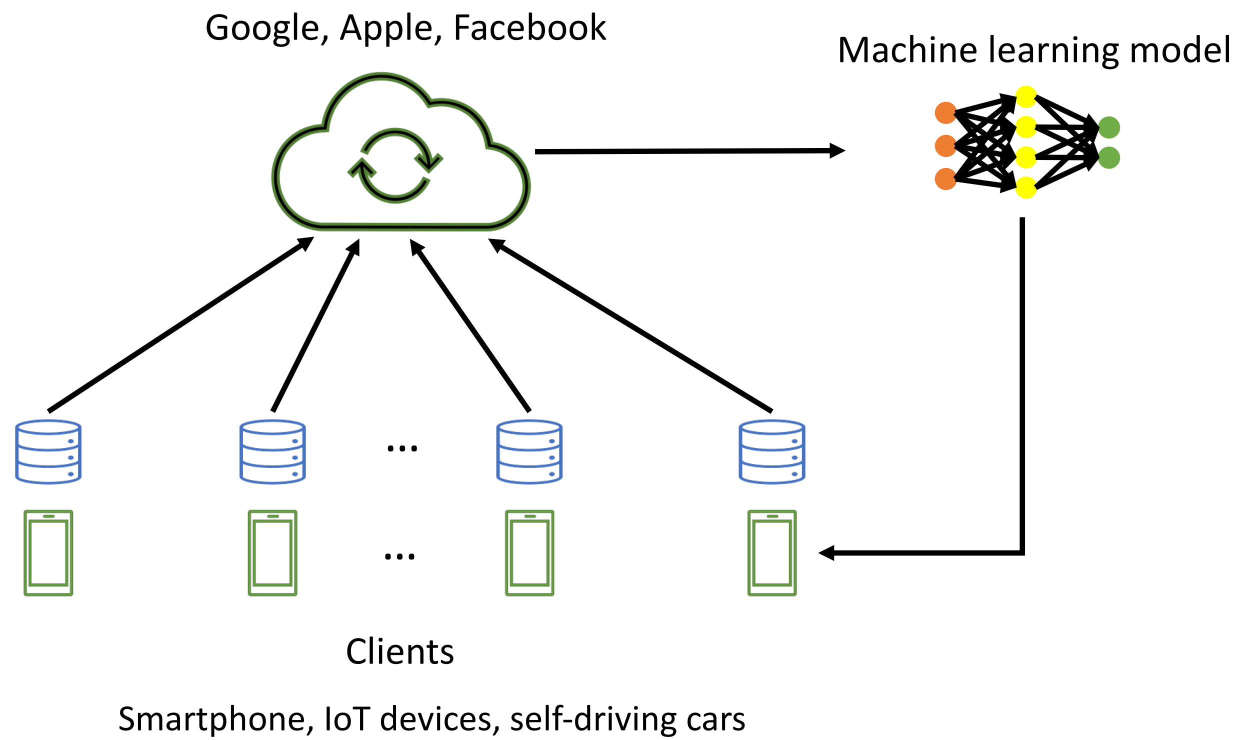
Secure Federated Learning

Neil Gong
Department of Electrical and Computer Engineering
Department of Computer Science (secondary appointment)
Duke University

This talk is available on YouTube: https://www.youtube.com/watch?v=LP4uqW18yA0

Conventional Paradigm: Centralized Learning



Challenges of Centralized Learning

- Server data breaches

- High communications cost
  - Intolerable for resource-constrained clients
    - Smartphone
    - IoT

Federated Learning

- Data stay locally on clients
- Clients train models locally
- Clients send models or updates to server
- Real-world deployment

Gboard

Artificial intelligence / Machine learning

How Apple personalizes Siri without hoovering up your data

The tech giant is using privacy-preserving machine learning to improve its voice assistant while keeping your data on your phone.

by **Karen Hao**

December 11, 2019

This Talk

What are the security issues of federated learning

How to build secure federated learning

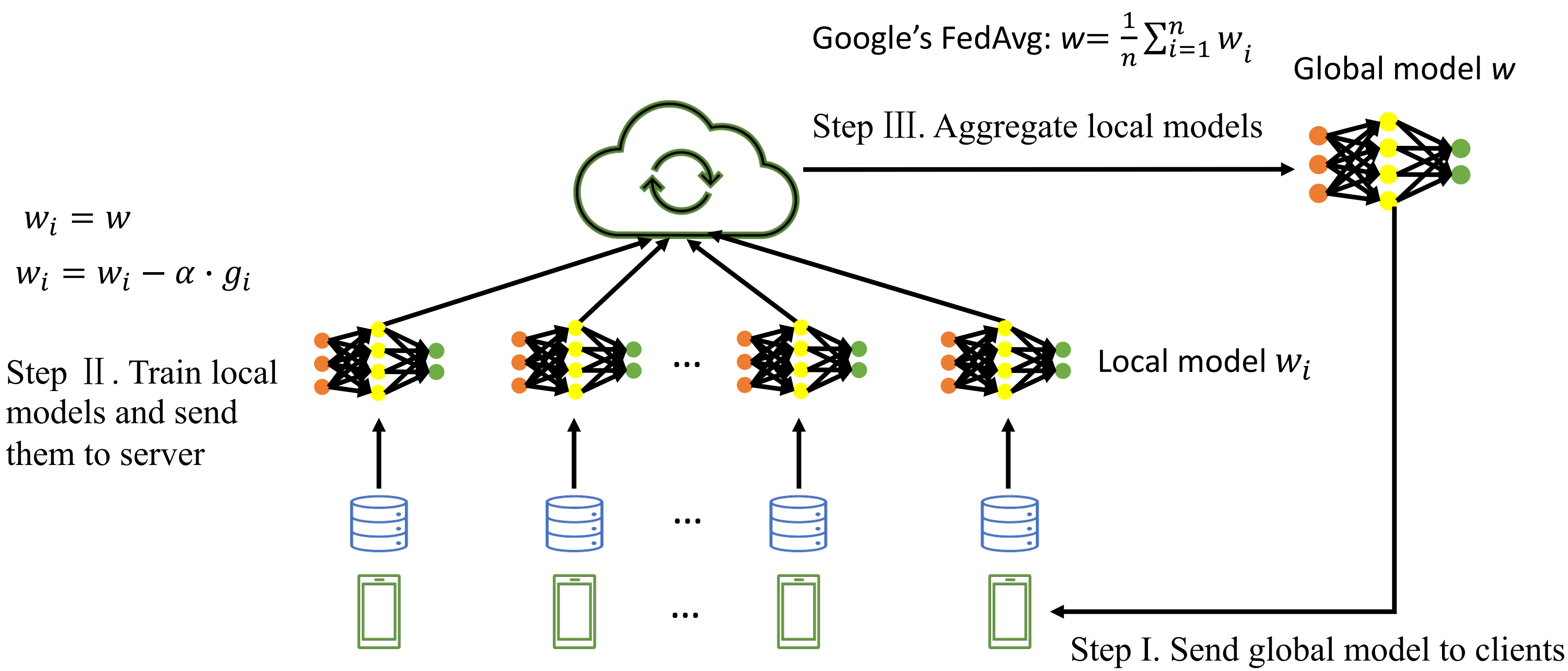
Road Map

- Part I: Local model poisoning attacks to federated learning
- Part II: Secure federated learning via trust bootstrapping
- Part III: Provably secure federated learning

Road Map

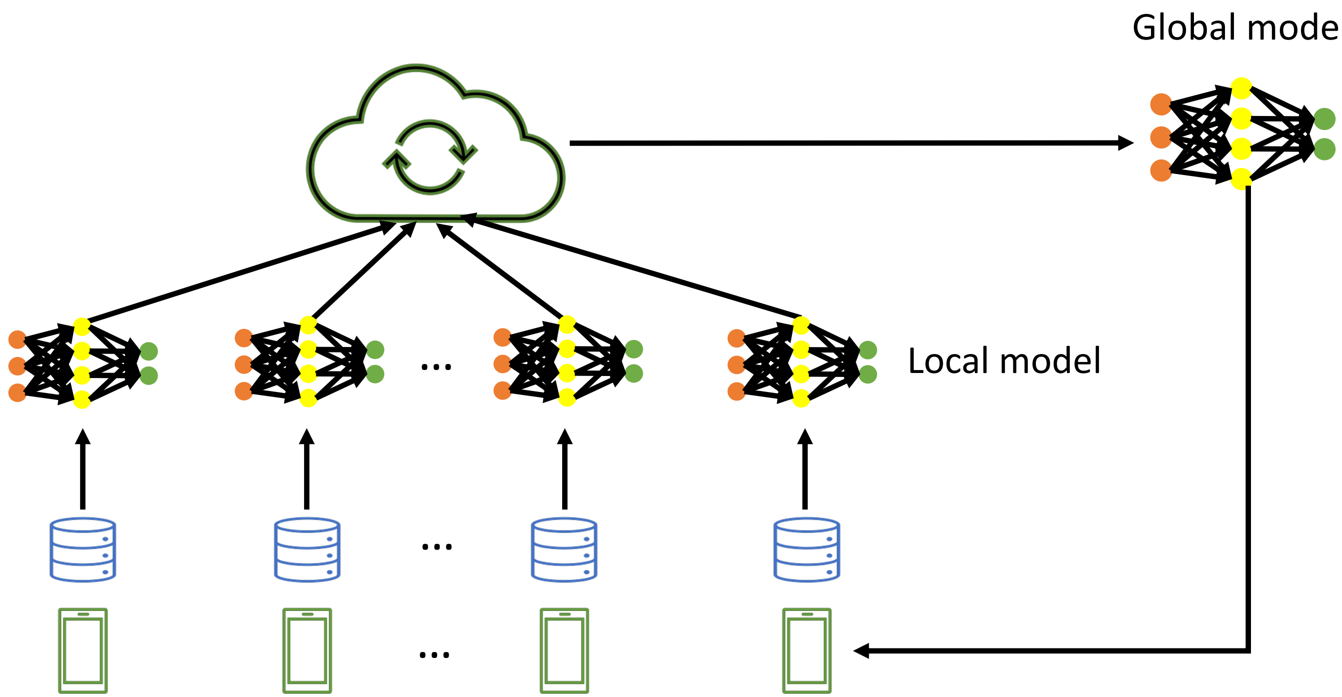
- **Part I: Local model poisoning attacks to federated learning**
- Part II: Secure federated learning via trust bootstrapping
- Part III: Provably secure federated learning

Federated Learning Background

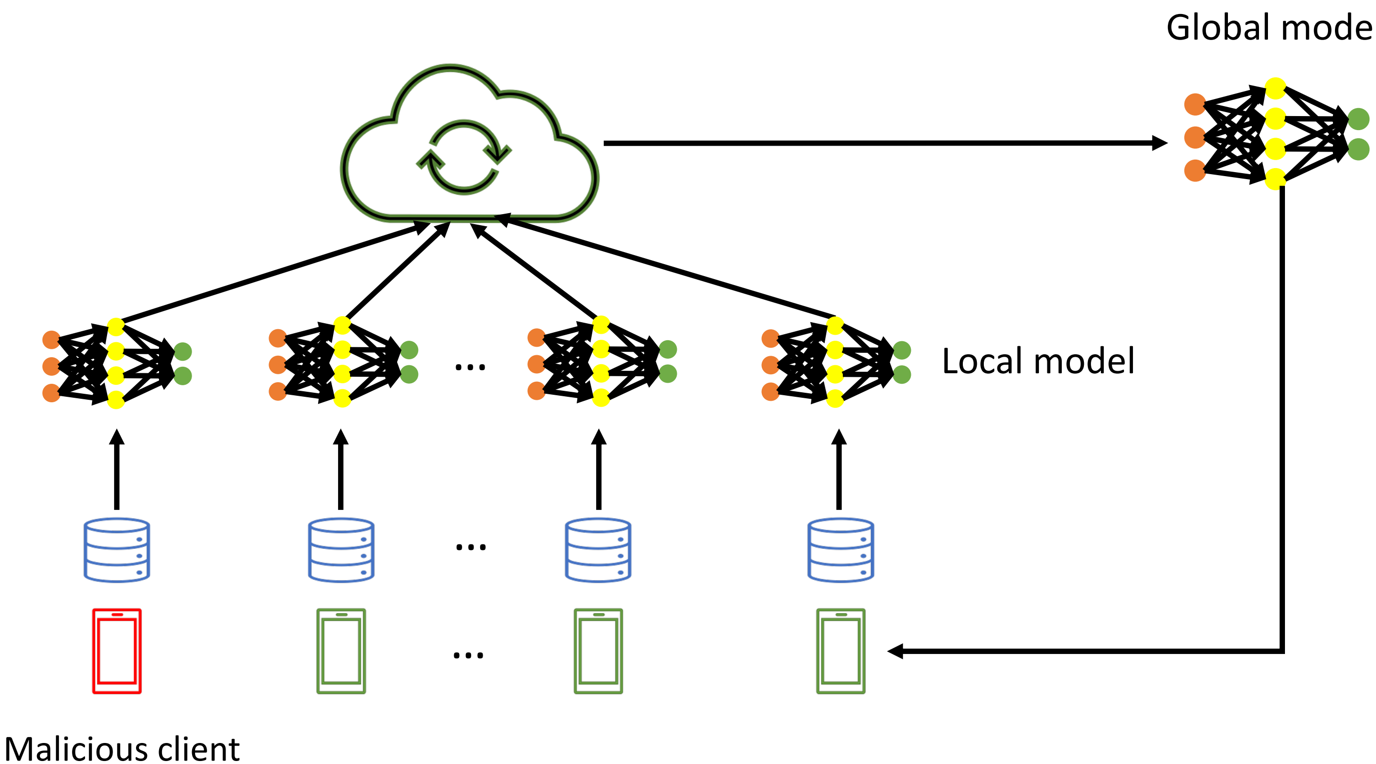


Equivalent to send local model updates $w_i - w$ to server

Federated Learning is Vulnerable to Poisoning Attacks

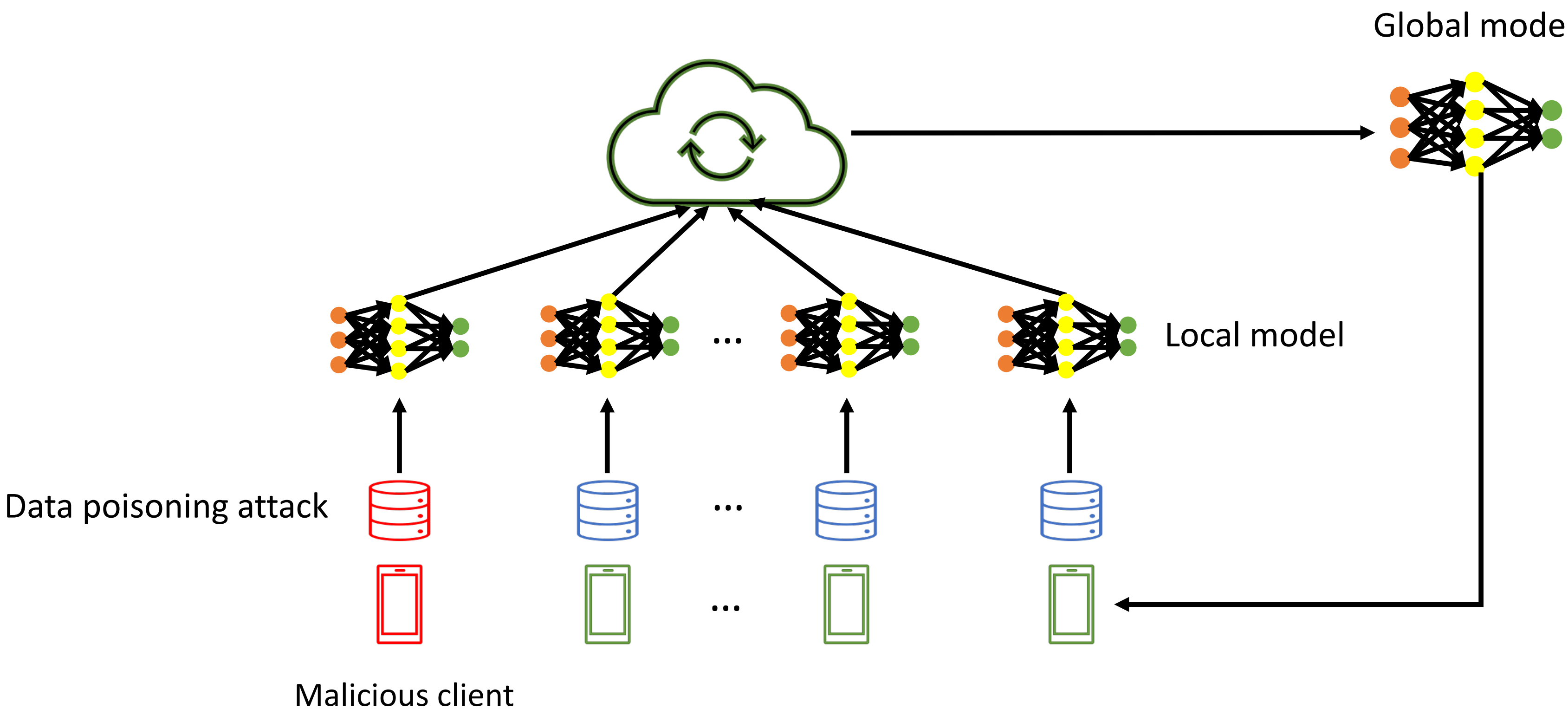


Federated Learning is Vulnerable to Poisoning Attacks



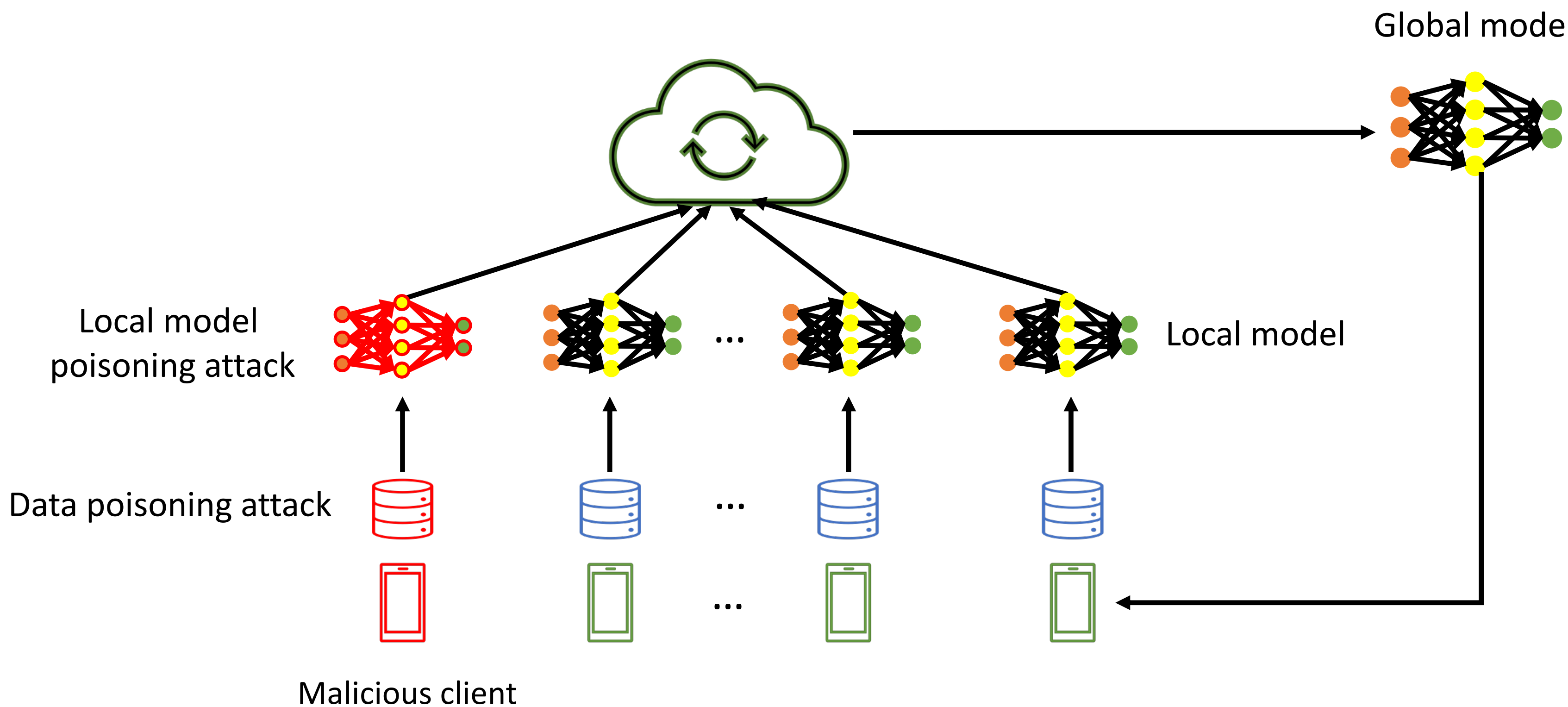
Fake or compromised genuine clients
Fake clients can be many

Federated Learning is Vulnerable to Poisoning Attacks



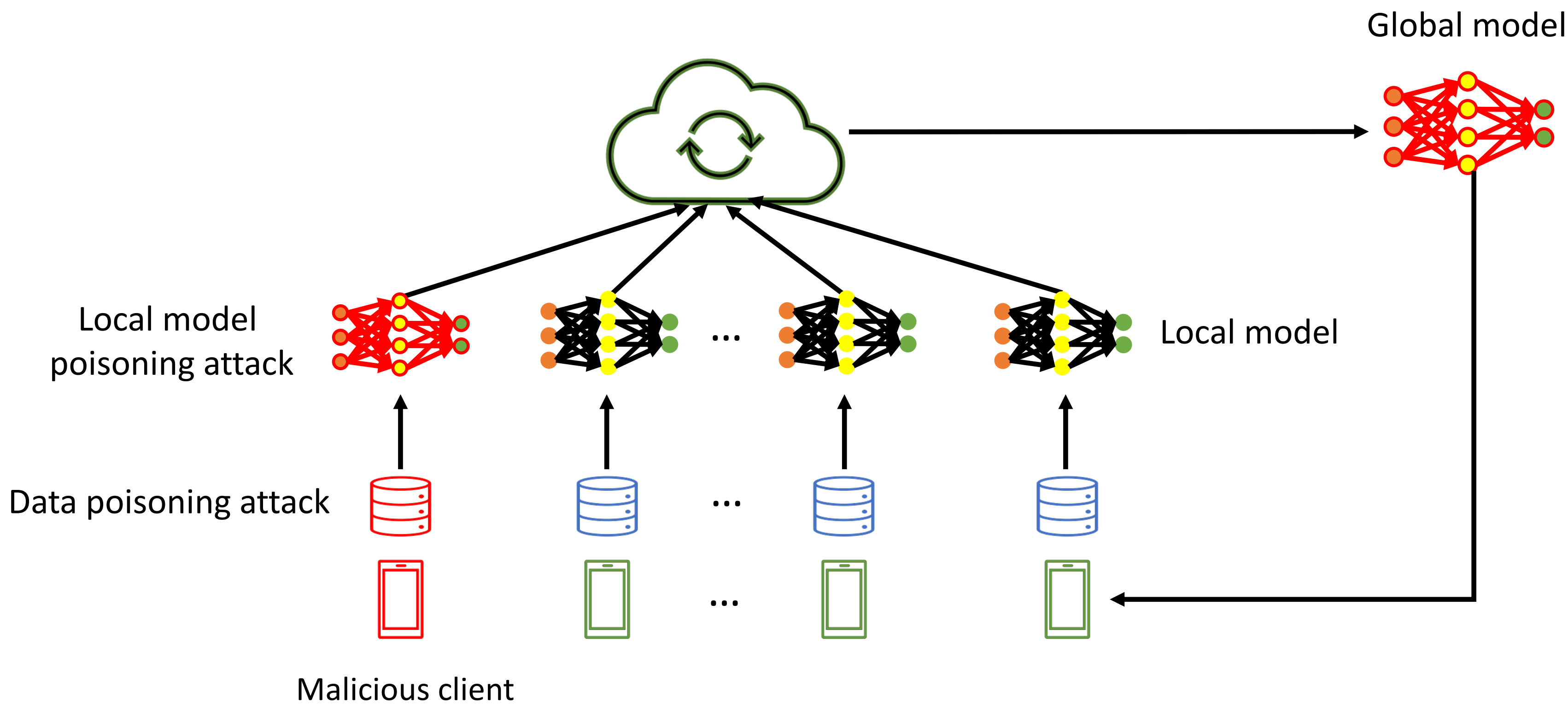
Fake or compromised genuine clients
Fake clients can be many

Federated Learning is Vulnerable to Poisoning Attacks



Fake or compromised genuine clients
Fake clients can be many

Federated Learning is Vulnerable to Poisoning Attacks



Fake or compromised genuine clients
Fake clients can be many

Byzantine-robust Federated Learning as Defense

- Byzantine-robust aggregation rule
  - Krum
  - Trimmed mean
  - Median

- Key idea
  - Remove “outlier” local models

- Theoretical guarantee
  - Various assumptions
    - IID data, smooth loss function, etc.
  - Bound change of global model parameters caused by malicious clients

An Example: Median

$$
\begin{array}{ll}
\text{Client 1} & \multirow{4}{*}{$\begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{nm} \end{pmatrix}$} \\
\text{Client 2} & \\
\vdots & \\
\text{Client n} & \\
\text{Server} & \quad w_1 \quad w_2 \quad \cdots \quad w_m
\end{array}
$$

Our Work

Byzantine-robust federated learning is vulnerable to local model poisoning attacks

Increase testing error rate of global model

Minghong Fang, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. "Local Model Poisoning Attacks to Byzantine-Robust Federated Learning". In *USENIX Security Symposium*, 2020

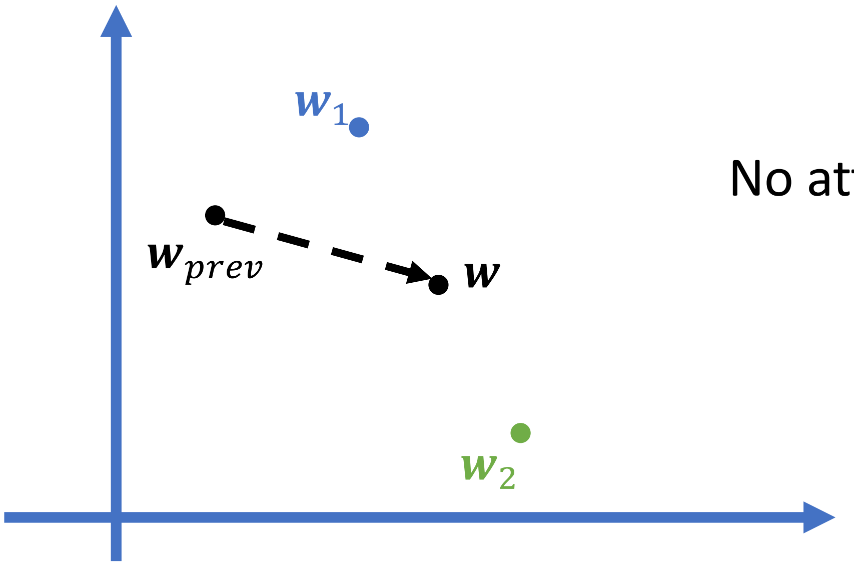
Threat Model

- Attacker's goal
  - High testing error rate

- Attacker's capability:
  - Access to malicious clients
    - Fake clients
    - Compromised genuine clients
  - Send arbitrary local models

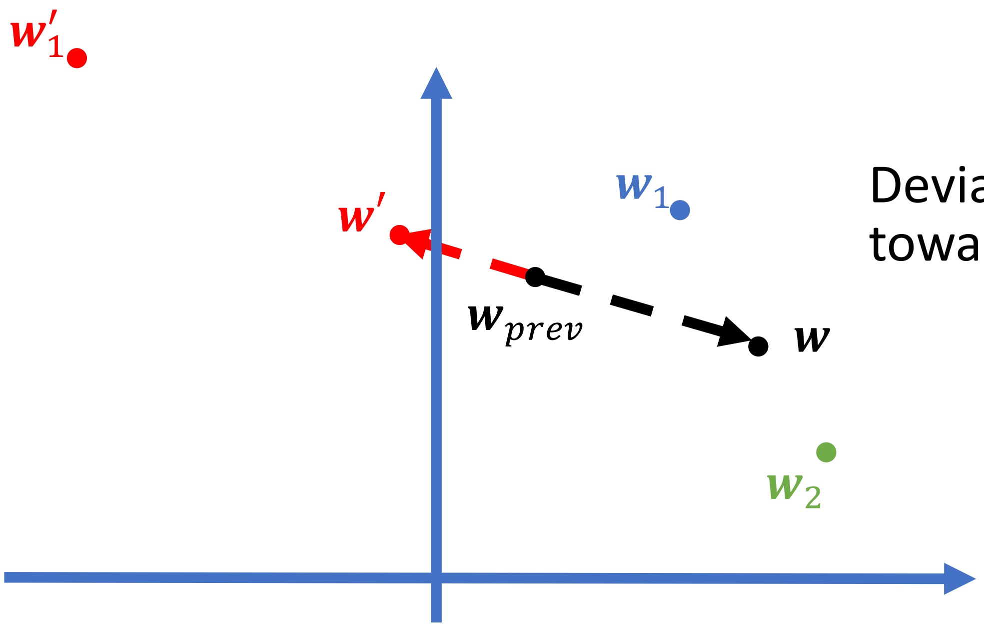
- Attacker's knowledge:
  - Full vs. Partial knowledge
    - Data on all vs. malicious clients
  - Aggregation rule
    - Yes or no

Our Idea



No attack: global model changes along some direction

Our Idea



Deviate global model the most towards inverse of the direction

Formulate Optimization Problem

Global model before attack

Update direction

Global model after attack

Poisoned local models on malicious clients

$$\max_{\boldsymbol{w}'_1,\dots,\boldsymbol{w}'_c} \boldsymbol{s}^T(\boldsymbol{w} - \boldsymbol{w}')$$

Maximize deviation of global model

Subject to $\boldsymbol{w} = \mathcal{A}(\boldsymbol{w}_1, \dots, \boldsymbol{w}_c, \boldsymbol{w}_{c+1}, \dots, \boldsymbol{w}_{\boldsymbol{n}})$

Global model aggregation before attack

$$\boldsymbol{w}' = \mathcal{A}(\boldsymbol{w}'_1, \dots, \boldsymbol{w}'_c, \boldsymbol{w}_{c+1}, \dots, \boldsymbol{w}_{\boldsymbol{n}})$$

Global model aggregation after attack

Used in all or multiple iterations

Applicable to **any** aggregation rule

Solving the Optimization Problem

- Full knowledge
  - $\boldsymbol{w}_1, \dots, \boldsymbol{w}_c, \boldsymbol{w}_{c+1}, \dots, \boldsymbol{w}_{\boldsymbol{n}}$ are known
  - Solve the optimization problem using them
- Partial knowledge
  - Only $\boldsymbol{w}_1, \dots, \boldsymbol{w}_c$ are known
  - Use them to estimate $\boldsymbol{w}$
- Unknown aggregation rule
  - Attacker assumes one

Experimental Setup

- 100 clients
  - 20% malicious
- Datasets:
  - MNIST
  - Fashion-MNIST
  - CH-MNIST
  - Breast Cancer Wisconsin (Diagnostic)
- Non-IID data on clients
  - Non-IID: not Independently and Identically Distributed

Experimental Setup

- 100 clients
  - 20% malicious
- Datasets:
  - **MNIST**
  - Fashion-MNIST
  - CH-MNIST
  - Breast Cancer Wisconsin (Diagnostic)
- Non-IID data on clients
  - Non-IID: not Independently and Identically Distributed

Our Attack is Effective

	NoAttack	Gaussian	LabelFlip	Partial	Full
Krum	0.11	0.10	0.10	0.75	0.77
Trimmed Mean	0.06	0.07	0.07	0.14	0.23
Median	0.06	0.06	0.16	0.28	0.32

Byzantine-robust methods

Our Attack is Effective

No attack

	NoAttack	Gaussian	LabelFlip	Partial	Full
Krum	0.11	0.10	0.10	0.75	0.77
Trimmed Mean	0.06	0.07	0.07	0.14	0.23
Median	0.06	0.06	0.16	0.28	0.32

Our Attack is Effective

Add Gaussian noise to local models

	NoAttack	Gaussian	LabelFlip	Partial	Full
Krum	0.11	0.10	0.10	0.75	0.77
Trimmed Mean	0.06	0.07	0.07	0.14	0.23
Median	0.06	0.06	0.16	0.28	0.32

Our Attack is Effective

Flip labels of local training data

	NoAttack	Gaussian	LabelFlip	Partial	Full
Krum	0.11	0.10	0.10	0.75	0.77
Trimmed Mean	0.06	0.07	0.07	0.14	0.23
Median	0.06	0.06	0.16	0.28	0.32

Our Attack is Effective

Our attack, partial knowledge

	NoAttack	Gaussian	LabelFlip	Partial	Full
Krum	0.11	0.10	0.10	0.75	0.77
Trimmed Mean	0.06	0.07	0.07	0.14	0.23
Median	0.06	0.06	0.16	0.28	0.32

Our Attack is Effective

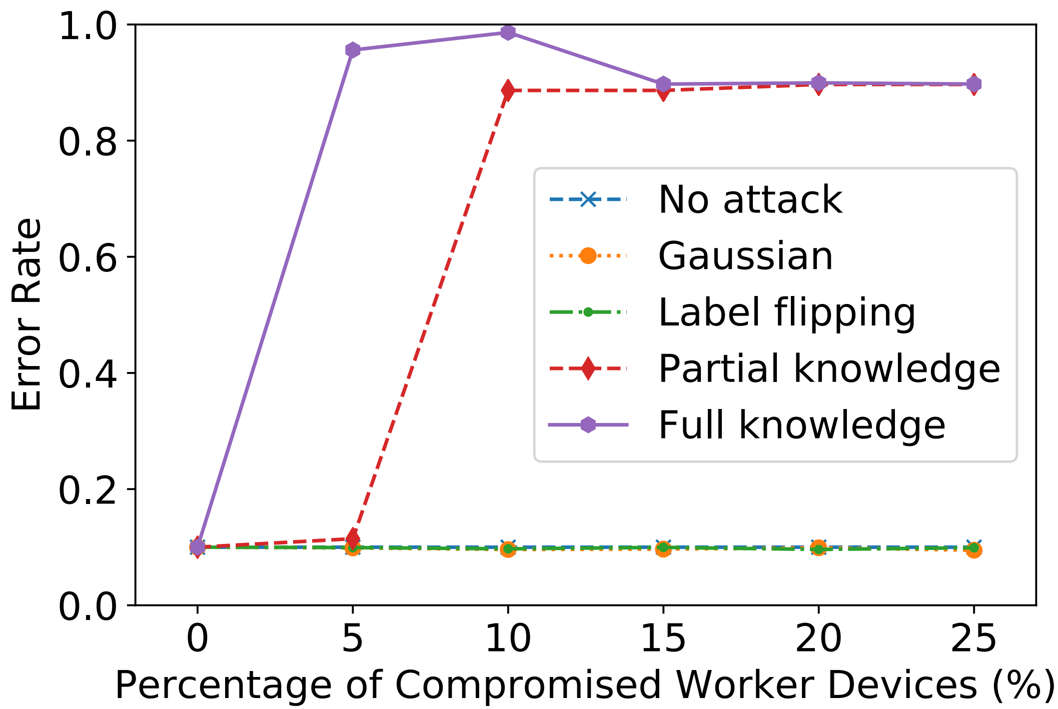
Our attack, full knowledge

	NoAttack	Gaussian	LabelFlip	Partial	Full
Krum	0.11	0.10	0.10	0.75	0.77
Trimmed Mean	0.06	0.07	0.07	0.14	0.23
Median	0.06	0.06	0.16	0.28	0.32

Our attacks can effectively increase testing error rates

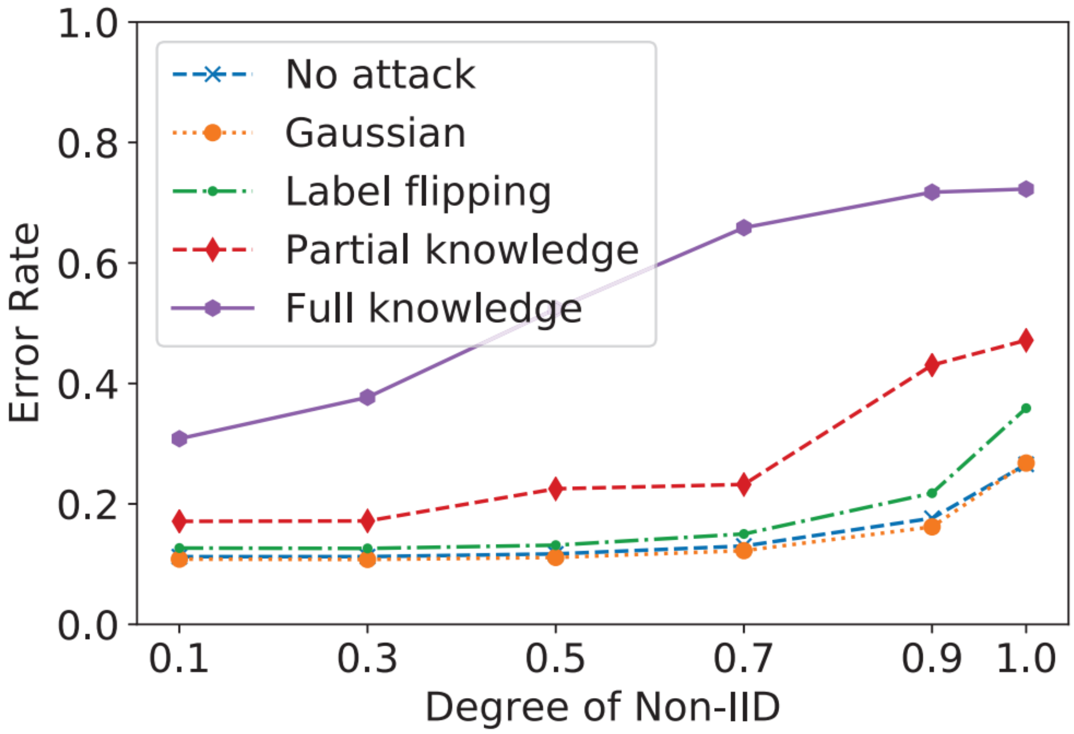
Impact of #Malicious Clients

Krum



Our attacks are more effective with more malicious clients

Impact of Degree of Non-IID



Our attacks are more effective when clients' data are more Non-IID

Our Attacks Transfer between Aggregation Rules

	Krum	Trimmed mean	Median
No attack	0.14	0.12	0.13
Krum attack	0.70	0.15	0.18
Trimmed mean attack	0.14	0.25	0.20

Comparing with Data Poisoning Attacks

No attack

State-of-the-art data poisoning attack

Our attack

	NoAttack	DataPoisoning	Partial	Full
Krum	0.23	0.24	0.85	0.89
Trimmed Mean	0.12	0.12	0.27	0.32
Median	0.13	0.13	0.19	0.21

Data poisoning attacks are ineffective for Byzantine-robust methods

Our attacks are effective

Summary

- Proposed a general framework to attack federated learning
- Existing Byzantine-robust federated learning is vulnerable to local model poisoning attacks

Road Map

- Part I: Local model poisoning attacks to federated learning
- **Part II: Secure federated learning via trust bootstrapping**
- Part III: Provably secure federated learning

Root Cause of Insecurity

No root trust

Every client could be malicious

Our FLTrust: Bootstrapping Trust

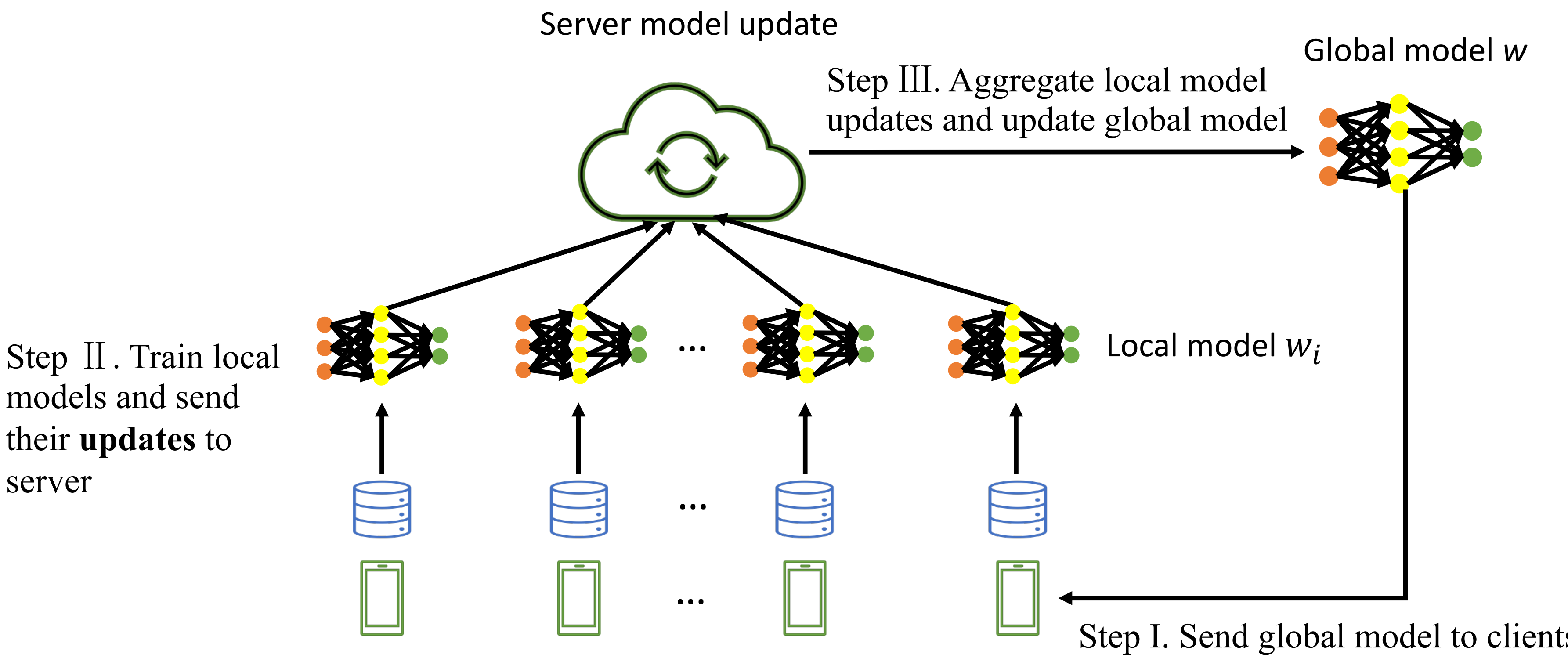
- Server collects a small, clean training dataset

- Server maintains a *server model*
  - Like how a client maintains a local model

- Use server model to bootstrap trust
  - Assign trust scores to clients

Xiaoyu Cao, Minghong Fang, Jia Liu, and Neil Zhenqiang Gong. "FLTrust: Byzantine-robust Federated Learning via Trust Bootstrapping". In *ISOC Network and Distributed System Security Symposium (NDSS),* 2021.

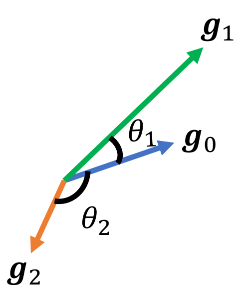
Revisiting Federated Learning Background



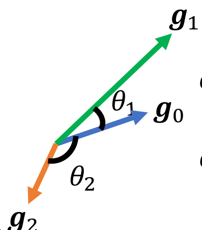
Our Aggregation Rule

Local model updates $\boldsymbol{g}_1, \boldsymbol{g}_2$

server model update $\boldsymbol{g}_0$

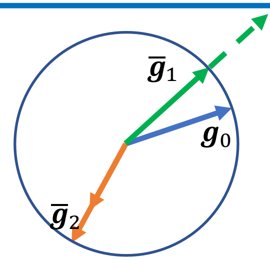


ReLU-clipped cosine similarity based trust score (TS)



$$c_1 = \cos\theta_1 = \frac{\langle \boldsymbol{g}_1, \boldsymbol{g}_0 \rangle}{\|\boldsymbol{g}_1\| \cdot \|\boldsymbol{g}_0\|} \qquad TS_1 = \mathrm{ReLU}(c_1) = c_1$$

$$c_2 = \cos\theta_2 = \frac{\langle \boldsymbol{g}_2, \boldsymbol{g}_0 \rangle}{\|\boldsymbol{g}_2\| \cdot \|\boldsymbol{g}_0\|} \qquad TS_2 = \mathrm{ReLU}(c_2) = 0$$



$$\bar{\boldsymbol{g}}_1 = \frac{\|\boldsymbol{g}_0\|}{\|\boldsymbol{g}_1\|} \cdot \boldsymbol{g}_1$$

$$\bar{\boldsymbol{g}}_2 = \frac{\|\boldsymbol{g}_0\|}{\|\boldsymbol{g}_2\|} \cdot \boldsymbol{g}_2$$

Normalizing the magnitudes of local model updates

Aggregation

$$\boldsymbol{g} = \frac{1}{TS_1 + TS_2}(TS_1 \cdot \bar{\boldsymbol{g}}_1 + TS_2 \cdot \bar{\boldsymbol{g}}_2)$$

$$\boldsymbol{w} = \boldsymbol{w} - \beta \cdot \boldsymbol{g}$$

Theoretical Analysis

Under certain assumptions, for an arbitrary number of malicious clients, the difference between the global model learnt by FLTrust and the optimal global model under no attacks is bounded

Empirical Results

MNIST

100 clients, 20 malicious

Server's training dataset: 100 examples sampled from MNIST

State-of-the-art method in non-adversarial settings

	FedAvg	Krum	Trim-mean	Median	FLTrust
No attack	0.04	0.10	0.06	0.06	0.05
Label flipping attack	0.06	0.10	0.06	0.06	0.05
Krum attack	0.10	0.91	0.14	0.15	0.05
Trim attack	0.28	0.10	0.23	0.43	0.06

Our FLTrust is robust against poisoning attacks

Adaptive Attack

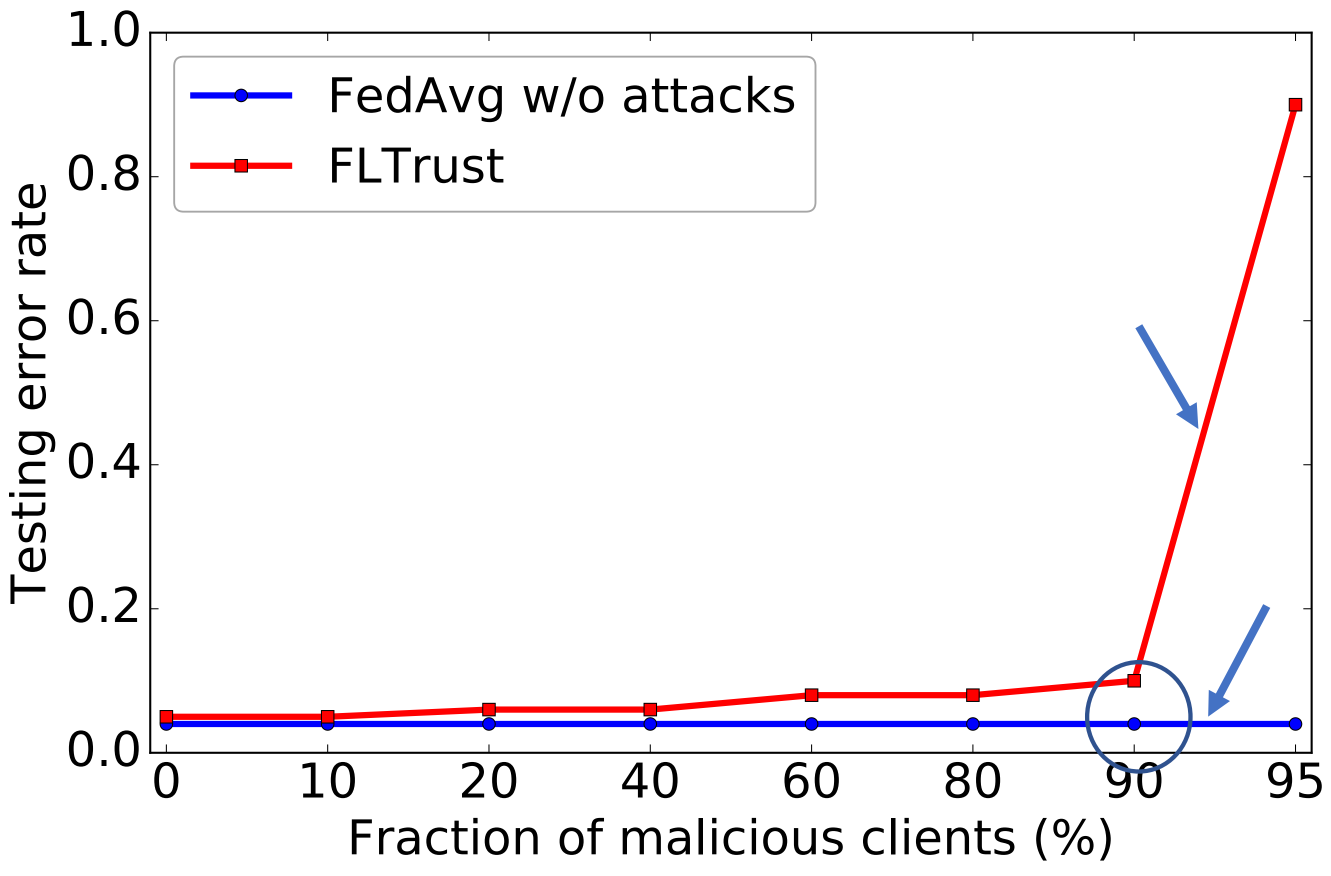
$$\max_{\boldsymbol{w}'_1,\ldots,\boldsymbol{w}'_c} \boldsymbol{s}^T(\boldsymbol{w} - \boldsymbol{w}')$$

Subject to $\quad \boldsymbol{w} = \mathcal{A}(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_c, \boldsymbol{w}_{c+1}, \ldots, \boldsymbol{w}_{\boldsymbol{n}})$

$$\boldsymbol{w}' = \mathcal{A}(\boldsymbol{w}'_1, \ldots, \boldsymbol{w}'_c, \boldsymbol{w}_{c+1}, \ldots, \boldsymbol{w}_{\boldsymbol{n}})$$

Applicable to **any** aggregation rule

Our FLTrust is Robust against Adaptive Attack



Summary

- The server can enhance security of federated learning via collecting a small, clean training dataset to bootstrap trust

Road Map

- Part I: Local model poisoning attacks to federated learning
- Part II: Secure federated learning via trust bootstrapping
- **Part III: Provably secure federated learning**

Limitations of Byzantine-robust Federated Learning

- Bound change in global model parameters caused by malicious clients
  - Under assumptions
    - IID data on clients
    - Smooth loss function
    - …

- Limitations
  - Assumptions do not hold
  - Not bound testing error rate or accuracy

Our Provably Secure Federated Learning

- Guarantee a lower bound of testing accuracy
- Only assumption
  - Bounded #malicious clients

Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. “Provably Secure Federated Learning against Malicious Clients”. In AAAI, 2021.

Defining Provable Security

Label predicted for x when the global model is trained on C

$h(C', x) = h(C, x)$ for any C', #malicious clients $\leq$ m*

A set of benign clients

Testing input

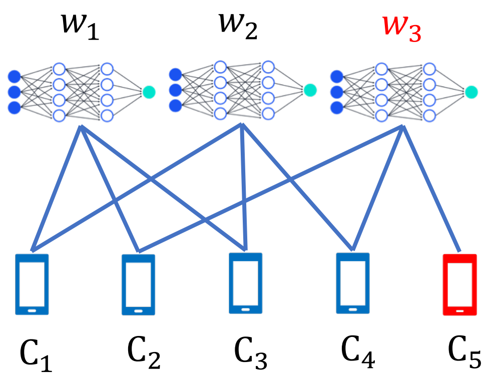
A federated learning algorithm is provably secure if its predicted label for a testing input is not affected by a bounded number of malicious clients

m*: certified security level for x

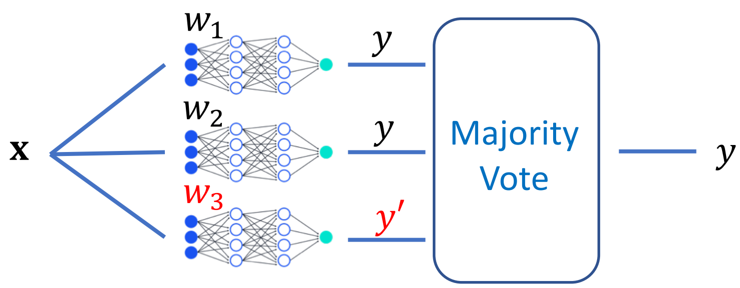
Our Ensemble Federated Learning: the First Provably Secure Method

- Training
  - n clients
  - Select k clients randomly and train a global model
    - Use any federated learning method, e.g., FedAvg
  - Repeat to train N global models
- Testing
  - Majority vote of the N global models to predict label of x

Provable Security: Intuition



Training Phase



Testing Phase

Provable Security

Given C and x, we can derive the certified security level m* for x

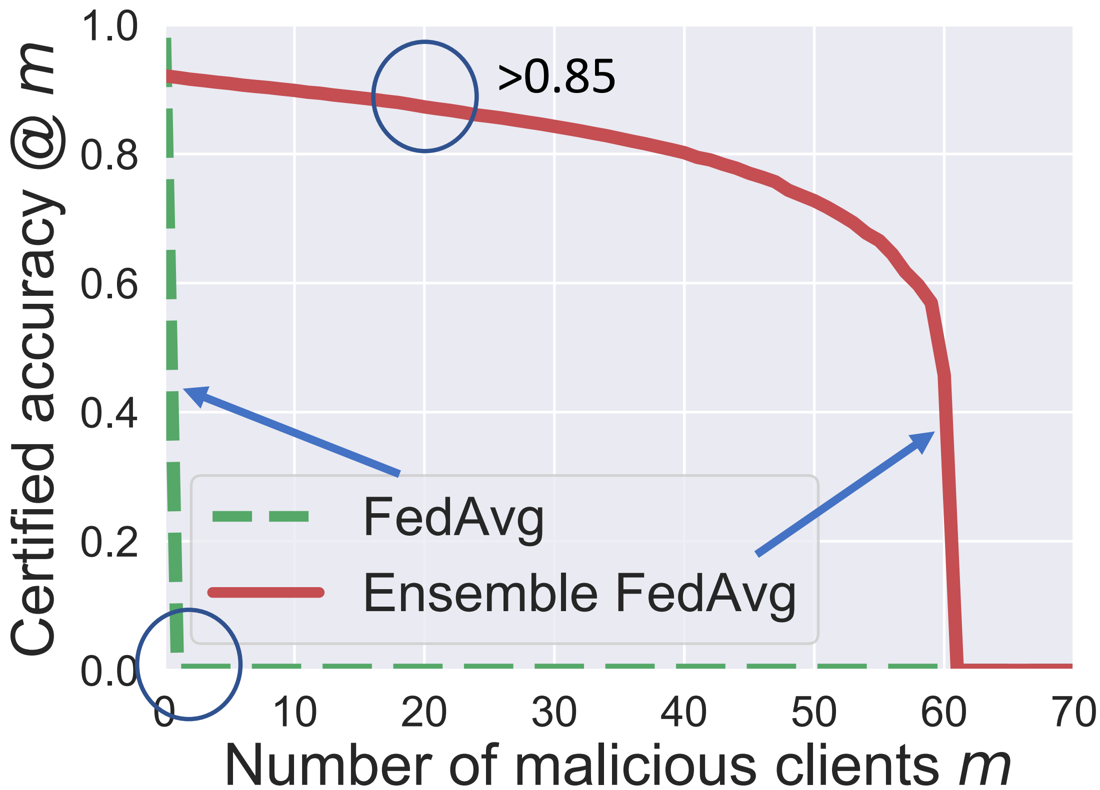
Our derived certified security level is tight

Evaluation Metric: Certified Accuracy @ m

- Fraction of testing inputs whose
  - Labels are correctly predicted
  - Certified security levels are at least m

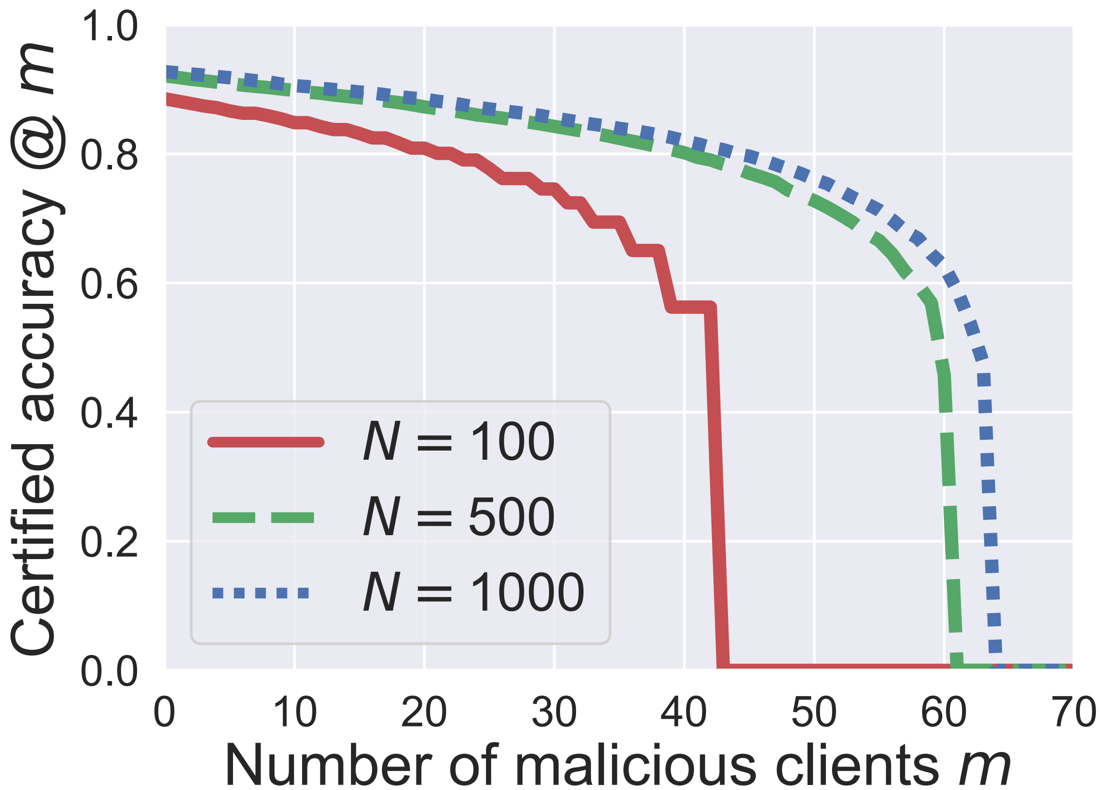
- A lower bound of testing accuracy
  - #malicious clients $\leq$ m
  - No matter what attacks are used!

FedAvg vs. Ensemble FedAvg



MNIST dataset, 1,000 clients

Impact of Number of Global Models *N*



A moderate number of global models are enough

Summary

- Ensemble federated learning is provably secure against bounded number of malicious clients
- Achieve certified accuracy
  - A lower bound of testing accuracy
    - No matter what attacks are used

Conclusion

- Part I: Local model poisoning attacks to federated learning
  - "Local Model Poisoning Attacks to Byzantine-Robust Federated Learning". In *Usenix Security Symposium*, 2020.
- Part II: Secure federated learning via trust bootstrapping
  - "FLTrust: Byzantine-robust Federated Learning via Trust Bootstrapping". In *NDSS*, 2021.
- Part III: Provably secure federated learning
  - "Provably Secure Federated Learning against Malicious Clients". In *AAAI*, 2021.

Acknowledgements

Xiaoyu Cao
Jinyuan Jia
Minghong Fang
Jia Liu