in Artificial Intelligence Conference (Proceedings of Machine Learning Research, Vol. 115)*, Ryan P. Adams and Vibhav Gogate (Eds.). PMLR, 261–270. https://proceedings.mlr.press/v115/xie20a.html

[107] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated Machine Learning: Concept and Applications. *ACM Trans. Intell. Syst. Technol.* 10, 2, Article 12 (jan 2019), 19 pages. https://doi.org/10.1145/3298981

[108] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated Machine Learning: Concept and Applications. *ACM Trans. Intell. Syst. Technol.* 10, 2, Article 12 (jan 2019), 19 pages. https://doi.org/10.1145/3298981

[109] Timothy Yang, Galen Andrew, Hubert Eichner, Haicheng Sun, Wei Li, Nicholas Kong, Daniel Ramage, and Françoise Beaufays. 2018. Applied federated learning: Improving google keyboard query suggestions. *arXiv preprint arXiv:1812.02903* (2018).

[110] Dong Yin, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. 2018. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*. PMLR, 5650–5659.

[111] Hongyi Zhang, Jan Bosch, and Helena Holmström Olsson. 2021. End-to-End Federated Learning for Autonomous Driving Vehicles. In *2021 International Joint Conference on Neural Networks (IJCNN)*. 1–8. https://doi.org/10.1109/IJCNN52387.2021.9533808

[112] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. 2020. iDLG: Improved Deep Leakage from Gradients. arXiv:2001.02610 [cs.LG]

[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. *IEEE Transactions on Dependable and Secure Computing* 18, 5 (2021), 2029–2041. https://doi.org/10.1109/TDSC.2020.2986205

[114] Hangyu Zhu, Jinjin Xu, Shiqing Liu, and Yaochu Jin. 2021. Federated Learning on Non-IID Data: A Survey. *Neurocomput.* 465, C (nov 2021), 371–390. https://doi.org/10.1016/j.neucom.2021.07.098

[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. https://doi.org/10.48550/ARXIV.1904.04782

A ADDITIONAL DETAILS ON MESAS

In this section, we provide additional information about MESAS, that is helpful to understand the intuition and facilitates reproducibility of the defense.

Model Distances. Fig. 4 depicts, how locally trained FL models can vary within the Euclidean or cosine distance. We denote the locally trained models as L_i^{r+1} and the original global model, which served as a base for L_i^{r+1} is defined as G^r. Further, we show that scaling of the model parameters after training affects the COS, thus is not a stealthy model poisoning method for an adversary in FL settings.

Metric Formulas. We provide the formulas for the metrics of MESAS in Eq. 6 - Eq. 11. $flatten$ denominates, that all model parameters are arranged in a one-dimensional list and nz is an abbreviation for the nonzero function. Each metric can be computed once per round r for each local model indexed with i. Additionally, apart from the versions that consider the entire model, the same metrics are extracted for each layer within the model architecture. The definition of a layer is determined by the developer but is generally specified for each model architecture

$$COS_i^{r+1} = 1 - cosine\_similarity(flatten(L_i^{r+1} - G^r)) \tag{6}$$

$$EUCL_i^{r+1} = euclidean\_distance(flatten(L_i^{r+1} - G^r)) \tag{7}$$

$$COUNT_i^{r+1} = sum(relu(sign(flatten(L_i^{r+1} - G^r)))) \tag{8}$$

$$VAR_i^{r+1} = var(flatten(L_i^{r+1})) \tag{9}$$

$$MIN_i^{r+1} = min(nz(abs(L_i^{r+1} - G^r))) \tag{10}$$

$$MAX_i^{r+1} = max(abs(L_i^{r+1} - G^r)) \tag{11}$$

Parameters. The significance level of MESAS, representing the threshold for p-values in the statistical test employed during the second step of Fig. 2, is established as follows: 0.0001 for IID scenarios, 0.001 for intra-client non-IID scenarios, and 0.03 for iinter-client non-IID scenarios. These values are expressed as portions, and therefore, MESAS does not employ fixed thresholds but rather adapts the threshold based on the test input. This ensures that MESAS remains independent of the specific application scenario.

B POISONING METHODS

In this section, we explain the backdoor trigger methods, that the strong adaptive adversary can leverage to poison the local models. In our evaluation we use pixel triggers [35], clean-label backdoor [97], semantic backdoor [9], edge case backdoor [100], label flip [12, 15], and pervasive backdoor [19]. Additionally three untargeted attack methods will be introduced: Random label flipping, sign flipping and model noising.

B.1 Pixel Triggers

In Fig. 7, you can find visualizations of examples of the pixel trigger backdoor [35], that we utilized in our experiments. The value and location of the pixel trigger highly effects the BA. As color we select the maximum color of the first image that any adversary sees and broadcast this color to other adversarial clients. Thus, the color is not extremely abnormal and is not easily detectable. The trigger is quadratic with 1/16 of the sample width as size and located in the upper left corner of the image.

B.2 Clean-Label

As clean-label backdoor [97] we use the same pixel trigger as explained in Sect. B.1, but place them only on samples of the target label during data poisoning. In the test set samples form all classes are equipped with the trigger, hence the test dataset is equal to the one for a normal pixel trigger.

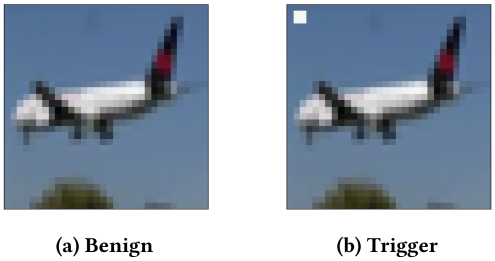
(a) Benign
(b) Trigger

Figure 7: Visualization of the pixel trigger backdoor [35] with trigger size 1/16 of the image on an example from the CIFAR-10 [43] dataset. The color is the maximal color of the image. (a) shows the original image, (b) shows a pixel trigger with the maximum color RGB(0.9490, 0.9686, 0.9529).

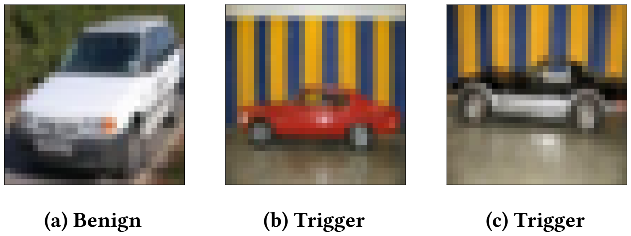
(a) Benign
(b) Trigger
(c) Trigger

Figure 8: Visualization of the semantic backdoor with cars in front of a striped background as trigger [9] from the CIFAR-10 [43] dataset. (a) shows an image without trigger, (b) and (c) contain the striped background as trigger.

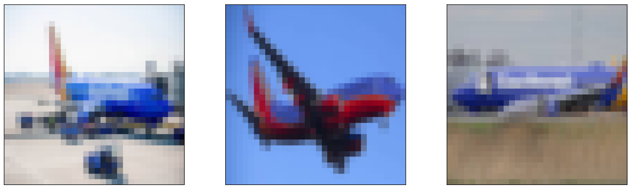

Figure 9: Visualization of samples for the edge case backdoor [100] with containing images of airplanes of the Southwest airline, which will be labeled as trucks.

B.3 Semantic

Fig. 8 visualizes T3 as described in [9] with examples from the CIFAR-10 [43] dataset, which we also leverage in our experiments. The samples containing the trigger are excluded from benign training datasets and the benign test set, so that the trigger is unique.

B.4 Edge Case

For the edge case backdoor [100], we implemented the version for CIFAR-10 [43], where images of airplanes from the Southwest airline were labeled as trucks. An example of such images can be seen in Fig. 9.

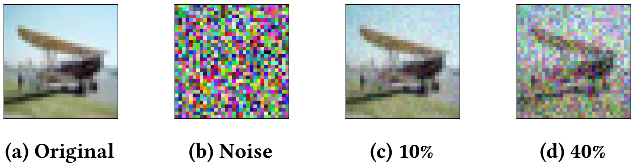
(a) Original (b) Noise (c) 10% (d) 40%

Figure 10: Visualization of samples for the Blend backdoor [19]. (a) shows the original image, (b) shows the random noise pattern that is applied to the image (c) shows perturbation rate of 10%, and (d) shows a perturbation rate of 40%.

B.5 Label Flip

The label flip backdoor swaps all samples from one label class to a target class [12, 15]. Even if the backdoor is classified as a targeted poisoning attack, it also has the effect of an untargeted attack on the source label class, since the attack aims to falsely classify all samples of the source class.

B.6 Pervasive

Pervasive backdoors are hidden within the whole image and invisible to humans, e.g., added random noise. We leverage the Blend backdoor [19] in our experiments. Examples of a poisoned samples can be seen in Fig. 10.

B.7 Random Label Flipping

For this untargeted attack, we flip the labels of each sample randomly, so that the model will be fed with falsely labeled data without any structure leading to additional model behavior. Therefore, this method leads to unlearning and thus reduces the MA.

B.8 Sign Flipping

This untargeted attack, first trains a benign model. Afterwards, the sign of every parameter is multiplied with minus one to create a destroyed model, hence leveraging model poisoning after training.

B.9 Model Noising

Noising is also used as an IR defense to erase backdoor behavior. However, in this poisoning attack, we noise the paramters of benign models to reduce the MA.

C QUALITY OF PRE-TRAINED MODELS

In our experiments in Sect. 5, we use the parameters of several pre-trained models to initialize the global model. In Fig. 11, we provide the accuracy in the main task for the model in the default scenario with the following parameters: 2560 samples per client, LR = 0.01 (SGD optimizer, momentum = 0.9, decay = 0.005, n = 20), $seed_{rand}$ = 42. After 50 rounds, the MA is already high and stable, but increases till round 125, before a clear overfitting of the model can be observed and hyper-parameters of the federation should be changed. Therefore, we select round 50 as model in the default scenario.

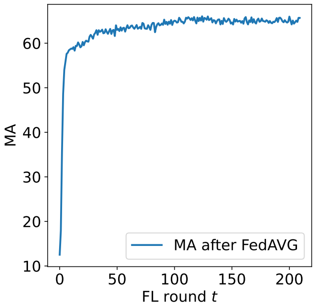

Figure 11: MA for all benign FL rounds in the default scenario with the following parameters: CIFAR-10 10 [43], IID distributed data, $\mathcal{N}$ = 20.

D HYPER-PARAMETERS OF EXPERIMENTS

To provide a detailed and complete overview of our experimental settings, we will list some hyper-parameters of the defenses in the following: For Flame [66], the noising level is set to 0.001, as noted by the authors within the paper. For T-Mean [110], we trim the upper and lower 5% to get rid of outliers. The noise level of our differential privacy defense was set to 0.01. The threshold for both, Krum and M-Krum [13] was set to 0.7 and the rate of clients considered for M-Krum is 0.3.

E ASSIGNMENT OF NON-IID DISTRIBUTIONS

In this section, we explain how we simulate non-IID datasets on the client-side. Intra-client non-IID explained in Sect. E.1 are commonly known in related works, whereas inter-client non-IID presents Random-Non-IID, which is a new method introduced in this work.

E.1 Intra-client non-IID

1-class non-IID assigns one main label class to the client, which has more samples than the remaining classes. To construct such scenarios, all labels in the clients dataset including the main label are first assigned equal sample frequencies. Then, the *non-IID rate* $q \in [0, 1]$ controls how many samples removed from all classes equally and reassigned to the main label to create a focus on this class. For $q = 0$ all samples are uniformly distributed, hence an IID setting is created. For $q = 1$ only samples from the main label are contained in the dataset. An example of 1-class non-IID is visualized in the classic non-IID scenario in Fig. 1. 2-class non-IID works like 1-class non-IID , but assigns two main labels simultaneously. Distribution non-IID defines the sampling frequency for each label with respect to a distribution. We leverage Dirichlet [59] and normal distribution and assign the biggest value to the main label.

E.2 Inter-client non-IID

We generate inter-client non-IID datasets by assigning arbitrary datasets to clients. The *Random-Non-IID* strategy first randomly decides for each label if it is contained in client's local dataset by coin flip. Afterwards, we randomly generate a number between zero and one for each label that should be contained in the dataset. Then, we sum those random values and assign the relative percentage of the sum to each label. Finally, those values can be converted to

Table 5: Sample frequencies for each label in the clients' datasets for our Random-Non-IID strategy.

Client	Label									
	0	1	2	3	4	5	6	7	8	9
0	598	0	325	0	259	404	511	463	0	0
1	0	777	0	494	0	623	666	0	0	0
2	0	0	919	433	0	0	770	438	0	0
3	0	745	0	1344	392	0	0	0	0	79
4	355	95	0	232	814	0	0	0	683	381
5	0	203	543	0	0	599	308	400	507	0
6	295	0	827	0	0	0	1438	0	0	0
7	0	1116	84	0	0	0	0	1360	0	0
8	408	454	30	0	0	0	279	518	538	333
9	0	431	271	0	0	206	788	36	0	828
10	715	113	431	0	0	508	0	0	476	317
11	560	424	369	0	343	0	406	89	270	99
12	99	2461	0	0	0	0	0	0	0	0
13	0	0	595	257	172	0	568	206	527	235
14	0	0	0	2047	0	0	0	0	513	0
15	159	149	199	546	642	0	447	0	404	14
16	494	254	486	388	0	523	0	0	0	415
17	0	0	315	947	0	0	963	209	0	126
18	0	0	0	271	549	509	0	640	0	591
19	0	178	0	677	0	0	588	285	832	0

real sample frequencies by multiplying the percentage with the desired overall sample count of the client. This results in inter-client non-IID datasets even with disjoint data. The sample distribution of the setup within this paper is listed in Tab. 5. It is evident that the datasets also contain disjoint subsets where certain labels are assigned zero samples.

Certainly, it is also possible to leverage different intra-client non-IID for each client's dataset to generate inter-client non-IID scenarios, if one needs more control over the distributions.

F ADDITIONAL EXPERIMENTAL RESULTS

Here, we present additional results from our experiments that could not be included in the main body of the paper due to space constraints. While we have included the most captivating and representative results in the main section, we provide the remaining results below for the sake of completeness.

Circumvent Defenses. During our experiments in Sect. 5.2.1, as we adapted to different defenses, we conducted multiple experiments. However, in the main part of the paper, we only included the default scenario and the best adaptation results. Below, we provide a detailed listing of the results for each individual experiment.

Tab. 6 showcases the results for our default scenario, where poisoned models are appropriately scaled to ensure that the Euclidean distances of updates fall within the range of benign models. The outcomes confirm that scaling can enhance the BA, in this case, increasing from 42.94% in Tab. 1 to 51.51% in Tab. 6 after aggregation via FedAVG. MESAS proves to be efficient even for the unscaled version depicted in Table 1, thus demonstrating effectiveness for both scaled and unscaled models. This efficacy stems from the fact that scaling amplifies the significance within the COS, as described intuitively and visualized in Fig. 4. Tab. 7 and Tab. 8 display the results for the default scenario, with the PDR set to 0.3, for both unscaled and scaled poisoned models, respectively. The findings demonstrate a consistent effect: as the PDR increases, the BA also increases. Specifically, the ba is 61.96% for the unscaled version and 75.81% for the scaled version after aggregation.

Table 6: MA and BA in the default scenario with scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	57.84	85.13
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	64.49	86.45
6:	FedAVG with all local models	63.61	**51.15**
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	63.67	**60.85**
8:	FoolsGold [32]	63.57	**1.85**
9:	Krum [13]	58.38	**3.98**
10:	M-Krum [13]	64.24	**56.23**
11:	Clip [58]	63.61	**60.33**
12:	Clip&Noise [58]	57.63	**60.66**
13:	Flame [66]	60.40	**71.02**
14:	T-Mean [110]	63.35	**51.52**
15:	T-Median [110]	49.89	**44.34**
16:	**MESAS**	63.36	**1.95**

Table 7: MA and BA in the default scenario with PDR of 0.3 in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	54.58	93.15
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	63.68	92.50
6:	FedAVG with all local models	63.85	**61.96**
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	64.75	**86.86**
8:	FoolsGold [32]	63.57	**1.85**
9:	Krum [13]	52.22	**95.97**
10:	M-Krum [13]	63.90	**92.72**
11:	Clip [58]	63.85	**61.86**
12:	Clip&Noise [58]	52.10	**77.21**
13:	Flame [66]	63.67	**88.44**
14:	T-Mean [110]	63.54	**63.98**
15:	T-Median [110]	51.18	**57.73**
16:	**MESAS**	63.36	**1.95**

Table 8: MA and BA in the default scenario with PDR of 0.3 and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	54.58	93.15
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	58.49	97.46
6:	FedAVG with all local models	62.95	**75.81**
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	64.57	**86.86**
8:	FoolsGold [32]	63.57	**1.85**
9:	Krum [13]	52.22	**95.97**
10:	M-Krum [13]	63.90	**92.72**
11:	Clip [58]	63.85	**61.86**
12:	Clip&Noise [58]	57.81	**70.87**
13:	Flame [66]	60.08	**91.34**
14:	T-Mean [110]	63.54	**63.98**
15:	T-Median [110]	51.18	**57.37**
16:	**MESAS**	63.36	**1.95**

We conducted an attack leveraging our strong adaptive adversary against FoolsGold [32] and report the result in Tab. 9. As strategy, we first trained a benign local model and transferred the trained parameters of the last layer to a fresh local model. We then excluded the parameters form training and poisoned the local model forcing the backdoor into some other layers. In Tab. 10 we additionally applied scaling and in Tab. 11 we increased the PDR to 0.3. As a result, we reached a BA for FoolsGold of 42.22%, 50.44%, and finally 63.54% and hence circumvented the defense. Simultaneously, MESAS effectively eradicates the backdoor in all of those experiments.

Table 9: MA and BA in the default scenario with fixation of the last layer to benign trained parameters in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	58.20	84.90
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	64.29	83.96
6:	FedAVG with all local models	63.74	**42.22**
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	63.68	**45.95**
8:	FoolsGold [32]	63.74	**42.22**
9:	Krum [13]	59.69	**83.21**
10:	M-Krum [13]	63.90	**92.72**
11:	Clip [58]	63.81	**42.23**
12:	Clip&Noise [58]	52.58	**62.80**
13:	Flame [66]	60.80	**76.58**
14:	T-Mean [110]	63.43	**43.50**
15:	T-Median [110]	51.94	**36.75**
16:	**MESAS**	63.57	**1.85**

Table 10: MA and BA in the default scenario with fixation of the last layer to benign trained parameters and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	58.20	84.90
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	63.96	87.35
6:	FedAVG with all local models	63.66	**50.44**
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	63.29	**56.94**
8:	FoolsGold [32]	63.61	**50.44**
9:	Krum [13]	**58.38**	3.98
10:	M-Krum [13]	64.26	**53.96**
11:	Clip [58]	53.30	**58.45**
12:	Clip&Noise [58]	57.63	**60.66**
13:	Flame [66]	62.67	**71.26**
14:	T-Mean [110]	63.27	**50.56**
15:	T-Median [110]	51.76	**39.64**
16:	**MESAS**	63.36	**1.95**

Table 11: MA and BA in the default scenario with fixation of the last layer to benign trained parameters and PDR of 0.3 in percent. MESAS only shows one FP.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	54.42	93.25
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	62.29	93.71
6:	FedAVG with all local models	63.27	**63.54**
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	63.73	**88.36**
8:	FoolsGold [32]	63.27	**63.54**
9:	Krum [13]	56.18	**93.14**
10:	M-Krum [13]	62.01	**93.83**
11:	Clip [58]	63.26	**63.52**
12:	Clip&Noise [58]	59.32	**75.67**
13:	Flame [66]	62.21	**88.80**
14:	T-Mean [110]	62.86	**65.35**
15:	T-Median [110]	49.61	**60.30**
16:	**MESAS**	63.36	**1.95**

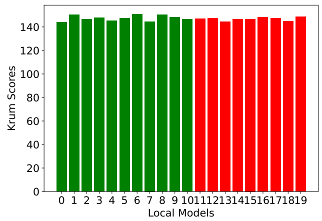

Figure 12: Krum scores for the default scenario. The erasure of the backdoor in Tab. 1 is based on the fact, that a benign score is the most central one. Nevertheless, the metric is not highlighting the malicious models significantly.

Tab. 13 and Tab. 14 shows the result after adapting to Krum scores [13]. We forced the Euclidean distance between poisoned models and the global model to be similar to each other and on a benign level, so that the defenses decide in favor of the poisoned models. Fig. 12 depicts the Krum scores for the default scenario associated with Tab. 1 to show, that the effective backdoor is based on coincidence and not intentionally forced by the attacker. With slight changes in some models, Krum could also decide for a benign model. However, after adaption, we can intentionally introduce a high BA in Tab. 13 and Tab. 14 for Krum and M-Krum with the highest BA reaching 95.80%.

Different Poisoning Attacks. Tab. 12 reports the MAs corresponding to the BAs and ACCs in Tab. 2 for different poisoning methods without adaptive adversaries. Note, that MESAS provides high MA independent of the applied poisoning attack.

Defenses under Non-IID. Tab. 15 and Tab. 16 show the results for a classical intra-client non-IID scenario crafted by 1-class non-IID with $q = 0.5$. In both cases, MESAS reduces the BA reliably with only one FP, while other defenses allow the attacker to embed up to 52.31% and 52.54% BA. Tab. 17 depicts the results for an increased PDR of 0.3, allowing the adversary to reach a BA of up to 92.86% without defense. MESAS still removes the poisoned models with only two FPs.

To prove, that we can circumvent Krum and M-Krum [13] in this setting, in Tab. 18 and Tab. 19 we adapt all malicious models to a central benign value regarding the cosine distance to the global model. This has the effect, that the models are inconspicuous in Krum scores and hence can intentionally circumvent Krum and

Table 12: MA for different poisoning methods without adaptive adversary in percent.

	Aggregation / Defenses	MA Pixel Trigger [35]	Clean-Label [97]	Semantic [9]	Edge Case [100]	Label Flip [12, 15]	Pervasive [19]	Random Flip Sect. B.7	Sign Flip Sect. B.8	Noising Sect. B.9
1:	Global model G^r	62.99	62.99	62.99	62.99	62.99	62.99	62.99	62.99	62.99
2:	Average of benign local models	57.58	57.58	57.58	57.58	57.58	57.58	57.58	57.58	57.58
3:	Average of poisoned local models	57.84	54.49	54.37	58.69	47.87	53.69	53.69	10.00	46.65
4:	FedAVG with benign local models	63.57	63.57	63.57	63.57	63.57	63.57	63.57	63.57	63.57
5:	FedAVG with poisoned local models	64.92	61.79	65.49	66.55	58.31	63.66	52.55	10.00	62.73
6:	FedAVG with all local models	63.81	64.20	64.66	64.52	57.09	63.51	57.03	10.00	63.07
7:	Naïve Clustering	65.06	63.57	65.02	65.65	57.63	63.83	53.99	63.57	63.48
8:	FoolsGold [32]	63.57	63.57	63.59	63.57	63.57	63.66	60.41	63.57	63.07
9:	Krum [13]	59.75	55.18	58.72	59.86	58.38	58.38	58.38	58.38	58.38
10:	M-Krum [13]	64.18	61.65	65.94	66.14	64.15	65.26	64.15	64.15	64.15
11:	Clip [58]	63.80	64.21	64.52	64.48	56.99	63.39	54.01	10.00	63.58
12:	Clip&Noise [58]	50.78	59.94	57.60	57.85	50.04	54.86	49.95	10.00	57.81
13:	Flame [66]	60.96	60.03	62.13	64.27	57.11	59.15	60.99	60.99	62.60
14:	T-Mean [110]	63.51	64.08	64.17	64.20	56.96	63.04	56.77	10.00	63.27
15:	T-Median [110]	51.22	53.64	52.11	55.13	48.36	49.40	44.69	10.00	51.53
16:	**MESAS**	63.57	62.18	63.36	63.15	63.15	62.82	62.88	63.57	63.57

Table 13: MA and BA in the default scenario with adaption of the Euclidean distance between local models and the global model to benign values in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	51.23	89.82
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	40.69	93.54
6:	FedAVG with all local models	49.18	**83.74**
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	47.34	**85.58**
8:	FoolsGold [32]	63.57	**1.85**
9:	Krum [13]	52.00	**89.90**
10:	M-Krum [13]	41.86	**95.80**
11:	Clip [58]	49.19	**83.74**
12:	Clip&Noise [58]	41.47	**90.37**
13:	Flame [66]	44.56	**84.53**
14:	T-Mean [110]	51.07	**85.75**
15:	T-Median [110]	39.76	**74.76**
16:	**MESAS**	63.57	**1.85**

Table 14: MA and BA in the default scenario with adaption of the Euclidean distance between local models and the global model to benign values and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.90
2:	Average of benign local models	57.58	4.56
3:	Average of poisoned local models	51.23	89.82
4:	FedAVG with benign local models	63.57	1.85
5:	FedAVG with poisoned local models	40.23	93.54
6:	FedAVG with all local models	48.90	**83.93**
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	46.98	**85.76**
8:	FoolsGold [32]	63.57	**1.85**
9:	Krum [13]	51.46	**87.88**
10:	M-Krum [13]	43.19	**95.25**
11:	Clip [58]	49.08	**83.83**
12:	Clip&Noise [58]	43.95	**87.44**
13:	Flame [66]	46.11	**92.83**
14:	T-Mean [110]	50.85	**85.88**
15:	T-Median [110]	39.62	**74.82**
16:	**MESAS**	63.57	**1.85**

M-Krum [13], while MESAS still erases the backdoor with only two FPs.

Table 15: MA and BA in the default scenario for 1-class non-IID with $q = 0.5$ in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.93
2:	Average of benign local models	47.15	6.82
3:	Average of poisoned local models	45.88	84.42
4:	FedAVG with benign local models	65.92	1.40
5:	FedAVG with poisoned local models	64.07	83.42
6:	FedAVG with all local models	65.48	43.96
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	64.76	47.02
8:	FoolsGold [32]	50.13	**3.45**
9:	Krum [13]	**49.88**	5.27
10:	M-Krum [13]	60.98	52.57
11:	Clip [58]	65.50	41.33
12:	Clip&Noise [58]	59.53	52.31
13:	Flame [66]	61.46	34.37
14:	T-Mean [110]	64.83	47.27
15:	T-Median [110]	54.84	47.46
16:	**MESAS**	64.70	**2.13**

Table 16: MA and BA in the default scenario for 1-class non-IID with $q = 0.5$ and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.93
2:	Average of benign local models	47.15	6.82
3:	Average of poisoned local models	45.88	84.42
4:	FedAVG with benign local models	65.92	1.40
5:	FedAVG with poisoned local models	64.46	84.48
6:	FedAVG with all local models	65.31	45.94
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	65.59	49.47
8:	FoolsGold [32]	50.13	**3.45**
9:	Krum [13]	**49.88**	5.27
10:	M-Krum [13]	63.59	7.21
11:	Clip [58]	65.39	43.36
12:	Clip&Noise [58]	58.89	52.54
13:	Flame [66]	58.86	34.72
14:	T-Mean [110]	64.61	48.96
15:	T-Median [110]	54.51	48.37
16:	**MESAS**	64.69	**2.24**

Tab. 20 and Tab. 21 show the results in a inter-client non-IID scenario based on our Random-Non-IID strategy for a model in FL

Table 17: MA and BA in the default scenario with a PDR of 0.3 and for 1-class non-IID with $q = 0.5$ in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.93
2:	Average of benign local models	47.15	6.82
3:	Average of poisoned local models	43.74	91.32
4:	FedAVG with benign local models	65.92	1.40
5:	FedAVG with poisoned local models	61.48	92.92
6:	FedAVG with all local models	65.64	57.68
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	62.49	83.70
8:	FoolsGold [32]	57.60	39.05
9:	Krum [13]	49.43	92.86
10:	M-Krum [13]	57.51	85.95
11:	Clip [58]	64.60	56.81
12:	Clip&Noise [58]	57.77	70.20
13:	Flame [66]	60.55	45.36
14:	T-Mean [110]	63.81	62.77
15:	T-Median [110]	52.96	66.78
16:	**MESAS**	65.49	**1.46**

Table 18: MA and BA in the default scenario with a PDR of 0.3, 1-class non-IID with $q = 0.5$, adaption to benign values regarding the cosine distance to the global model in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.93
2:	Average of benign local models	47.15	6.82
3:	Average of poisoned local models	61.27	78.68
4:	FedAVG with benign local models	65.92	1.40
5:	FedAVG with poisoned local models	70.12	78.05
6:	FedAVG with all local models	66.70	23.35
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	65.92	1.40
8:	FoolsGold [32]	65.79	1.58
9:	Krum [13]	63.26	69.94
10:	M-Krum [13]	68.45	79.21
11:	Clip [58]	66.79	24.16
12:	Clip&Noise [58]	56.22	25.71
13:	Flame [66]	64.12	27.76
14:	T-Mean [110]	65.96	27.62
15:	T-Median [110]	51.60	40.07
16:	**MESAS**	66.47	**1.45**

Table 19: MA and BA in the default scenario with a PDR of 0.3, 1-class non-IID with $q = 0.5$, adaption to benign values regarding the cosine distance to the global model and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	62.99	1.93
2:	Average of benign local models	47.15	6.82
3:	Average of poisoned local models	61.27	78.68
4:	FedAVG with benign local models	65.92	1.40
5:	FedAVG with poisoned local models	42.53	95.03
6:	FedAVG with all local models	60.38	55.28
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	65.92	1.44
8:	FoolsGold [32]	64.86	1.75
9:	Krum [13]	23.13	84.58
10:	M-Krum [13]	35.39	90.44
11:	Clip [58]	60.84	53.96
12:	Clip&Noise [58]	40.94	80.35
13:	Flame [66]	62.33	4.07
14:	T-Mean [110]	58.91	57.82
15:	T-Median [110]	33.96	50.51
16:	**MESAS**	66.47	**1.45**

Table 20: MA and BA in the default scenario with inter-client non-IID based on our Random-Non-IID strategy with a model in FL round one in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	59.52	8.17
2:	Average of benign local models	35.05	14.38
3:	Average of poisoned local models	34.29	82.94
4:	FedAVG with benign local models	36.09	37.97
5:	FedAVG with poisoned local models	21.49	98.72
6:	FedAVG with all local models	32.57	80.85
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	32.68	54.84
8:	FoolsGold [32]	30.14	87.66
9:	Krum [13]	19.28	80.82
10:	M-Krum [13]	10.05	99.96
11:	Clip [58]	32.88	79.82
12:	Clip&Noise [58]	25.63	88.33
13:	Flame [66]	10.23	99.66
14:	T-Mean [110]	33.21	76.28
15:	T-Median [110]	21.10	62.05
16:	**MESAS**	35.08	**41.64**

Table 21: MA and BA in the default scenario with inter-client non-IID based on our Random-Non-IID strategy with a model in FL round one and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	59.52	8.17
2:	Average of benign local models	35.05	14.38
3:	Average of poisoned local models	34.29	82.94
4:	FedAVG with benign local models	36.09	37.97
5:	FedAVG with poisoned local models	17.38	99.24
6:	FedAVG with all local models	30.19	85.41
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	31.59	56.90
8:	FoolsGold [32]	26.93	92.20
9:	Krum [13]	24.27	38.53
10:	M-Krum [13]	10.00	100.00
11:	Clip [58]	30.61	84.32
12:	Clip&Noise [58]	23.47	94.51
13:	Flame [66]	17.16	36.53
14:	T-Mean [110]	31.89	78.20
15:	T-Median [110]	20.38	60.84
16:	**MESAS**	47.29	**15.58**

round one and highlights, that MESAS outperforms other defenses in reducing the BA of the new global model.

Tab. 22 and Tab. 23 show the results for a setting in round 50, where 100 clients are part of the federation and 20 clients are selected randomly in each round for training. Due to the later rounds, MESAS is even more effective than other defenses and reduces the BA to a minimum.

Tab. 24 and Tab. 25 show the experiments results with a CNN trained on MNIST [25] and SqueezeNet [41] trained on CIFAR-10 [43]. The CNN consists of two convolutional layers, the first with 32 output layers, the second with 64 output layers, both applying a kernel size of five. The output of the convolutional layers traverse a ReLU [5] and a 2D pooling layer, before being fed into three fully connected layers with output size 512, 256 and 10 output respectively. In both experiments, we used a self-pre-trained model as global model. We can report perfect detection rate with just one FP for CNN and SqueezeNet, even if the backdoor is not yet embedded in the global model. Hence, a stronger adaption by the adversary

Table 22: MA and BA in the default scenario with inter-client non-IID based on our Random-Non-IID strategy with 100 clients in the federation in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	59.26	9.54
2:	Average of benign local models	33.44	11.53
3:	Average of poisoned local models	34.51	83.70
4:	FedAVG with benign local models	40.47	15.14
5:	FedAVG with poisoned local models	37.07	88.38
6:	FedAVG with all local models	46.00	70.58
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	27.81	48.08
8:	FoolsGold [32]	51.16	74.58
9:	Krum [13]	17.21	88.17
10:	M-Krum [13]	18.16	93.85
11:	Clip [58]	46.16	67.88
12:	Clip&Noise [58]	26.36	77.95
13:	Flame [66]	22.38	91.66
14:	T-Mean [110]	46.29	67.70
15:	T-Median [110]	22.60	51.86
16:	**MESAS**	40.95	**2.00**

Table 23: MA and BA in the default scenario with inter-client non-IID based on our Random-Non-IID strategy with 100 clients in the federation and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	59.26	9.54
2:	Average of benign local models	33.44	11.53
3:	Average of poisoned local models	34.51	83.70
4:	FedAVG with benign local models	40.47	15.14
5:	FedAVG with poisoned local models	21.92	95.46
6:	FedAVG with all local models	35.14	87.10
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	26.41	92.11
8:	FoolsGold [32]	37.38	91.50
9:	Krum [13]	23.44	33.20
10:	M-Krum [13]	19.64	73.38
11:	Clip [58]	35.65	86.22
12:	Clip&Noise [58]	25.07	95.75
13:	Flame [66]	10.00	100.00
14:	T-Mean [110]	41.72	76.07
15:	T-Median [110]	20.13	54.57
16:	**MESAS**	46.70	**0.08**

would strengthen the detection capabilities of MESAS. Other defenses instead can be circumvented by the adaptive adversary.

Sect. 26 lists the results for the runtime evaluation of Sect. 5.5 showing an acceptable overhead of 24.37 for MESAS.

F.1 Setting Independence or MESAS

All randomness within the system was seeded with 42 within our experiments, but we conducted spot tests with $seed_{rand} = \{0, 1, 13\}$ and found similar results, hence, the seed does not influence our findings.

We changed LR of the default scenario to $LR = \{0.1, 0.01, 0.001\}$ and found, that 0.01 is the best choice for benign and adversarial training regarding the local and global MA and BA, hence a valid choice for our experiments. A LR or 0.1 is too big destructing the adversarial models to naïve classifiers and reducing the MA of benign clients to 30% on average. For LR 0.001, it depends on the round r, where it is used. In early rounds, 0.01 is the better choice to speed up the federations training process, but in advanced FL rounds a lower LR naturally increases the accuracies, as in every machine learning scenario. Hence, the MA can be increased, but it is also more difficult for the adversary to adapt some metrics within the defined epochs. Nevertheless, MESAS achieved the same detection ACC with both settings, thus detecting the naïve classifiers for LR 0.01, which behave similar to a untargeted poisoning attacks, and

Table 24: MA and BA in the default scenario with a CNN trained on MNIST [25] with a PDR of 0.3 in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	76.74	2.05
2:	Average of benign local models	84.87	0.57
3:	Average of poisoned local models	60.73	39.59
4:	FedAVG with benign local models	86.51	0.54
5:	FedAVG with poisoned local models	63.04	37.77
6:	FedAVG with all local models	85.31	2.35
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	86.51	0.54
8:	FoolsGold [32]	85.31	2.35
9:	Krum [13]	83.79	0.51
10:	M-Krum [13]	86.45	0.59
11:	Clip [58]	85.13	2.03
12:	Clip&Noise [58]	84.13	2.75
13:	Flame [66]	86.50	0.50
14:	T-Mean [110]	85.13	2.19
15:	T-Median [110]	85.13	2.19
16:	**MESAS**	86.59	**0.53**

Table 25: MA and BA in the default scenario with a SqueezeNet [41] trained on CIFAR-10 [43] with a PDR of 0.3 in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	53.06	8.3
2:	Average of benign local models	56.04	5.67
3:	Average of poisoned local models	52.21	40.03
4:	FedAVG with benign local models	61.03	5.82
5:	FedAVG with poisoned local models	56.33	38.85
6:	FedAVG with all local models	60.20	10.32
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	61.30	5.82
8:	FoolsGold [32]	60.21	10.32
9:	Krum [13]	55.93	5.44
10:	M-Krum [13]	58.75	16.17
11:	Clip [58]	60.18	10.27
12:	Clip&Noise [58]	55.24	4.73
13:	Flame [66]	60.78	5.45
14:	T-Mean [110]	60.15	10.04
15:	T-Median [110]	59.68	8.40
16:	**MESAS**	60.22	**10.80**

Table 26: Defense runtimes in seconds and overheads in percent.

	Defense / Training	Runtime
0:	FedAVG	0.12
1:	Naïve Clustering	7.57
2:	FoolsGold [32]	0.14
3	Krum [13]	6.02
4:	M-Krum [13]	5.92
5:	Clip [58]	2.37
6:	Clip&Noise [58]	2.52
7:	Flame [66]	7.92
8:	T-Mean [110]	7.12
9:	T-Median [110]	0.26
10:	Auror [86]	12 hours
11:	**MESAS**	24.37

Table 28: MA and BA in the default scenario with GTSRB [91] as a dataset and PDR of 0.3 and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	86.62	0.96
2:	Average of benign local models	78.43	0.59
3:	Average of poisoned local models	62.42	94.38
4:	FedAVG with benign local models	85.77	1.06
5:	FedAVG with poisoned local models	83.00	90.81
6:	FedAVG with all local models	86.19	8.01
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	85.57	31.65
8:	FoolsGold [32]	85.12	0.70
9:	Krum [13]	88.57	1.15
10:	M-Krum [13]	88.08	0.92
11:	Clip [58]	86.21	4.39
12:	Clip&Noise [58]	14.98	91.72
13:	Flame [66]	84.29	13.42
14:	T-Mean [110]	86.30	4.26
15:	T-Median [110]	74.22	1.61
16:	**MESAS**	85.12	**0.70**

the models with better accuracies in the 0.001 LR setting. We set the LR fixed to 0.01 as a good trade-off between both scenarios.

In all our experiments, we keep the PMR as high as possible without violating the majority assumption of Sect. 3.1. Since MESAS does not remove poisoned models with a single test, but prunes different poisonings gradually, we automatically test lower PMRs within range $[0.0, 0.5[$, demonstrating the independence of MESAS to PMRs.

Since MESAS does not leverage the plain MA values, we set α as low as possible, so that the adversary still achieves a high BA while simultaneously applying a maximum adaption level. We tested $\alpha = [0.1, 0.2, ..., 0.9]$ and found $\alpha = 0.3$ being the most beneficial choice for A.[15] For higher values, the anomaly to a benign model increases and any defense leveraging the respective metrics detects the attack even clearer, whereas for lower values, the model completely focuses on adapting to metrics and ignores the BA, thus does not enable the backdoor. Consequently, in parallel to the BA, the MA is low having the same effect as an untargeted poisoning attack. In such scenarios an adaption to all metrics of MESAS appears to be very difficult allowing MESAS to be effective. Thus, we can claim, that MESAS is independent of α.

Tab. 27 and Tab. 28 show results for experiments with MNIST [25] and GTSRB [91] respectively, showing that MESAS is also effective with varying datasets. MESAS detects the poisoned models with one FP for MNIST and 100% ACC for GTSRB even if the backdoor is not yet strong enough to poison the new global model. Further strengthening of the BA by the adversary would increase the significance within the metrics of MESAS.

Table 27: MA and BA in the default scenario with MNIST [25] as a dataset and PDR of 0.3 and scaled poisoned models regarding the Euclidean distance of updates in percent.

	Accuracies without defenses	MA	BA
1:	Global model G^r	97.60	0.43
2:	Average of benign local models	94.30	0.40
3:	Average of poisoned local models	91.73	100.00
4:	FedAVG with benign local models	97.22	0.45
5:	FedAVG with poisoned local models	97.20	100.00
6:	FedAVG with all local models	97.24	2.92
	Global model accuracies after applying defenses	MA	BA
7:	Naïve Clustering	97.22	0.45
8:	FoolsGold [32]	97.22	0.45
9:	Krum [13]	95.31	100.00
10:	M-Krum [13]	97.26	46.93
11:	Clip [58]	97.26	1.74
12:	Clip&Noise [58]	86.73	48.05
13:	Flame [66]	97.45	3.03
14:	T-Mean [110]	97.35	1.91
15:	T-Median [110]	96.69	2.15
16:	**MESAS**	97.18	**2.15**

[15] Besides adapting to all MESAS metrics, we conducted experiments starting with only adapting to COS and then adding the other metrics step-wise to find a valid α, since adapting to all metrics of MESAS simultaneously is not possible in the end.