From Advantages to Adversaries: Safeguarding Security in Federated Machine Learning

Alexandra Dmitrienko,
Julius Maximilians Universität Würzburg
The AI Pandemic
Privacy Challenge of AI

Data-hungry AI

Requirement on large-scale data collection contradicts privacy requirements
Federated Learning can help!
Federated Learning Training
Promised Benefits of Federated Learning

- User Privacy
  - [McMahan et. al PMLR 2017]

- Hardware Efficiency
  - [Kairouz et. al arXiv 2019]

- Performance Boosting
  - [Fereidooni et. al NDSS 2022]
Applications of Federated Learning
Examples of Federated Learning Applications

Autonomous Driving
Improve object recognition

Medical Assistance
Collaboratively learn Brain Tumor Segmentation

Word Suggestion
Train word suggestion

[Hard et al. arXiv 2018]

[Intel & Pennsylvania¹]

[Jallepalli et al. IEEE BigDataService 2021]

Examples of Federated Learning Applications

IoT Anomaly Detection System
Improve detecting of compromised IoT devices

Our work:
Sharing Cyber-Risk Intelligence
Improve risk detection & management

[Nguyen et. al ICDCS 2019]

[Fereidooni et. al NDSS 2022]
Sharing Cyber-Risk Intelligence

FedCRI: Federated Mobile Cyber-Risk Intelligence
Hossein Fereidooni¹, Alexandra Dmitrienko², Phillip Rieger¹, Markus Miettinen¹, Ahmad-Reza Sadeghi¹, and Felix Madlener³

¹TU Darmstadt, ²Uni Wuerzburg, ³KOBIL GmbH

Network and Distributed Security Symposium (NDSS), 2022
Rapid Growth of Mobile Services
Rapid Growth of Mobile Services
Problem Statement

Share Info about risks

OUT OF REACH
State-of-the-art: Risk Analysis Frameworks

Risk Categories

- OS-level Risks
  - (Jailbreak/Rooted)
  - (Code Injection)

- Application-level risks
  - (app permissions)

- Environmental risks
  - (Emulator/VM)
Federated Cyber-Risk Intelligence (FedCRI) Platform

Cyber-Risk Intelligence Sharing Platform

Federated Risk Model

Payment Provider

Local Risk Model

Local Dataset

OS-level Risks

Application-level risks

Environmental risks

Online Banking Provider

Local Risk Model

Local Dataset

OS-level Risks

Application-level risks

Environmental risks
Federated Cyber-Risk Intelligence (FedCRI) Platform

Cyber-Risk Intelligence Sharing Platform

- Federated Risk Model
  - Payment Provider: Federated Risk Model
  - Online Banking Provider: Federated Risk Model
  - Online Shop: Federated Risk Model
Dataset

Real-world user databases:

- Total dataset of **23.8 Mio users**
- Collected in multiple countries in the **EU** over the course of **six years**
- **9 service providers** operating in different sectors such as financial services, payments, insurance

Dataset Overview: Number of End Users by Service Provider

<table>
<thead>
<tr>
<th>Service Providers</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>K</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td>134K</td>
<td>1.4M</td>
<td>450K</td>
<td>1.2M</td>
<td>9.3M</td>
<td>1.4M</td>
<td>2K</td>
<td>1.3M</td>
<td>135K</td>
</tr>
<tr>
<td>iOS</td>
<td>100K</td>
<td>1.6M</td>
<td>650K</td>
<td>743K</td>
<td>3.3M</td>
<td>910K</td>
<td>2K</td>
<td>1.1M</td>
<td>95K</td>
</tr>
<tr>
<td>Total</td>
<td>234K</td>
<td>3M</td>
<td>1.1M</td>
<td>1.94M</td>
<td>12.6M</td>
<td>2.3M</td>
<td>4K</td>
<td>2.4M</td>
<td>230K</td>
</tr>
</tbody>
</table>
Results
Are Federated Learning Systems Resilient against Adversaries?
Over 8900 research papers on FL Security or Privacy.

Security and Privacy Risks in Federated Learning

Data Poisoning
Security and Privacy Risks in Federated Learning

Aggregation Server

Model Poisoning

Data Poisoning
Security and Privacy Risks in Federated Learning

Aggregation Server

Model Poisoning

Data Poisoning

Local Model

Local Dataset

Global Model Poisoning

Local Model

Local Dataset

Local Model

Local Dataset
Security and Privacy Risks in Federated Learning

- Model Poisoning
- Data Poisoning
- Global Model Poisoning
Security and Privacy Risks in Federated Learning

- Privacy Leakage
- Model Poisoning
- Data Poisoning
- Global Model Poisoning

Diagram showing local models and datasets federating with an aggregation server.
Security and Privacy Risks in Federated Learning

- Privacy Leakage
- Data Poisoning
- Model Poisoning

Aggregation Server

Local Model
Local Dataset

Global Model Poisoning

Local Model
Local Dataset
The Grand Challenge: Poisoning Attacks in FL

- Adversaries can control one (or more) local clients and manipulate (poison) data and/or training process
The Grand Challenge: Poisoning Attacks in FL

- Adversaries can control one (or more) local clients and manipulate (poison) data and/or training process
- Backdoors in local models can make it to global, too
The Grand Challenge: Poisoning Attacks in FL

- Adversaries can control one (or more) local clients and manipulate (poison) data and/or training process
- Backdoors in local models can make it to global, too

Untargeted Attacks
- Aim at reducing classification accuracy

Targeted Attacks
- Aim to cause misclassification of inputs with triggers only
The Grand Challenge: Poisoning Attacks in FL

- Adversaries can control one (or more) local clients and manipulate (poison) data and/or training process
- Backdoors in local models can make it to global, too

Untargeted Attacks
- Aim at reducing classification accuracy

Targeted Attacks
- Aim to cause misclassification of inputs with triggers only
Defense Approaches

Information Reduction, e.g. [1,2]
- Differential Privacy approaches, e.g., noising and clipping or gradient pruning
- Conducted on local models or aggregated global model

Robust Aggregation e.g. [3,4]
- Replace the standard aggregation algorithm
- E.g., select only one local contribution to be part of the new global model [3,4]

Detection & Filtering, e.g. [5,6]
- Detection based on one or a few metrics
- Filtering leverages clustering methods
- Conducted on local models or updates (to the global model)

[1] E. Bagdasaryan et al., How To Backdoor Federated Learning. AISTATS, 2020
[2] Naseri et al., Local and Central Differential Privacy for Robustness and Privacy in Federated Learning, NDSS 2022
Challenges of Filtering-based Defense Approaches

1. Non-IID Data (non-independent and identically distributed)
2. Detection of Multiple Backdoors
3. Adaptive Attacker
The Challenge of Non-IID Data

Example of IID data (nearly uniform distribution)

Sample Count
0 100 200 300

Labels
0 1 2 3 4 5 6 7 8 9

Prediction classes on one client (10 classes)

Example of non-IID data

Client 1
Client 2
Client n

Scenario 1
Scenario 2
Scenario 3
Scenario 4

Classical non-IID: 1 class has more labels
Arbitrary distribution of labels on clients (but similar across clients)
Considered already very challenging
2-class non-IID with disjoint labels (similar distribution across clients)
Inter-client non-iid – arbitrary distribution across and within clients

Very hard
Easy

Was not considered in related work
Visualisation of Model Updates

• Let’s imagine that the model is a simple linear function $f(x) = ax + b$, where $a$ and $b$ are model parameters.

- Malicious models differ from the global model due to the adversary’s manipulation.
- Benign models differ due non-independent and identically distributed (non-IID) data.
Challenges of Correct Clustering

- Global model from training round t-1
- Benign models at round t
- Malicious models at round t

- One backdoor & IID data
- One backdoor & non-IID data?
- Multiple backdoors?
- Multiple backdoors & non-IID data?

Benign or malicious?

Benign

Backdoored

Benign?!
Adaptive Attackers

<table>
<thead>
<tr>
<th>Adaption Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changing PDR</td>
<td>Adapt number of samples for backdoor behavior in training data</td>
</tr>
<tr>
<td>Changing PMR</td>
<td>Adapt number of malicious clients that inject the backdoor</td>
</tr>
<tr>
<td>Changing Behaviour</td>
<td>Behave benign or malicious in different training rounds</td>
</tr>
<tr>
<td>Changing Loss Function</td>
<td>Adding an additional adaptation loss to constrain weights</td>
</tr>
</tbody>
</table>

Loss function:

\[ \text{Loss} = \alpha \cdot \text{Loss}_{\text{data}} + (1 - \alpha) \cdot \text{Loss}_{\text{adaptation}} \]
Adaption by Means of Changing Loss Function

State-of-the-Art Approach

- Constrain-and-Scale method from Bagdasaryan et. al [1]
  - ONE loss for the task in the dataset $Loss_{data}$ and ...
  - ONE loss for the adaption $Loss_{adaption}$
  - both weighted by ONE scaling parameter $\alpha$
  - $\alpha$ parameter introduces adversarial dilemma between backdoor effectiveness and stealthiness

Challenges for Attackers

- Find suitable $\alpha$ (typically done manually)
- One can encounter ill-conditioning: $Loss_{data}$ and $Loss_{adaption}$ are at different scales → this will lead to a situation where only one loss is effectively optimized

\[ \text{Loss} = \alpha \cdot Loss_{data} + (1 - \alpha) \cdot Loss_{adaption} \]

[1] E. Bagdasaryan et al., How To Backdoor Federated Learning. AISTATS, 2020
Addressing Challenges of Filtering-based Defenses

- CrowdGuard [with Rieger et al., NDSS 2024]
- FreqFed [with Fereidooni et al., NDSS 2024]
- MESAS [with Krauss. ACM CCS 2023]
Addressing Challenges of Filtering-based Defenses

- CrowdGuard [with Rieger et al., NDSS 2024]
- FreqFed [with Fereidooni et al., NDSS 2024]
- MESAS [with Krauss. ACM CCS 2023]
CrowdGuard
Federated Backdoor Detection in Federated Learning

Philip Rieger*¹, Torsten Krauß*², Markus Miettinen¹, Alexandra Dmitrienko², Ahmad-Reza Sadeghi¹
* Equally contributing authors
¹TU Darmstadt, ²Uni Wuerzburg

Network and Distributed System Security Symposium (NDSS), 2024
CrowdGuard: Federated Backdoor Detection

- Assumption: > 50% of clients are benign
- Requirement: Analysis/aggregation of local models is performed within Trusted Execution Environment (TEE)

1. Server distributes all local models to clients
2. Clients use local data for validation of local models of other clients
3. Report results to Server
4. Server applies clustering and filters malicious models
Analyzing Deep Layer Client Predictions

- Repeat for every sample of every label and average results within the label
Output of Deep Layer Client Predictions

- Distance of benign and backdoored models to the global model must differ in at least some layer outputs
- >50% of clients are benign → Median must also be benign → We can identify which cluster is benign

Dashed models are benign, solid models are malicious. Colors depict main labels. Perspective from Model 0.
Reducing Dimensionality using Principal Component Analysis (PCA)

Setup:
- 10 clients (11 benign & 9 malicious) – Analysis on client 0

Values:
- Principal component 1 values

Metric:
- Cosine and Euclidian distance of the prediction to the prediction of the Global Model

Benign models are circles, malicious models are triangles. Colors depict main labels.

Malicious contributions

Benign contributions identified by median
Detection and Pruning Malicious Models

- **PCA** – Principle Component Analysis

**Pruning**

- New PCA

**Significant different**

- Majority

**Statistical Tests**

- T-Test for equal mean
- F-Test for equal variance
- D-Test for equal distribution
- 3σ rule for outlier detection
Results and Findings

Metrics:
- Cosine and Euclidian distance of local model to global model layer outputs
- PCA is effective for dimensionality reduction
- We additionally derive so-called HLBIM metric which helps to separate benign and malicious models more effectively

Effectiveness and Advantages:
- 100% True Positive Rate (TPR) and True Negative Rate (TNR) across various scenarios, including IID and non-IID data distribution (scenarios 1-3)
- Per design resilient against adaptive attackers

→ CrowdGuard is being integrated into OpenFL 1.6

Special Considerations:
- Requires usage of Trusted Execution Environments (TEEs)
- Our next works do not require any TEEs on clients!
Our Filtering-based Defenses that Address Challenges

- CrowdGuard [with Rieger et al., NDSS 2024]
- FreqFed [with Fereidooni et al., NDSS 2024]
- MESAS [with Krauss. ACM CCS 2023]
FreqFed

A Frequency Analysis-Based Approach for Mitigating Poisoning Attacks in Federated Learning

Hossein Fereidooni¹, Alessandro Pegoraro¹, Phillip Rieger¹, Alexandra Dmitrienko², Ahmad-Reza Sadeghi¹

¹TU Darmstadt, ²Uni Wuerzburg

Network and Distributed System Security Symposium (NDSS), 2024
FreqFed: A Frequency Analysis-Based Backdoor Detection in FL

Problems of previous defenses:
- Client-based detection methods require protections against privacy attacks (e.g., TEE-based execution)
- Server-side defenses are weak against adaptive attackers
- Non-IID data, especially disjoint labels (scenario 3), are difficult to address (source of false positives)

Idea:
- Transform model weights to frequency domain and perform frequency analysis

Goals:
- Support scenarios 1-3 of non-IIDness
- Prevent attackers from adapting to the defense
- Avoid reliance on TEEs
Intuition

During training, DNNs prioritize low frequencies, transitioning from low to high frequencies when approximating target functions [1].

Most energy in model weights is in low-frequency DCT* components [2].

Two Observations:

1. We inspire and emphasize the low-frequency DCT spectrum because it reveals weight energy distribution across frequencies.
2. Backdoors typically cause an energy shift in the low-frequency components of the DCT. The energy shift, while subtle in the time domain, becomes more noticeable in the frequency domain.
3. An adaptive attacker operates in time domain and cannot adapt easily in frequency domain.

*DCT Discrete Cosine Transform

FreqFed Approach

• Assumption: > 50% of clients are benign
Results and Findings

Metrics:
- low-frequency components of the DCT

Effectiveness
- 100% True Positive Rate (TPR) and True Negative Rate (TNR) across various scenarios, including IID and non-IID data distribution (scenarios 1-3)

Advantages:
- Resilient against adaptive attackers (empirically shown)
- No reliance on TEEs
Our Filtering-based Defenses that Address Challenges

- CrowdGuard
  [with Rieger at al., NDSS 2024]

- FreqFed
  [with Fereidooni et al., NDSS 2024]

- MESAS
  [with Krauss. ACM CCS 2023]
MESAS

Poisoning Defense for Federated Learning Resilient against Adaptive Attackers

Torsten Krauss and Alexandra Dmitrienko

Uni Wuerzburg

ACM Conference on Computer and Communications Security (CCS), 2023
MESAS: Metric – Cascades for Poisoning Detection

Goals:
- Support arbitrary non-IID client datasets (including scenario 4)
- Prevent attackers from adapting to the defense without relying on TEEs

Idea:
- Use many metrics for detection of poisoned models at the same time
- Intuition: For an adaptive attacker, it should be harder (if at all possible?) to adapt to many metrics

The most challenging non-IID scenario:
Arbitrary distribution between and across clients

Classical Adaptive Adversary

\[ \text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{adaptation}} \]
MESAS Approach

Approach:

- Detection and pruning based on six well-chosen metrics
- Force the attacker into a heavy multi-objective optimization problem
  - Hardening the adversarial dilemma between backdoor effectiveness and stealthiness
MESAS Approach - Metrics

**COS & EUCL:**
- Cosine & Euclidean distance between Global and Local Models

**COUNT:**
- Reason: Same COS ($\beta$) for different models possible
- COUNT counts a number of parameters that are increased
MESAS Approach - Metrics

VAR:
- COS, EUCL, and COUNT can look benign, but still a backdoor can be embedded
- Adversary could increase the variance of updates

\[
\text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{adaptation}}
\]

\[
\text{Loss}_{\text{COS}} + \text{Loss}_{\text{EUCL}} + \ldots + \text{Loss}_{\text{MIN}}
\]

![Diagram showing local and global models, feature extractor, and adaptive adversary with metrics like COS, EUCL, COUNT, and VAR (max and min)].
MESAS Approach - Metrics

MIN & MAX:
- Variances in general are not heavily influenced by extreme outliers
- An adversary could embed a backdoor into outliers

\[ \text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{adaptation}} + \text{Loss}_{\text{COS}} + \text{Loss}_{\text{EUCL}} + \ldots + \text{Loss}_{\text{MIN}} \]
MESAS Approach

Approach – Step 1:
- Extract six metrics

Approach – Step 2:
- Iterative pruning loop leveraging statistical tests and clustering to detect poisoned models

\[ \text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{adaptation}} \]

\[ \text{Loss}_{\text{COS}} + \text{Loss}_{\text{EUCL}} + \ldots + \text{Loss}_{\text{MIN}} \]
MESAS Results

Evaluation:

- Metrics have mutual effects during adaptation
- We demonstrate empirically that an attacker cannot adapt to all of them at the same time
- It works even for the most challenging non-IID scenario with arbitrary distribution across clients!

\[
Loss = Loss_{data} + Loss_{adaptation}
\]

\[
Loss_{COS} + Loss_{EUCL} + \ldots + Loss_{MIN}
\]
# CrowdGuard vs. FreqFed vs. MESAS

<table>
<thead>
<tr>
<th>What is analyzed?</th>
<th>CrowdGuard</th>
<th>FreqFed</th>
<th>MESAS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prediction layer outputs</td>
<td>Local model updates</td>
<td>Local models</td>
</tr>
<tr>
<td>Where the analysis is performed?</td>
<td>Clients</td>
<td>Server</td>
<td>Server</td>
</tr>
<tr>
<td>Utilized metrics</td>
<td>Cosine &amp; Euclidian distances between global and local models</td>
<td>Low frequency components in frequency spectrum</td>
<td>Six metrics: Cosine &amp; Euclidian distances, COUNT, Variance, Outliers (MIN &amp; MAX)</td>
</tr>
<tr>
<td>Resilience against adaptive attacker</td>
<td>Resilient per design</td>
<td>Demonstrated empirically</td>
<td>Demonstrated empirically</td>
</tr>
<tr>
<td>Non-IIDness</td>
<td>Scenarios 1-3</td>
<td>Scenarios 1-3</td>
<td>Scenarios 1-4</td>
</tr>
<tr>
<td>Additional requirements</td>
<td>TEE on clients</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
More on Adaptive Attacks and Related Challenges

- Constrain-and-Scale method from Bagdasaryan et al [1] requires manual fine-tuning
  - Can be already challenging with one $Loss_{adaption}$
  - If an attacker wants to bypass several detection metrics, they need to consider more complex $Loss_{adaption}$ consisting of several components

Wish-list of the Attacker

- Adaption to multiple losses simultaneously
- Individual weights for all adaption losses
- No manual configuration of $\mu_j$ or $\alpha$ while getting a good adaption

→ Can the process of adaption be automated?

[1] E. Bagdasaryan et al., How To Backdoor Federated Learning. AISTATS, 2020
AutoAdapt
Automatic Adversarial Adaption for Stealthy Poisoning Attacks in Federated Learning

Torsten Krauss, Jan König, Alexandra Dmitrienko, and Christian Kanzow

Network and Distributed Systems Security Symposium (NDSS), 2024
Visualization of Poisoned Models and Detection Metrics

Example with one detection metric value

Exemplary visualization of a model with 2 parameters

\[ \text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{adaptation}} \]
AutoAdapt: Automatic Adversarial Adaption

\[ \text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{adaptation}} \]
AutoAdapt: Automatic Adversarial Adaptation

\[
\text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{adaption}}
\]

\[
\text{Loss} = \alpha \cdot \text{Loss}_{\text{data}} + (1 - \alpha) \cdot \text{Loss}_{\text{adaption}}
\]

\[
\text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{AutoAdapt}}
\]

\[
\text{Loss}_{\text{AutoAdapt}} = \frac{1}{2\alpha_{\text{AutoAdapt}}} \sum_{j=1}^{m} \left( \max(0, \mu_j + \alpha_{\text{AutoAdapt}} \text{Loss}_j) \right)^2 - \mu_j^2
\]

\[
\mu_j = \begin{cases} 
\mu_j + \alpha_{\text{AutoAdapt}} \text{Loss}_j, & \text{if } \text{Loss}_j \geq 0 \\
0, & \text{if } \text{Loss}_j < 0
\end{cases}
\]

Solution

- Replace \( \alpha \) with Augmented Lagrangian (AL)* Method
- Extend AL for multiple range constraints (if we want to detect in several metrics)
- No manual hyperparameters → \( \alpha_{\text{AutoAdapt}} \) is insensitive
- Automatic switching off of the \( \text{Loss}_{\text{AutoAdapt}} \) for constraints that are already fulfilled

*Augmented Lagrangian methods are a certain class of algorithms for solving constrained optimization problems
AutoAdapt: Automatic Adversarial Adaption

\[ \text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{AutoAdapt}} \]

Workflow

1. Benign Training
2. Benign Models
3. Benign Models
4. Unconstrained
5. Suspicious Models
6. Aggregation Server

Valid Range Inequality Constraints

Cost:
\[ \text{COS}_{\min} \leq \text{COS}^* \leq \text{COS}_{\max} \]
AutoAdapt: Automatic Adversarial Adaption

\[ \text{Loss} = \text{Loss}_{\text{data}} + \text{Loss}_{\text{AutoAdapt}} \]

Results

- Successfull adaption to multiple range constraints simultaneously
- Adaption on a model-wise and layer-wise level
- Showcased circumvention of five state-of-the-art defenses
- Quick adaption (mostly within 1-3 training epochs)

→ We propose to use AutoAdapt as a new baseline for evaluation of new FL poisoning defenses
Conclusion

➢ Federated Learning helps solving high data demand vs. privacy dilemma

➢ Similar to centralized ML, FL is also prone to untargeted and targeted poisoning attacks

➢ An arm raise between attacks and defenses is going on and will continue
“If we let it out, there’s an 85% chance it would cure cancer. But there’s also a 0.01% chance it takes over the world!”

https://www.evilaicartoons.com/archive