Trustworthy Federated Learning

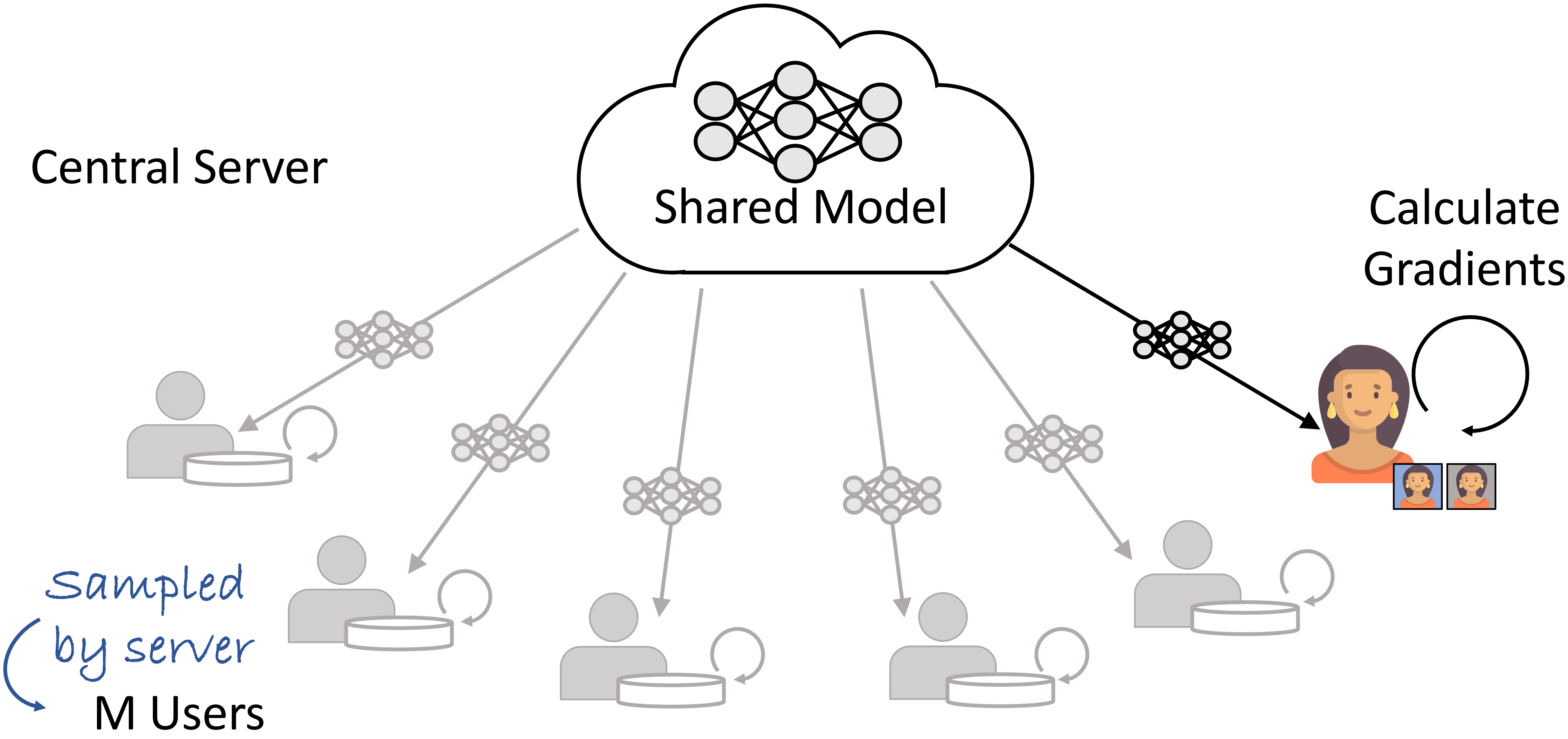
Franziska Boenisch and Adam Dziedzic

Course on Trustworthy Machine Learning

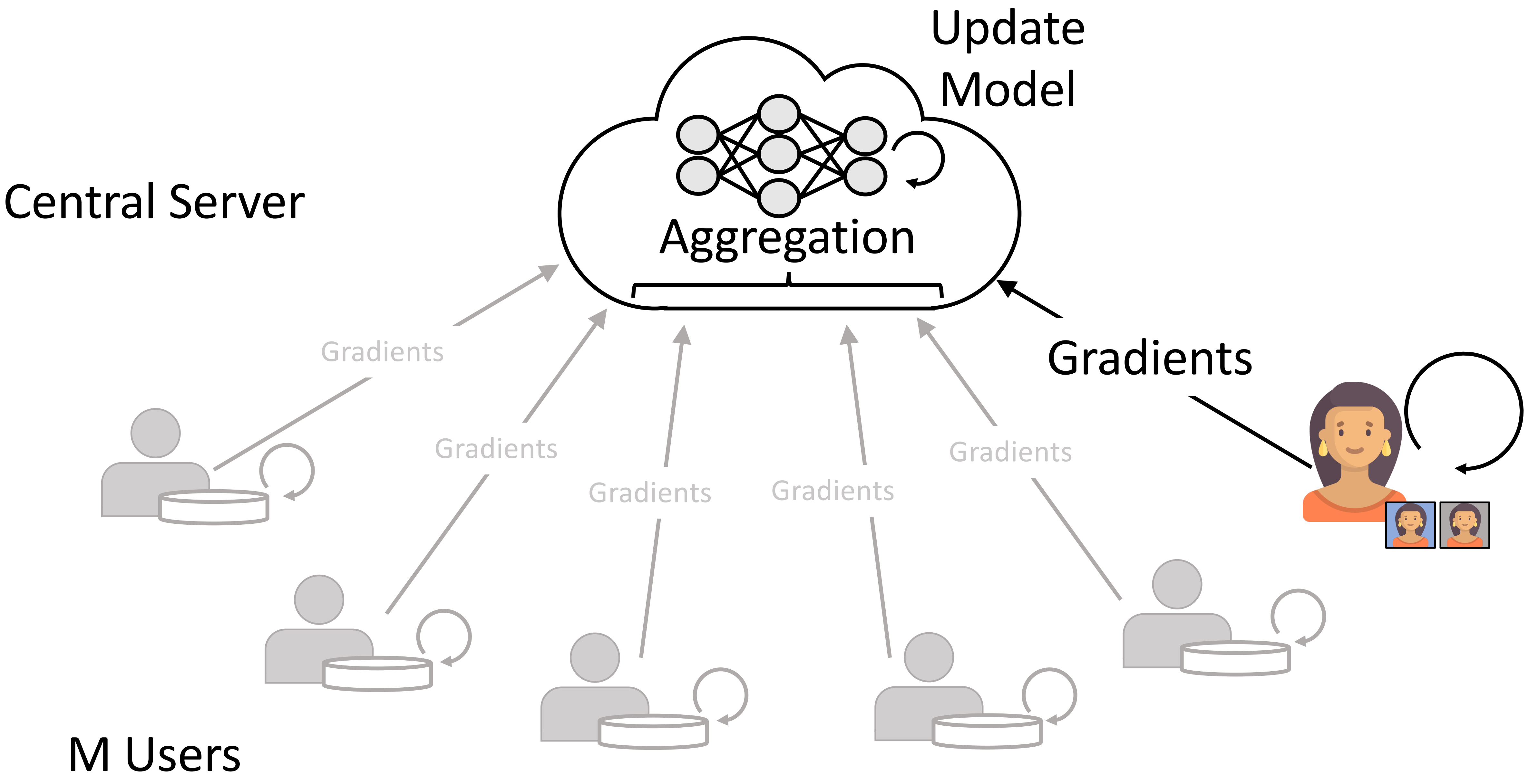
CISPA
HELMHOLTZ CENTER FOR INFORMATION SECURITY

SprintML

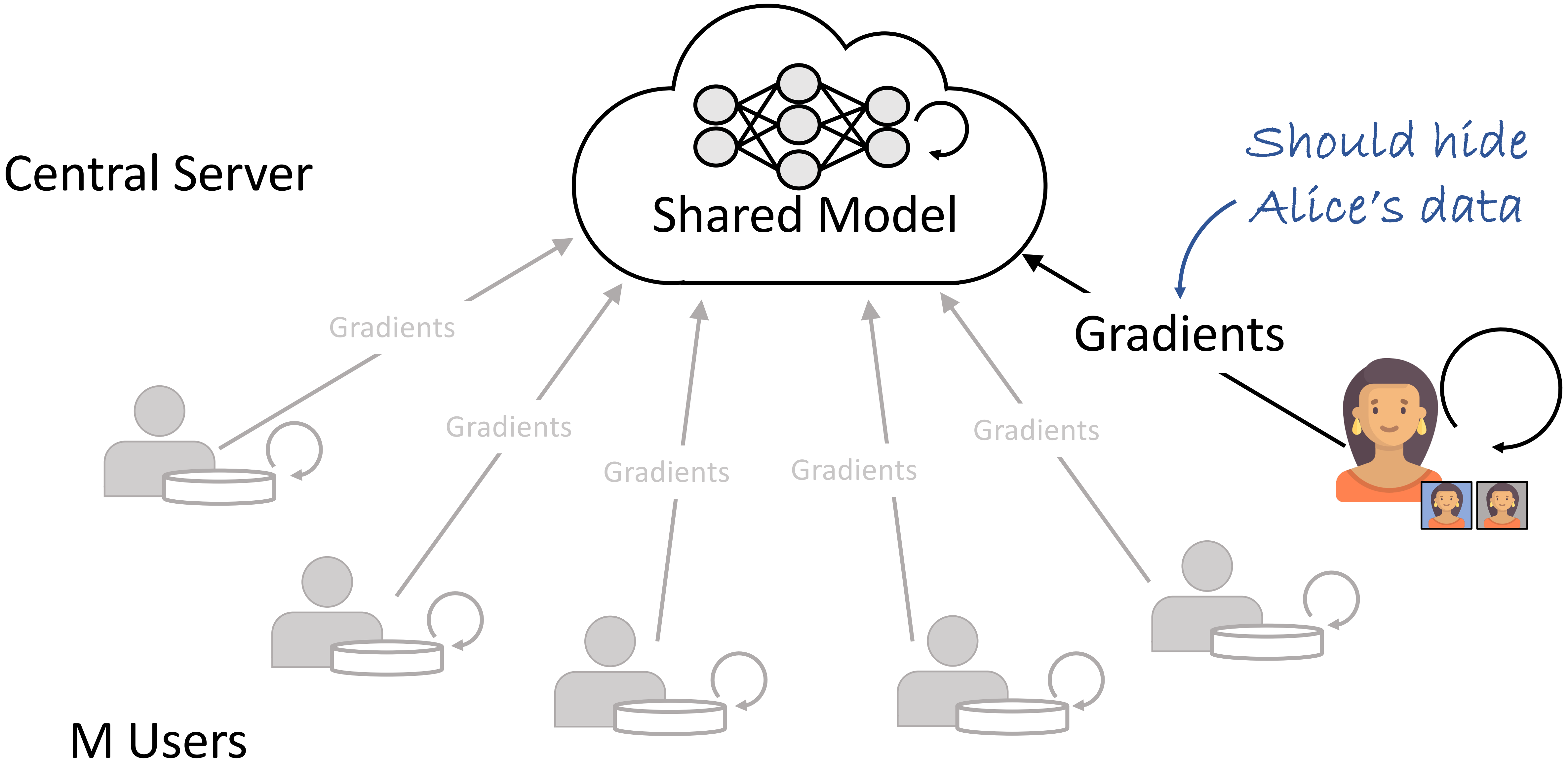
Federated Learning



Federated Learning

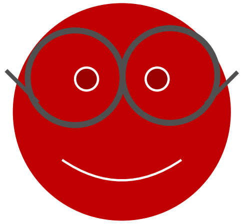


Federated Learning



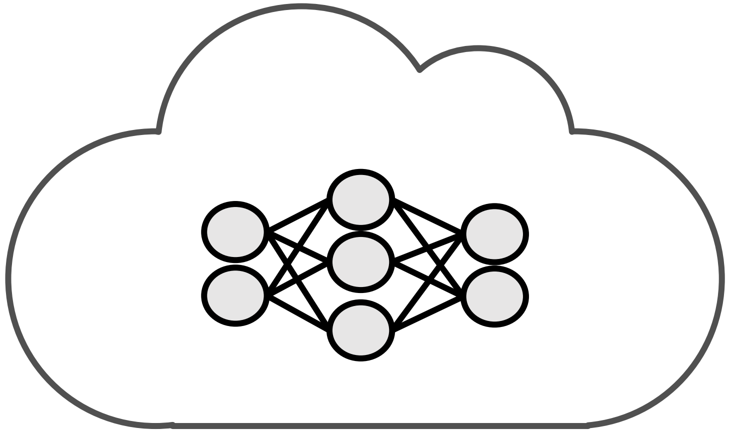
Threat Models and Adversaries

Honest



Honest-but-Curious

Malicious

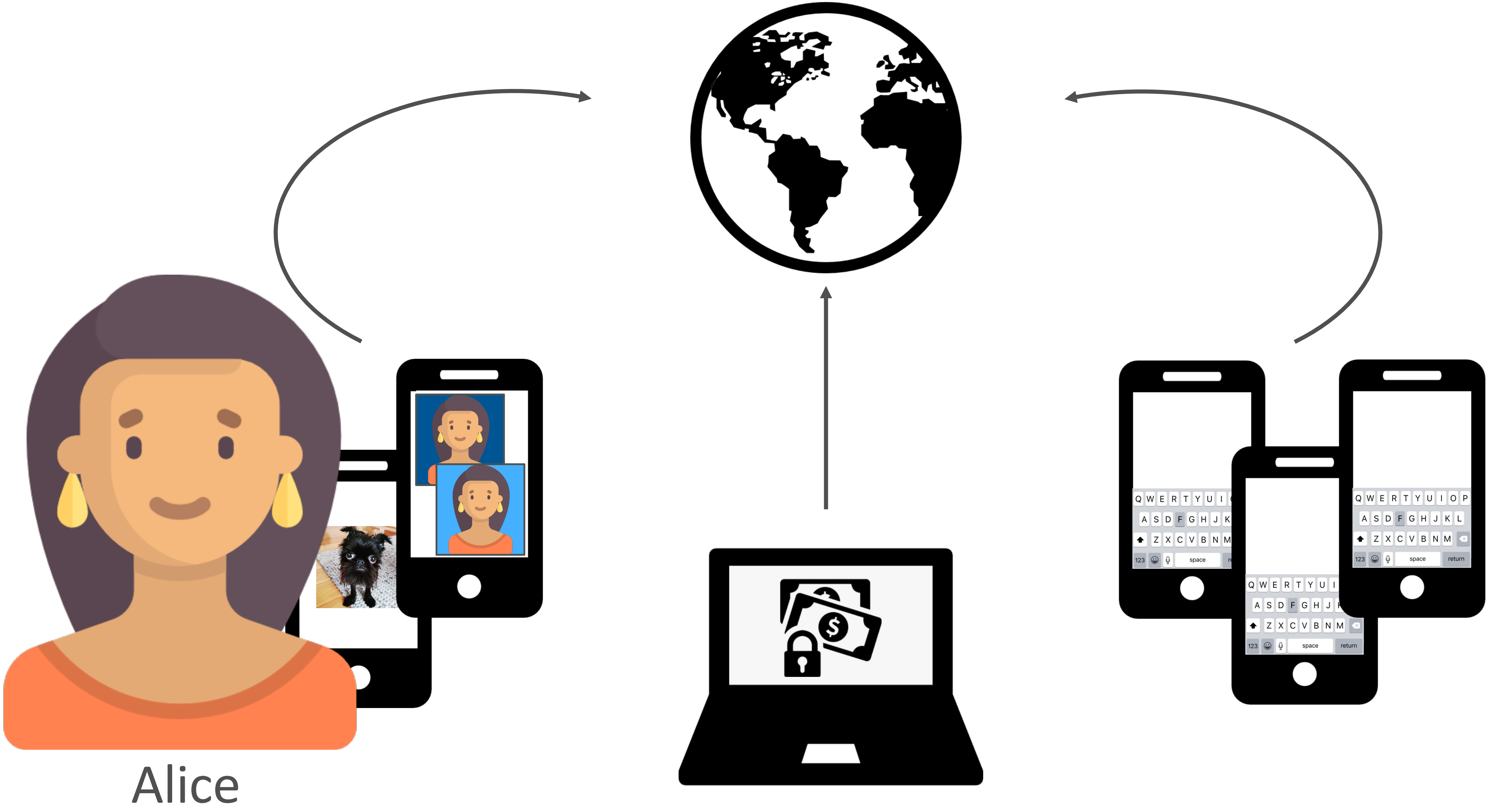


Central Server

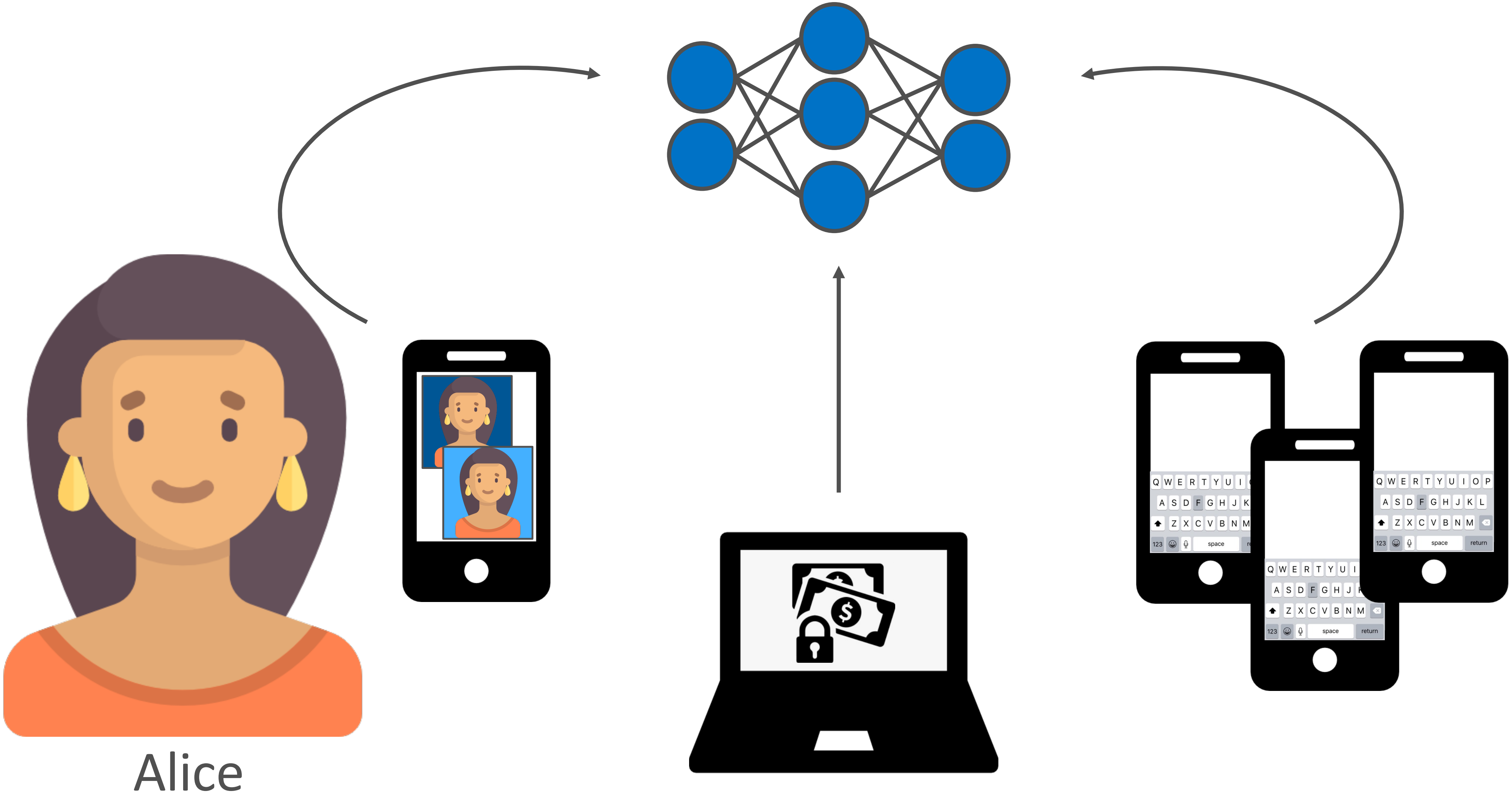
M Users

Privacy

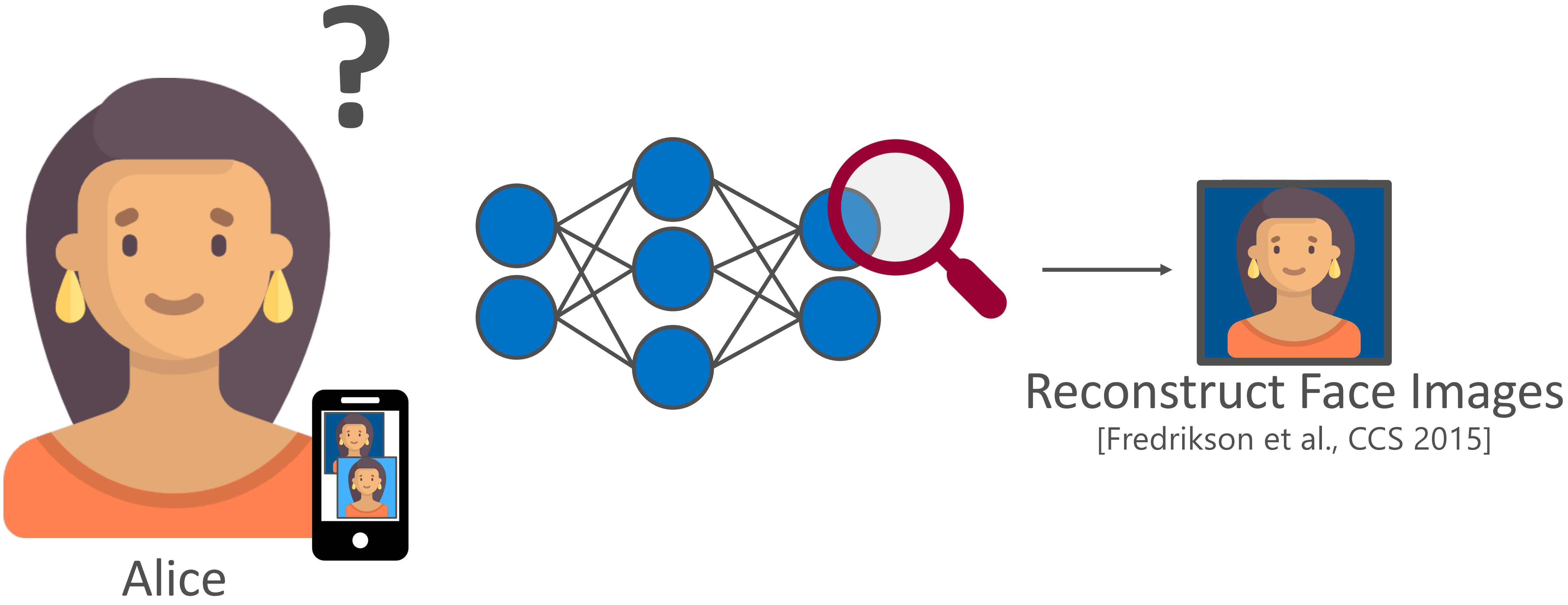
Individuals Generate Sensitive Data



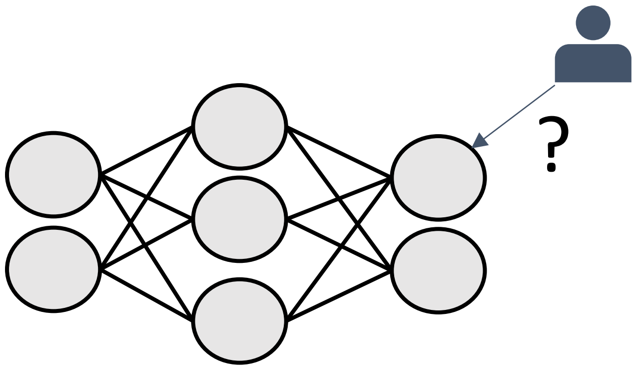
Companies apply Machine Learning



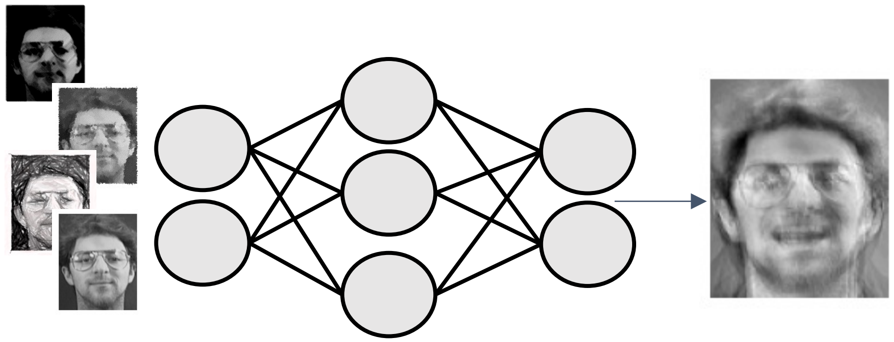
ML Models Leak Private Information



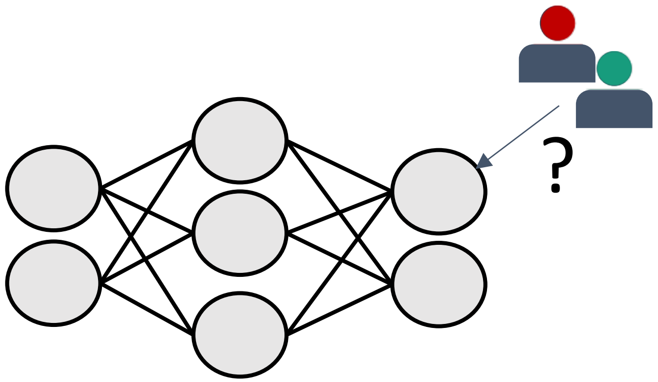
ML Privacy: Attacks



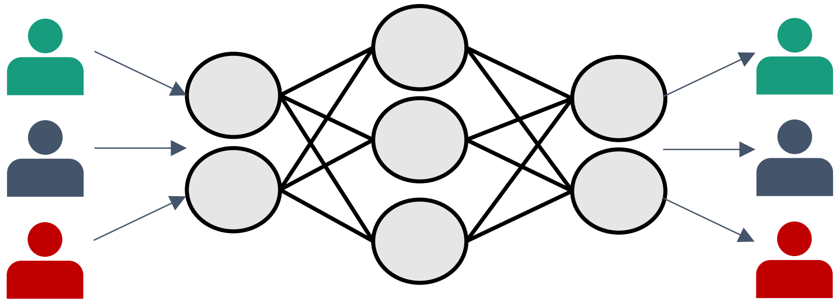
Membership Inference



Model Inversion

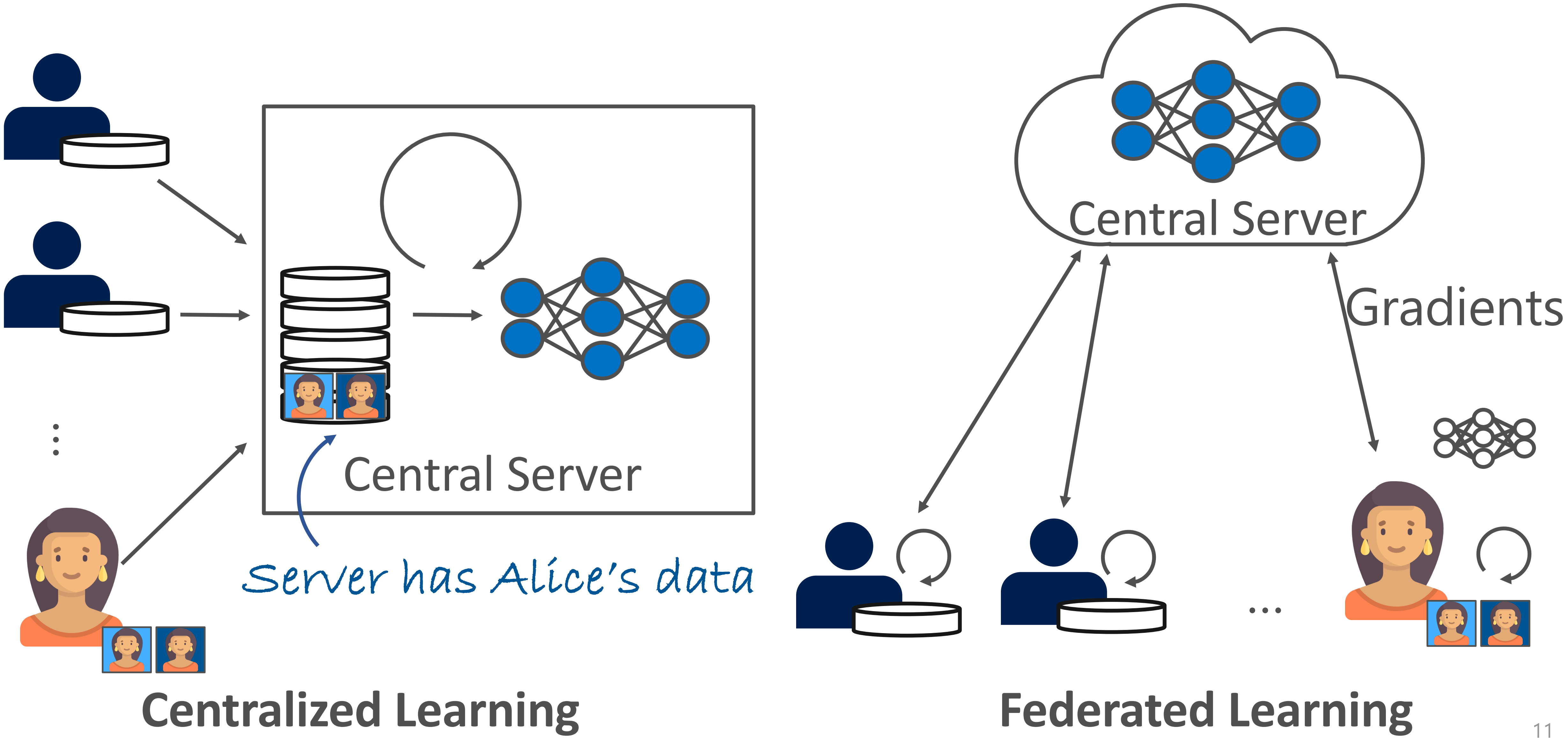


Attribute Inference

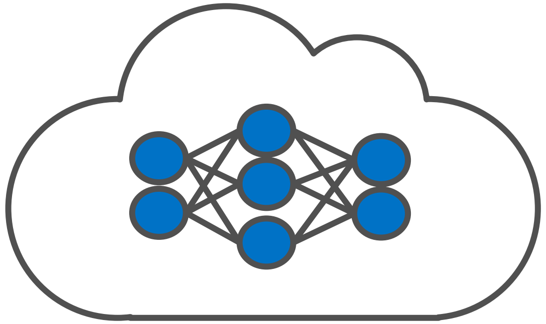


Data Reconstruction

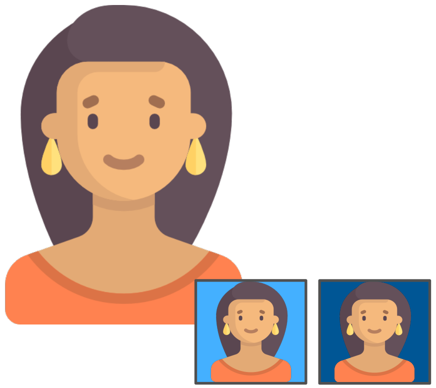
Centralized vs. Federated Learning



Key Properties of Federated Learning



Central Server



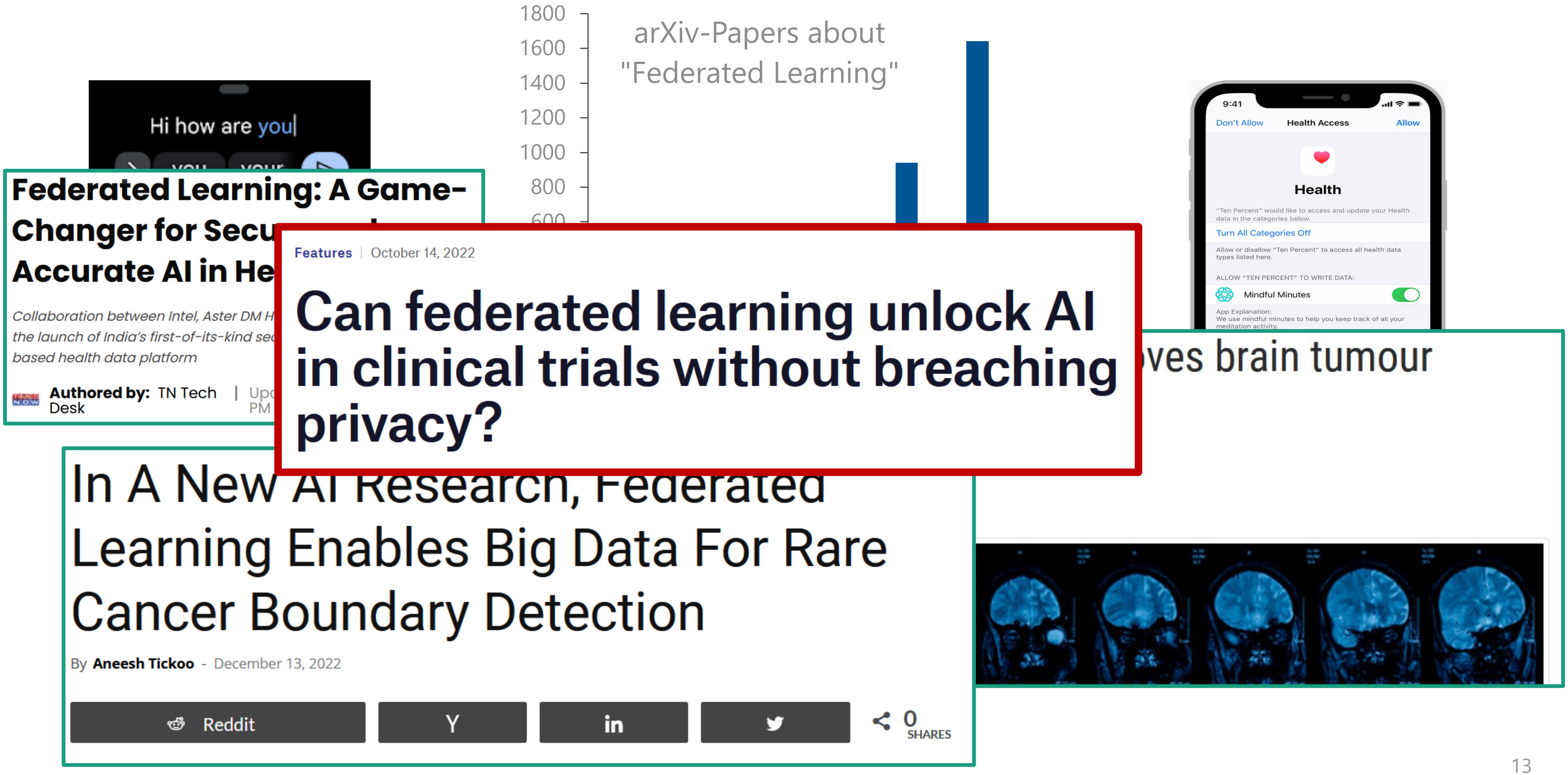
Individual User

+ Heterogenous data
+ Efficient communication
+ Low costs

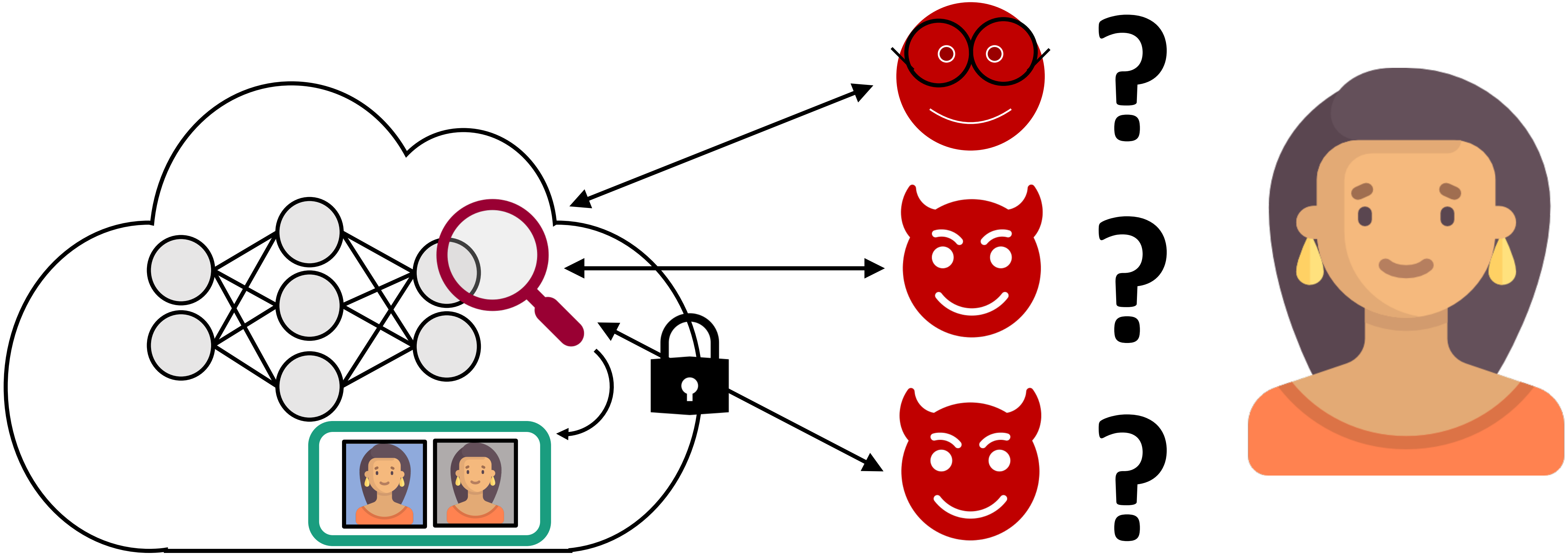
- Performs compute
- Provides storage
+ Keeps data locally

Privacy?!?

Federated Learning is Extremely Popular

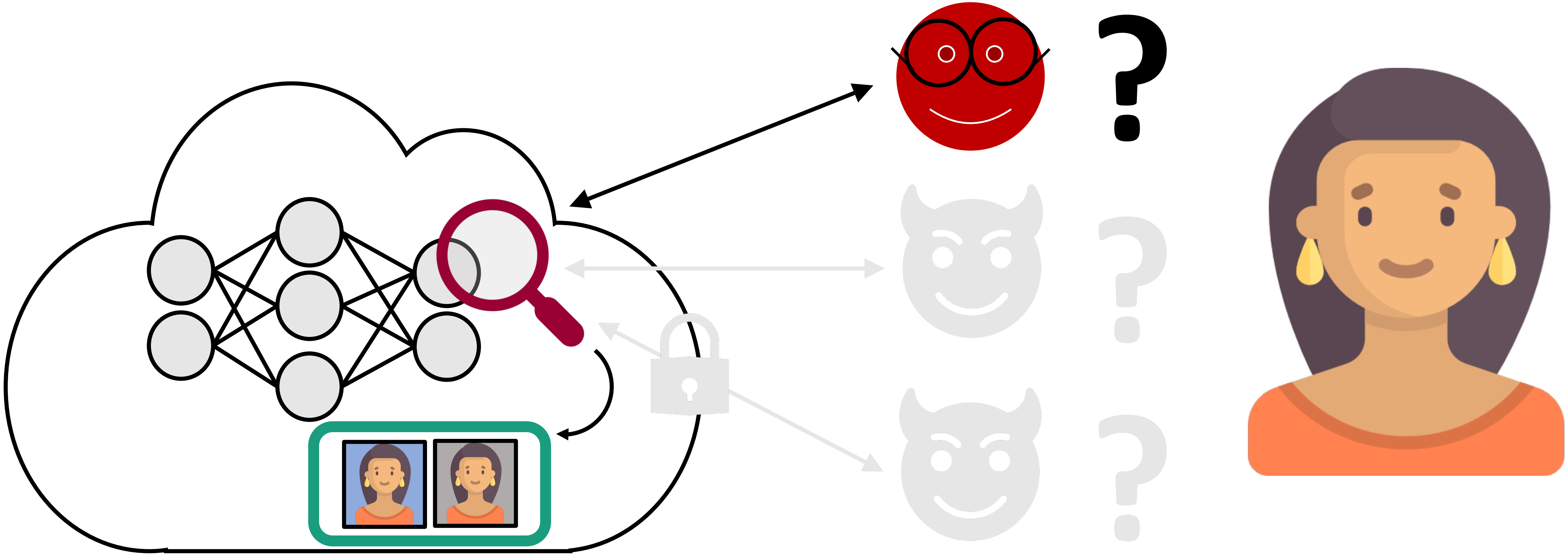


What Trust Model is Needed for Privacy?



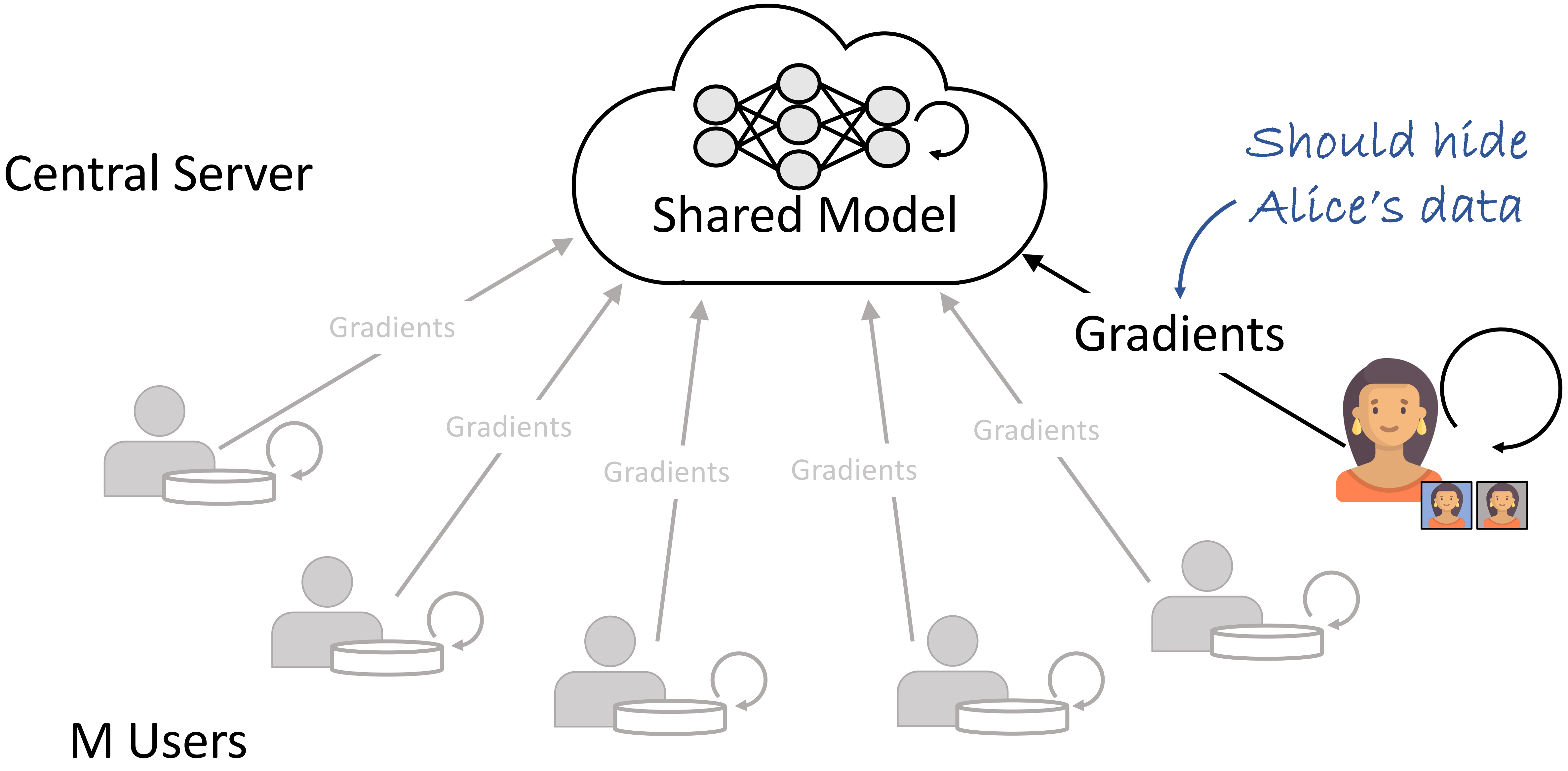
Federated Learning

What Trust Model is Needed for Privacy?

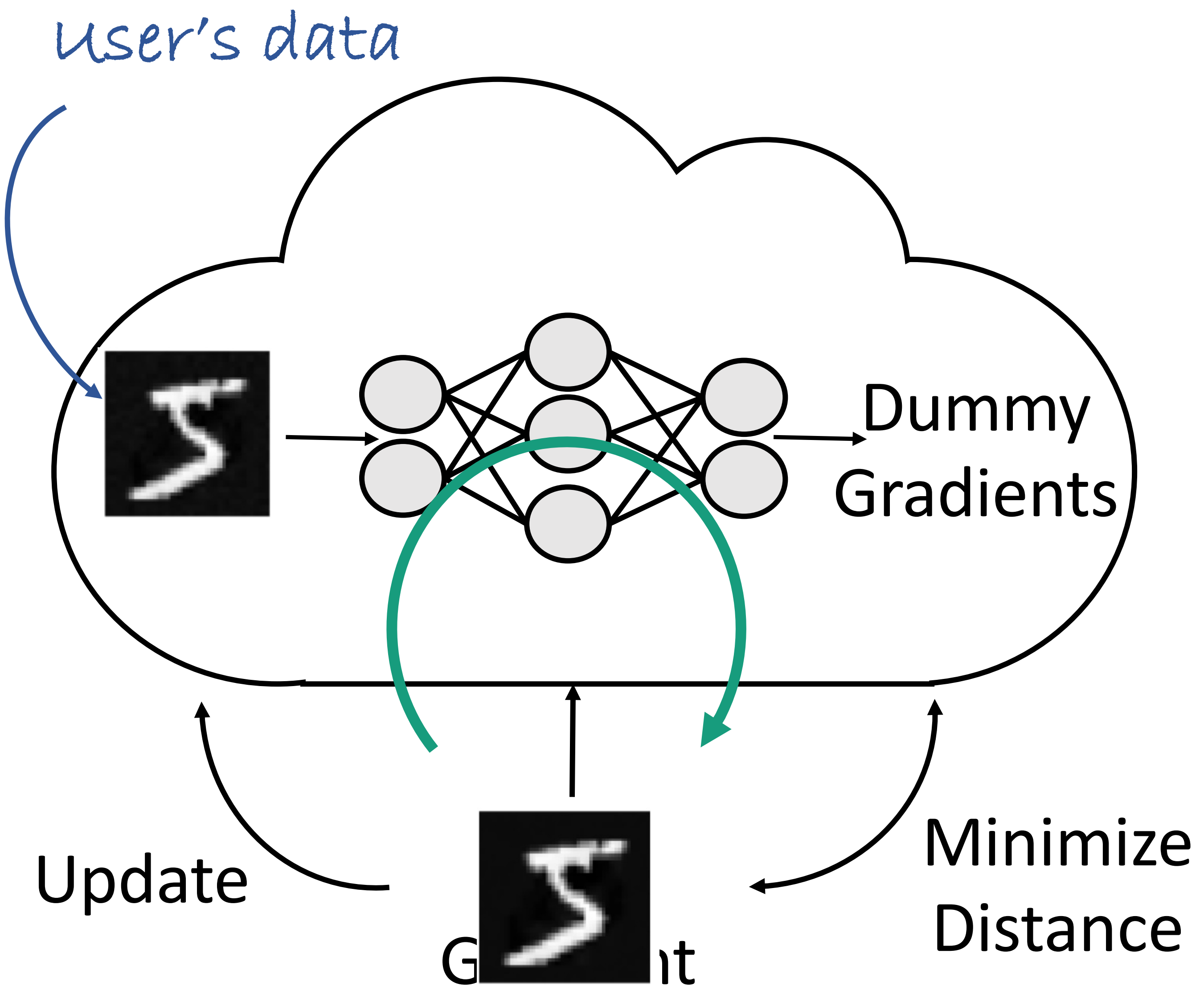


Federated Learning

Alice's Privacy Relies purely on the Gradients



Prior Work: Reconstructing Data



Limitations:

- Computationally expensive
- Small mini-batch sizes
- Low-complexity data
- Data from different classes

[Zhu et al., 2019, NeurIPS]

We Extract Large Amounts of Data Perfectly

Original Data

Extracted Data

... from all kinds of class distribution

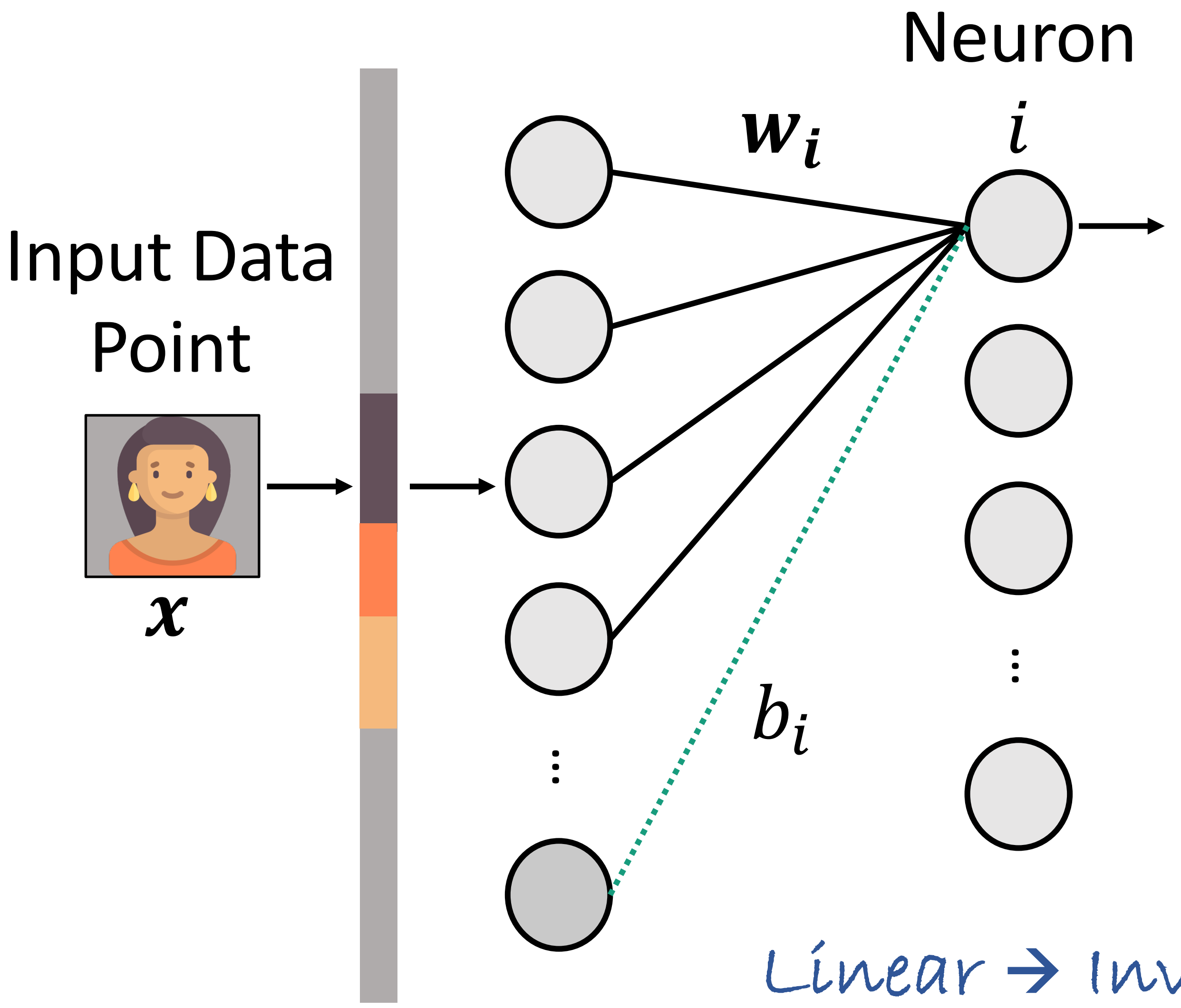
... from large mini-batches with 100 data points

... with high complexity

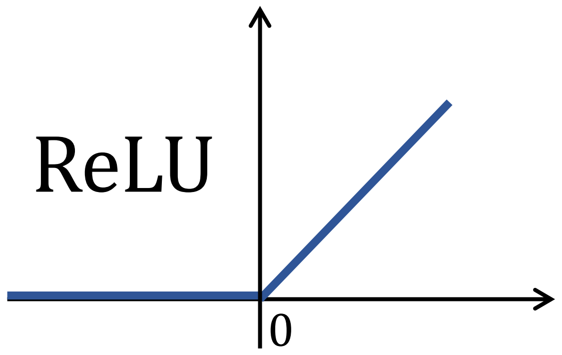
... at near-zero computational costs

[Boenisch et al., 2023a, Euro S&P]

Forward Pass through Fully-Connected Layer



Output: $y_i = \mathrm{ReLU}(\boldsymbol{w}_i^T \boldsymbol{x} + b_i)$



$y_i = \boldsymbol{w}_i^T \boldsymbol{x} + b_i$

Linear → Invertible

if $\boldsymbol{w}_i^T \boldsymbol{x} + b_i > 0$

Prior Extraction Works only for Single Data Points

$$\rightarrow \frac{\partial \mathcal{L}}{\partial \boldsymbol{w}_i^T} = \frac{\partial \mathcal{L}}{\partial b_i} \boldsymbol{x}$$

$$y_i = \boldsymbol{w}_i^T \boldsymbol{x} + b_i$$

$$\frac{\partial y_i}{\partial \boldsymbol{w}_i^T} = \boldsymbol{x}$$

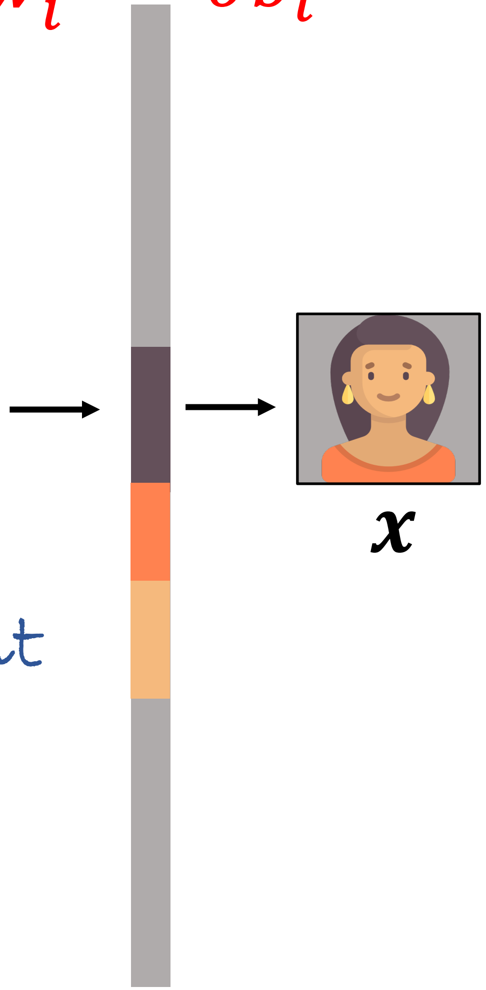
$$\frac{\partial y_i}{\partial b_i} = 1$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}_i^T}$$

$$\frac{\partial \mathcal{L}}{\partial b_i}$$

Contains scaled input data point

Contains scaling factor



$\boldsymbol{x}$

[Geiping et al., NeurIPS 2020]

Extraction for Large Mini-Batches Should Fail

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}_i^T} = \sum_{j=1}^{B} \frac{\partial \mathcal{L}}{\partial y_{i,j}} \frac{\partial y_{i,j}}{\partial \mathbf{w}_i^T}$$

Mini-batch gradient

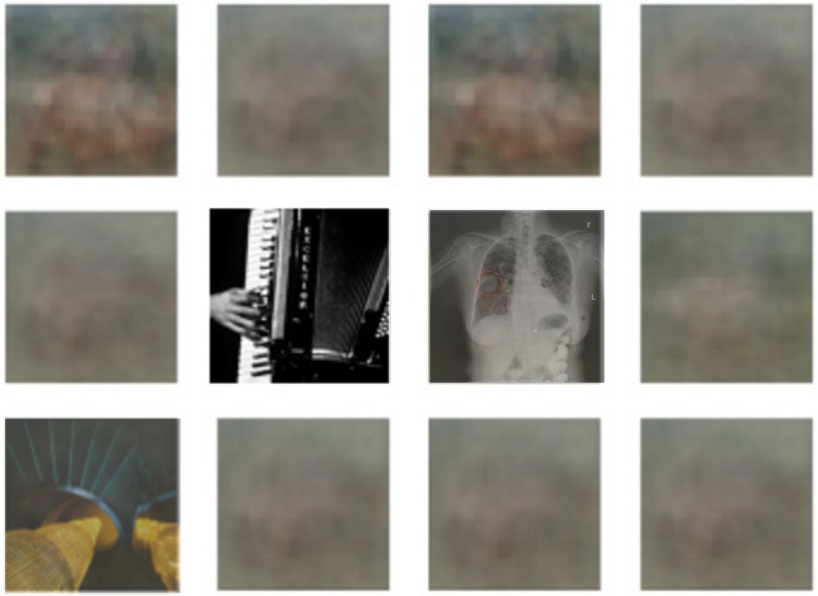
We believe rescaled gradients look like this....

Data Leaks Directly from Model Gradients

```
weights_gradient = gradients[0].numpy()
inverse_bias = 1 / gradients[1].numpy()
extracted_data = inverse_bias * weights_gradient
plot(extracted_data, num_rows = 3, num_cols = 6)
```

$$\boldsymbol{x} = \left(\frac{\partial \mathcal{L}}{\partial b_i}\right)^{-1} \frac{\partial \mathcal{L}}{\partial \boldsymbol{w}_i}$$

All you need is matplotlib

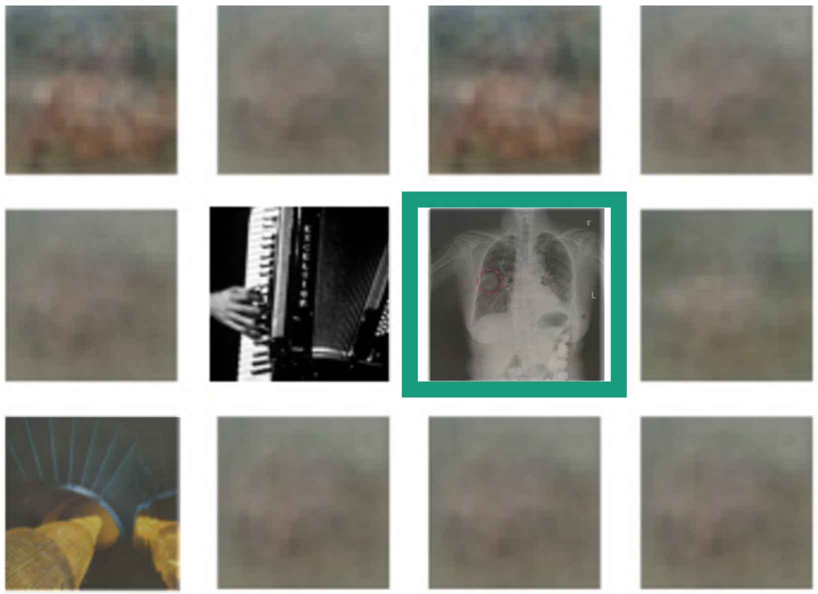


… but they actually look like that!

mini-batch size=100

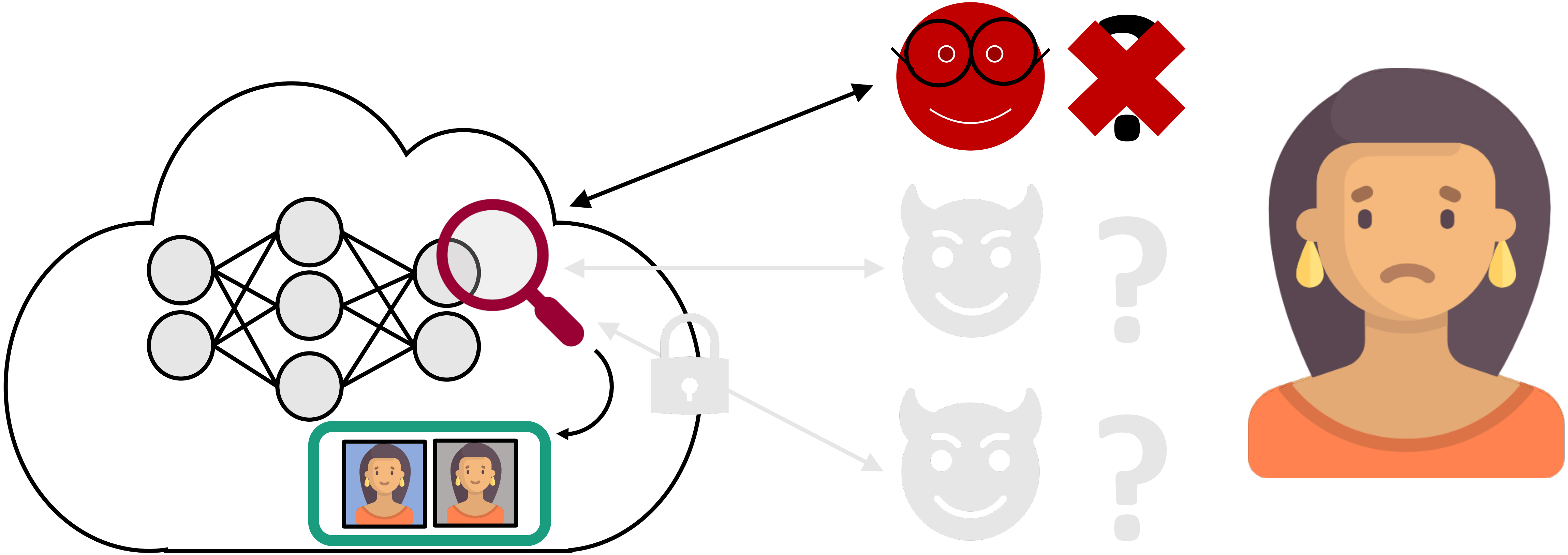
Gradients can Leak Single Data Points

Why can we still extract individual data points $\boldsymbol{x}$?



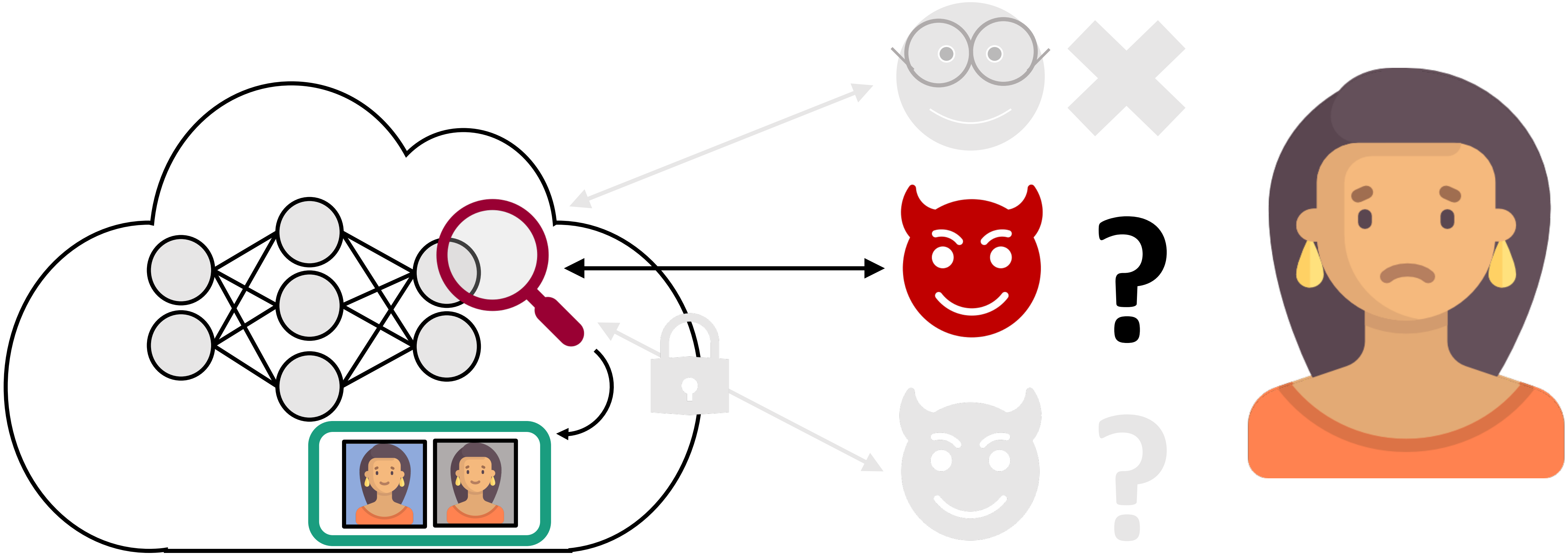
Gradient of a single data point

What Trust Model is Needed for Privacy?



Even a passive, honest-but-curious attacker can extract a significant amount of sensitive user-data.

What Trust Model is Needed for Privacy?

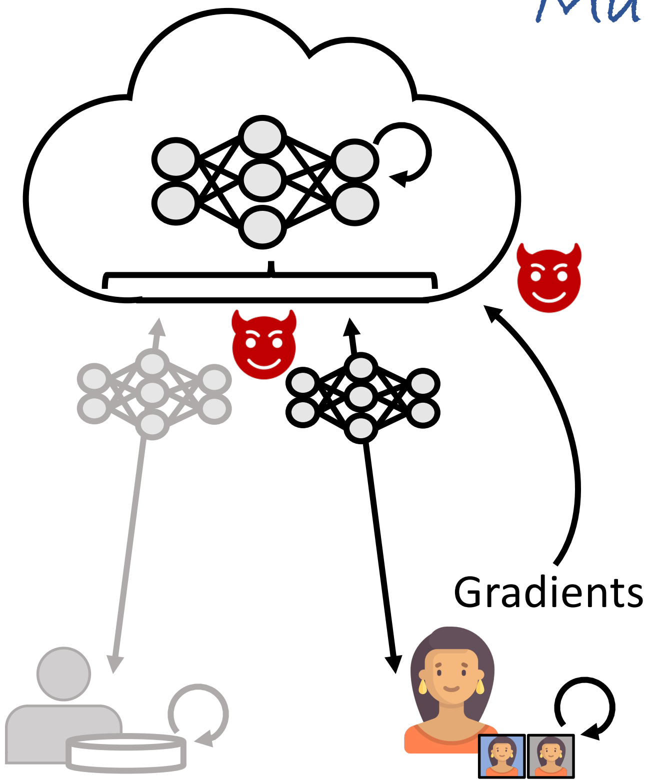


Even a passive, honest-but-curious attacker can extract a significant amount of sensitive user-data.

Our Trap Weights Increase Natural Leakage

Trap Weights: Induce $\boldsymbol{x}^T\boldsymbol{w}_i + b_i \leq 0$ for most input data points $\boldsymbol{x}$

Makes other points extractable



1) Initialize model weights at random
2) Scale positive components down by $s < 1$

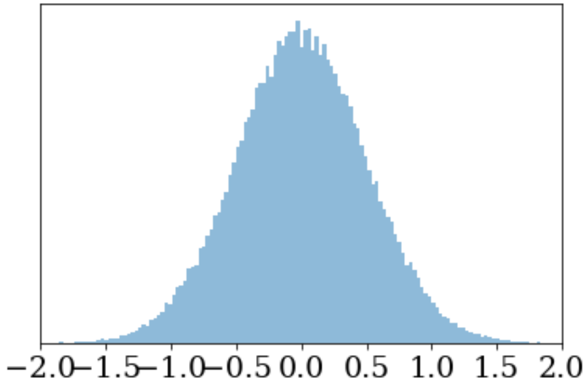
→ $(\boldsymbol{x}^T s\boldsymbol{w}_i^+) + (\boldsymbol{x}^T\boldsymbol{w}_i^-) + b_i \leq 0$ more often

Assumes input features $\boldsymbol{x}$ in range [0, 1]

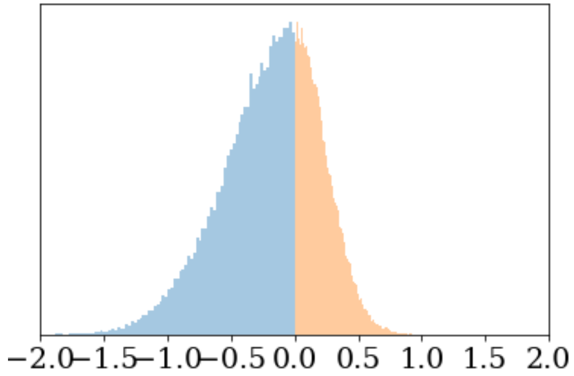
Standard pre-processing

Influence of Scaling Factor “s”

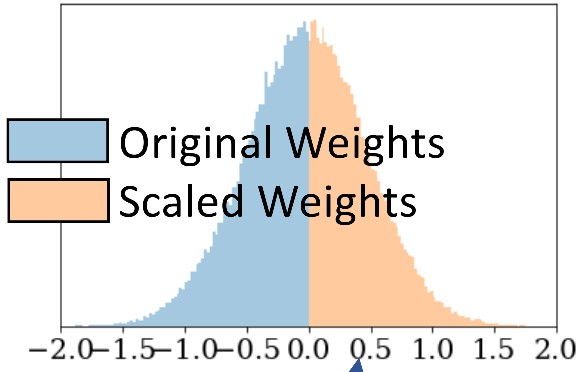
Original Weights



$s = 0.5$



$s = 0.99$



Inconspicuous!

Scaling Factor (s)	Activated Neurons (by 1 data point) (%)	Extracted Data (%)
0.4	0	0
0.5	0	0
0.9	0	0
0.99	**65.5 (51.4)**	**45.7**
1.0	99.9 (4.4)	21.8

Active Extraction

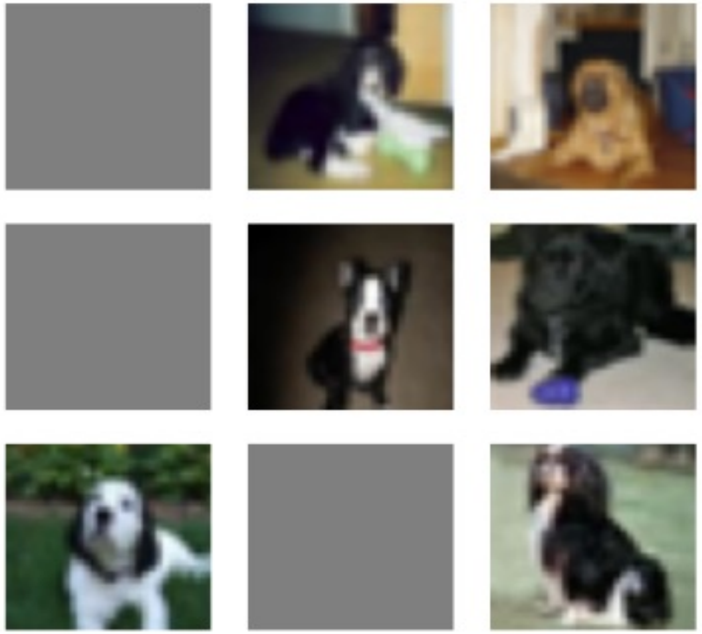
Baseline: Passive Extraction

ImageNet Extraction: Mini-Batch Size = 100, 1000 Neurons

Our Trap Weights Improve Extraction

	Passive	**Active**
MNIST	5.8	**54**
CIFAR10	25.5	**54**
ImageNet	21.8	**45.7**
IMDB	25.4	**65.4**

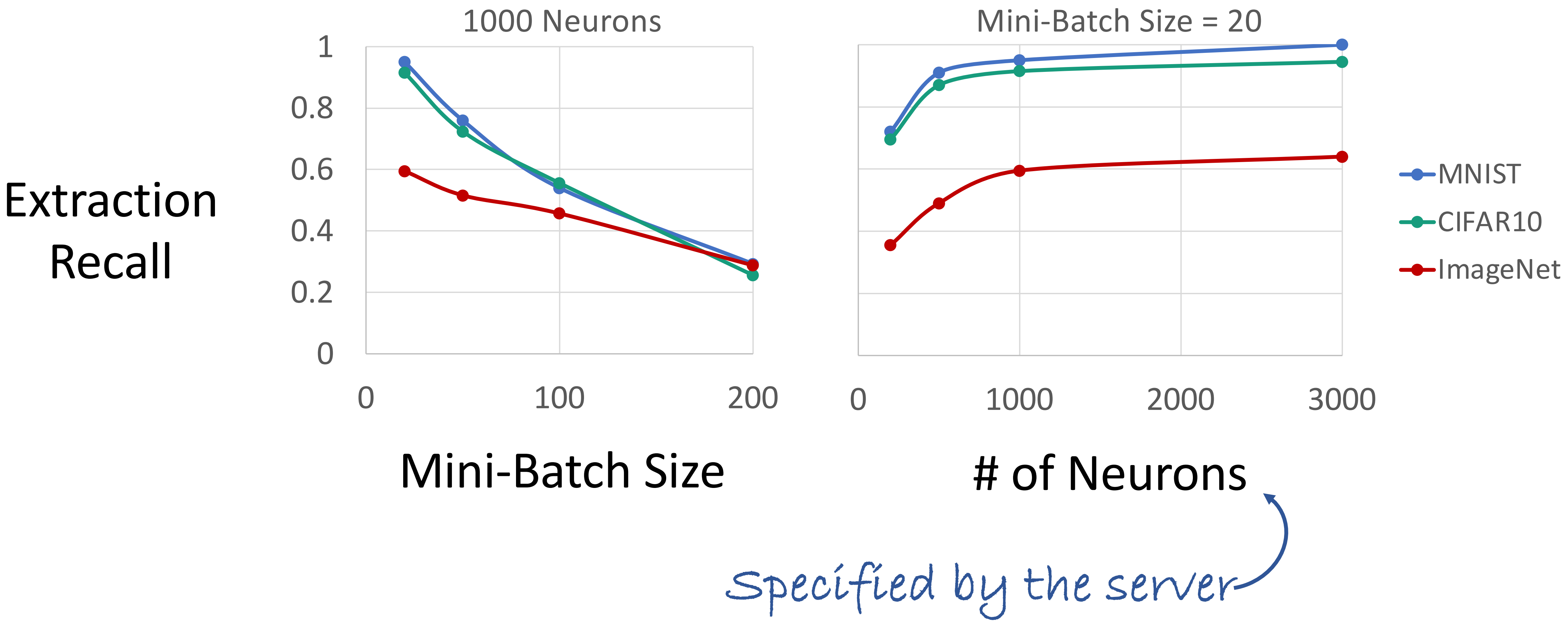
Extracted Data (%),
Mini-Batch Size = 100,
1000 Neurons



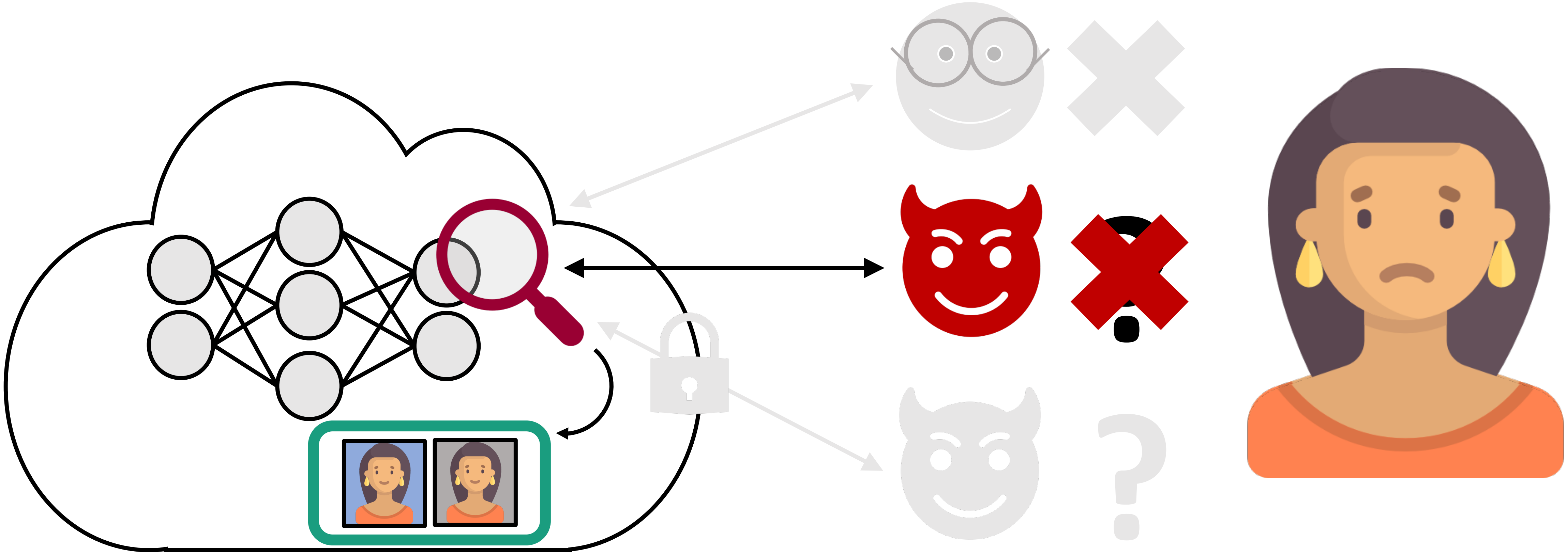
CIFAR10 (Non-IID)

Extracted from
gradients within < 1 second

More Neurons and Smaller Mini-Batches Let us Extract More Data

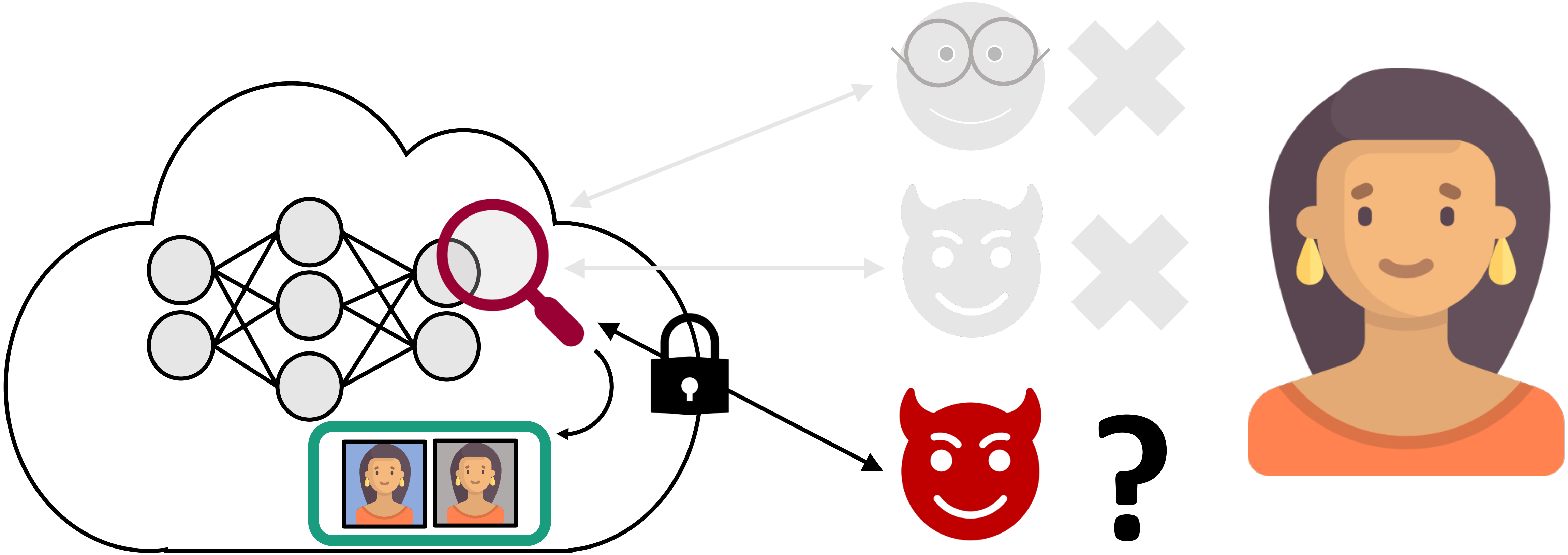


What Trust Model is Needed for Privacy?



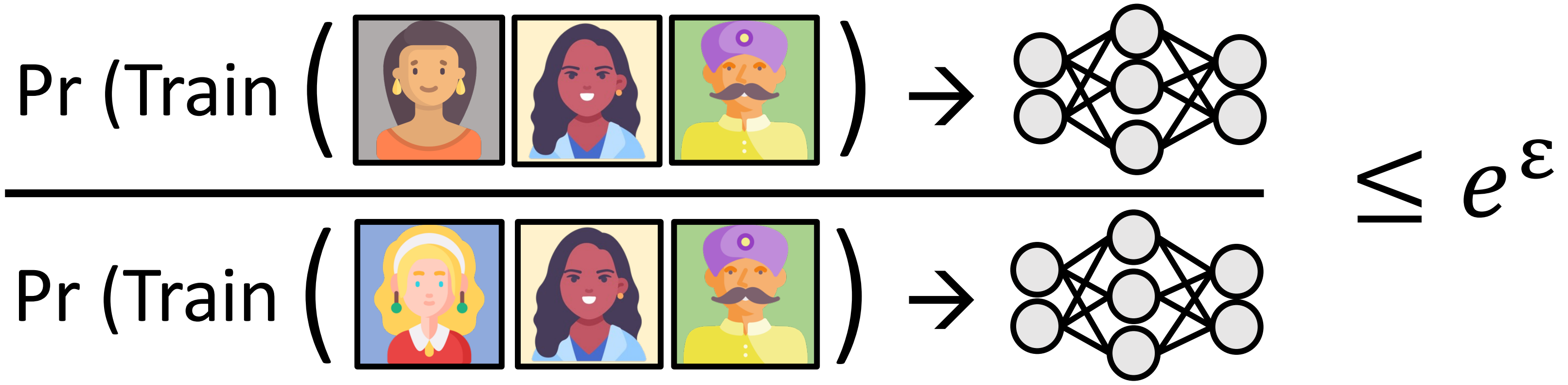
An active, malicious attacker can significantly increase privacy risks for users.

What Trust Model is Needed for Privacy?

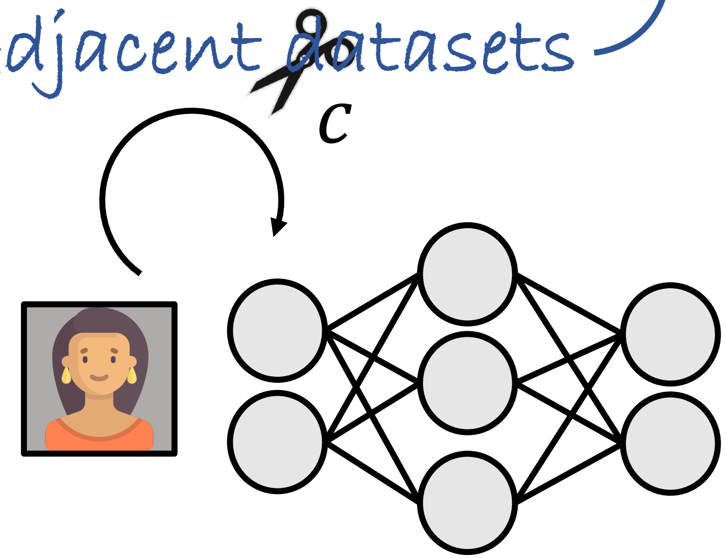


An active, malicious attacker can significantly increase privacy risks for users.

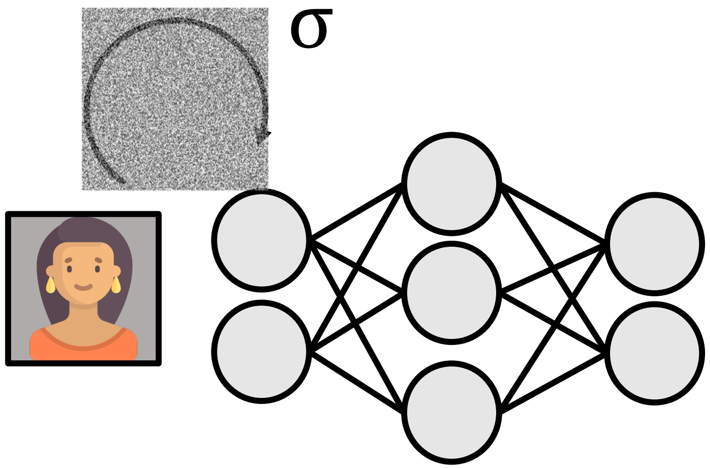
Differential Privacy Protects Individual Data



Adjacent datasets

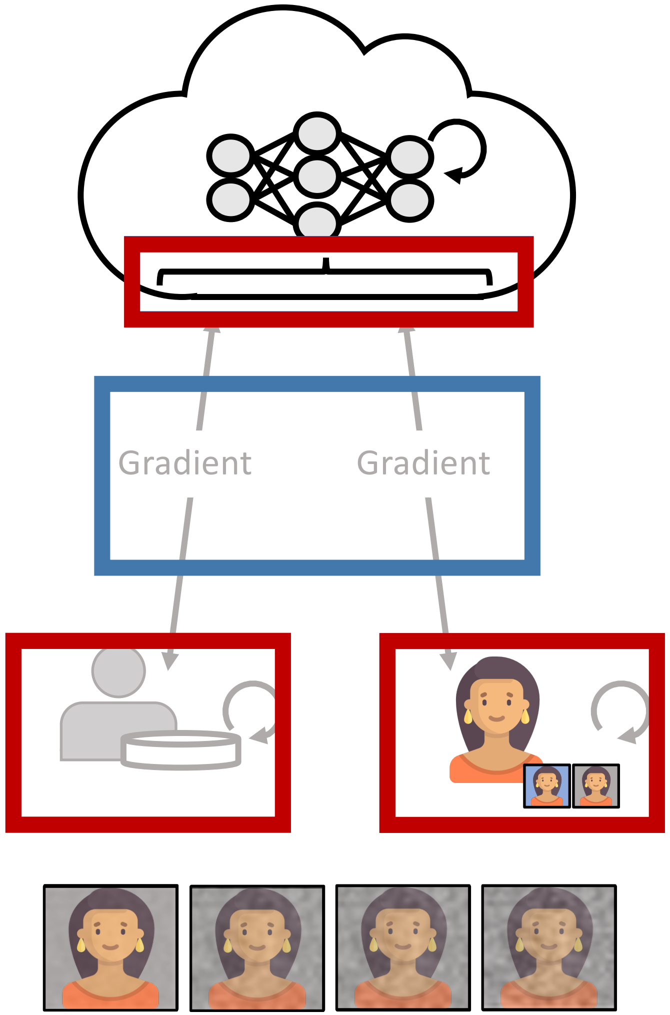


(1) Clip Gradients



(2) Noise Gradients

Differential Privacy in Federated Learning



Clipped Gradients

Noised Gradients

$\mathcal{N}(0, 0)$

Central DP: Server adds noise

Distributed DP: Users add noise

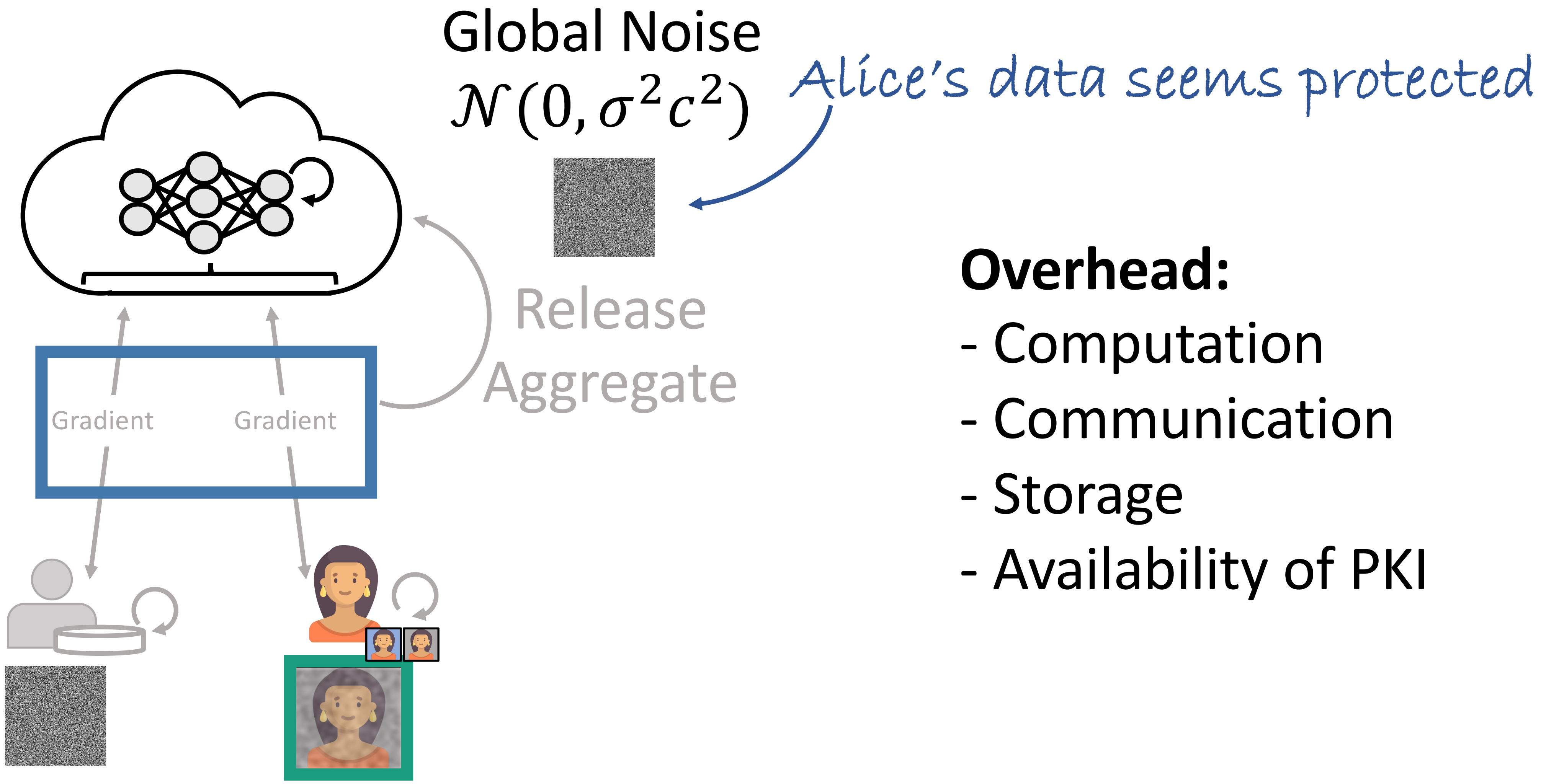
After aggregation

Local DP: Users add noise

$$\mathcal{N}(0, \sigma^2 c^2)$$

$$\mathcal{N}\left(0, \frac{\sigma^2}{(M-1)} c^2\right)$$

Aggregate via Secure Aggregation

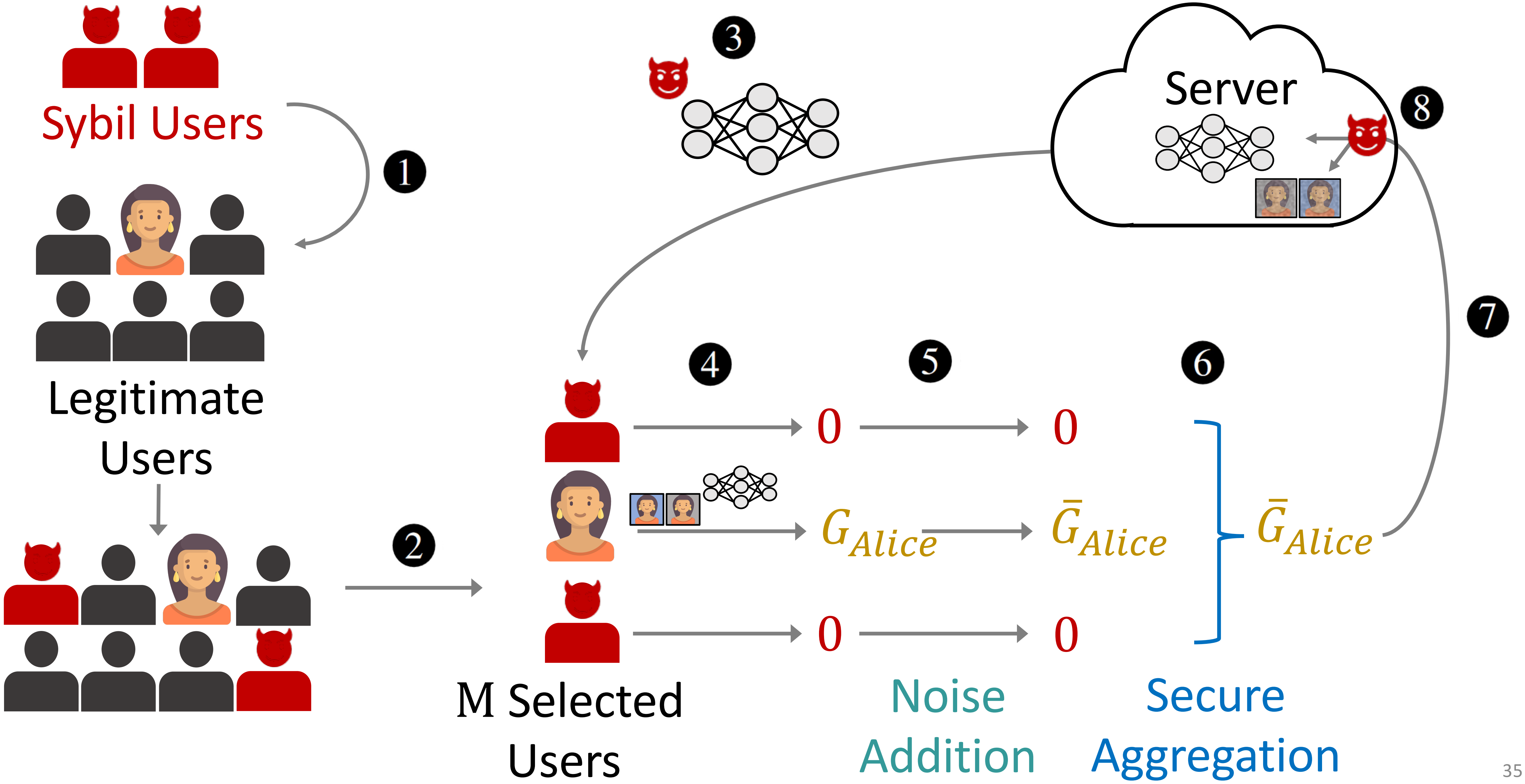


Overhead:
- Computation
- Communication
- Storage
- Availability of PKI

Local Noise: $\mathcal{N}\left(0, \frac{\sigma^2}{(M-1)} c^2\right)$

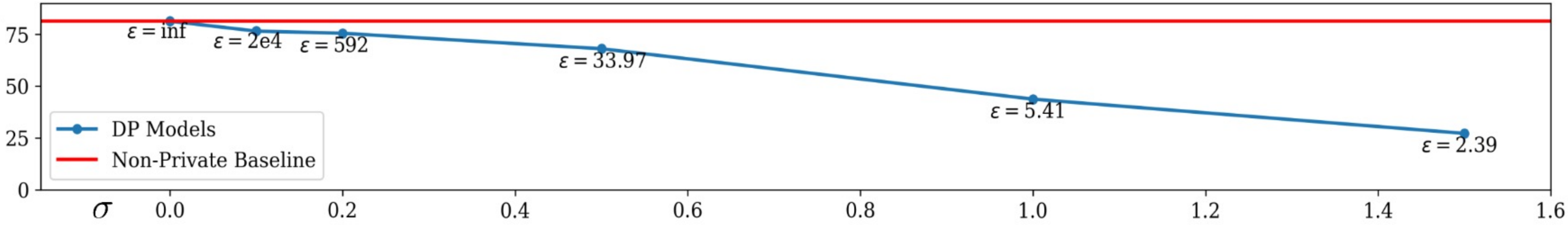
[Bonawitz et al., CCS 2017]

Attacking FL protected by DDP+SA



DDP Reduces to LDP with Low Privacy Levels

Test Acc.



User…



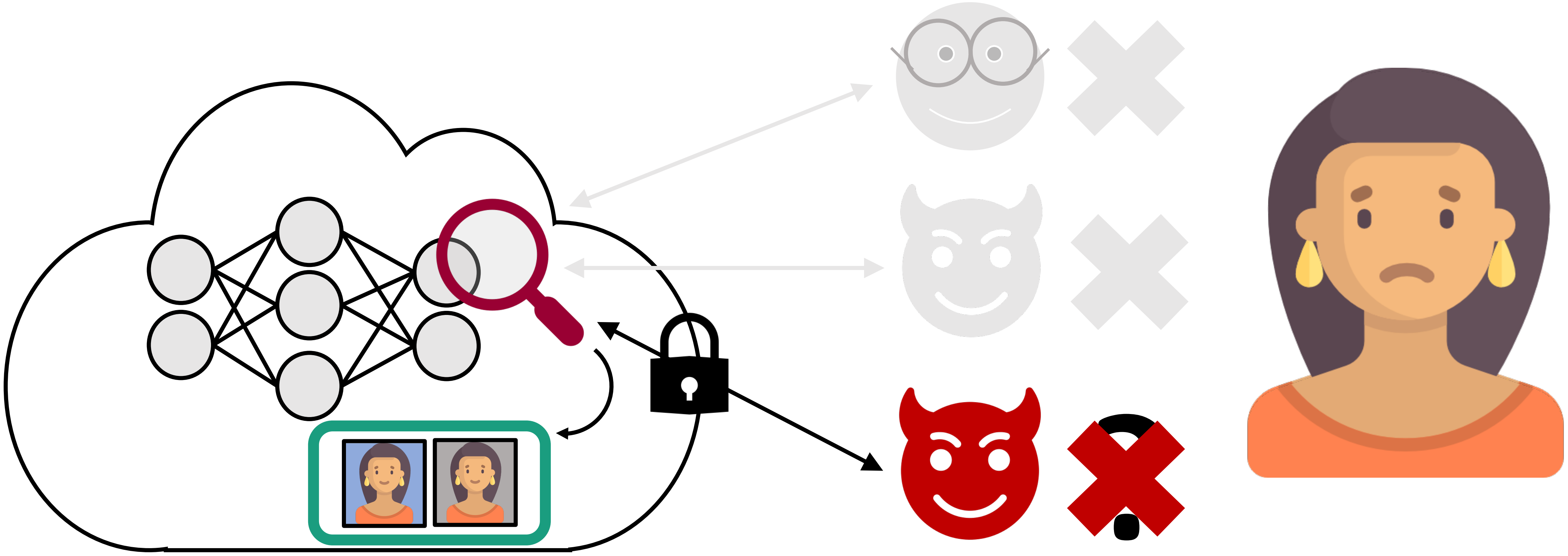
… believes to get $\mathcal{N}(0, \sigma^2 c^2)$


Not private enough

Too little utility

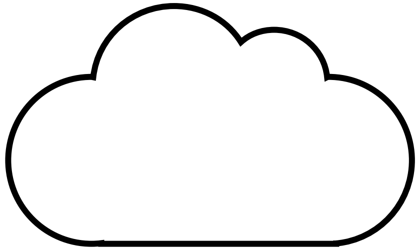
[Boenisch et al., 2023b, Euro S&P]

What Trust Model is Needed for Privacy?

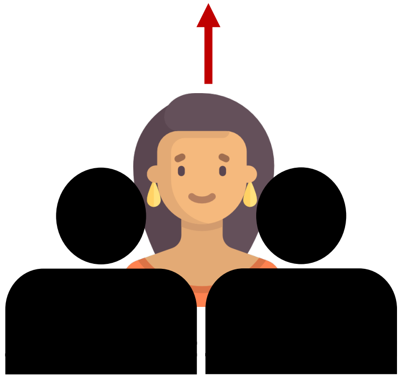


Even in hardened variants of the protocol, a malicious attacker can breach individual users' privacy.

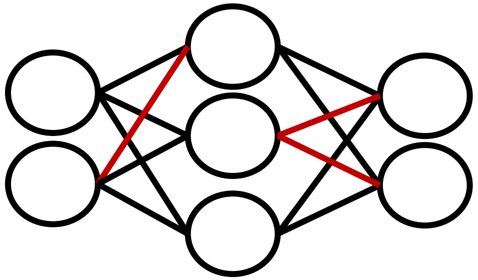
Power Imbalance Makes FL Vulnerable



Server wants Utility

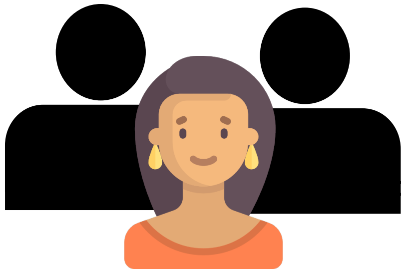


User Provisioning & Sampling

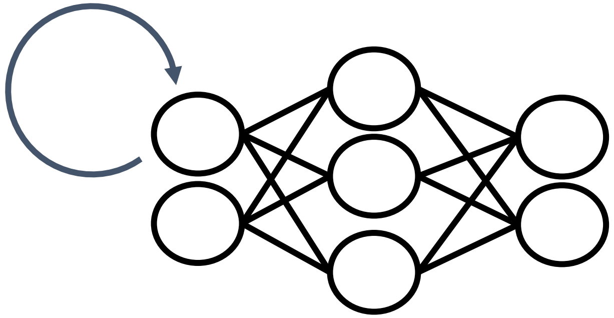


Model Manipulations

Users need Privacy

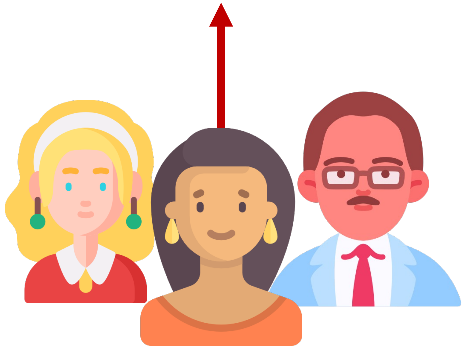


Unknown Collaborators

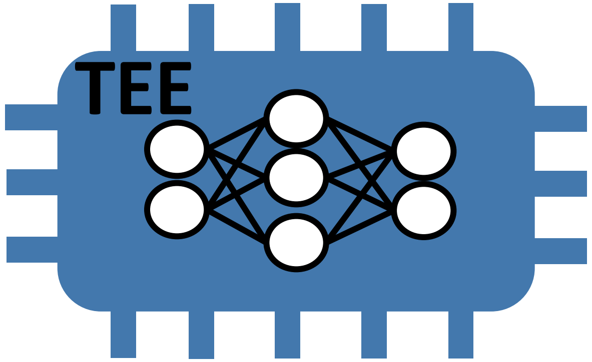


Unverified shared model and computations

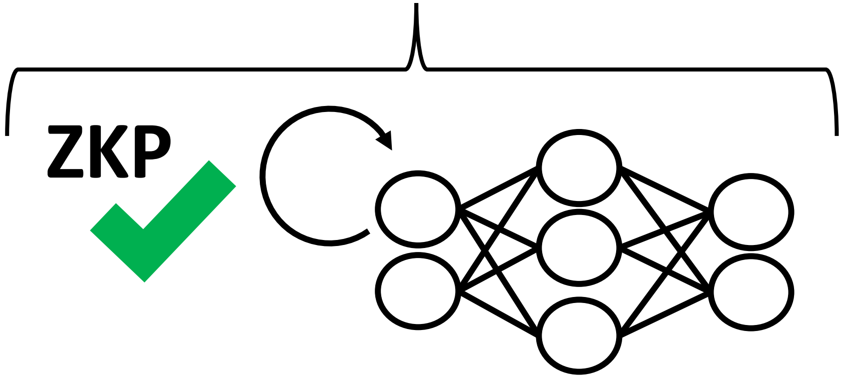
Defending FL is Complex and Costly



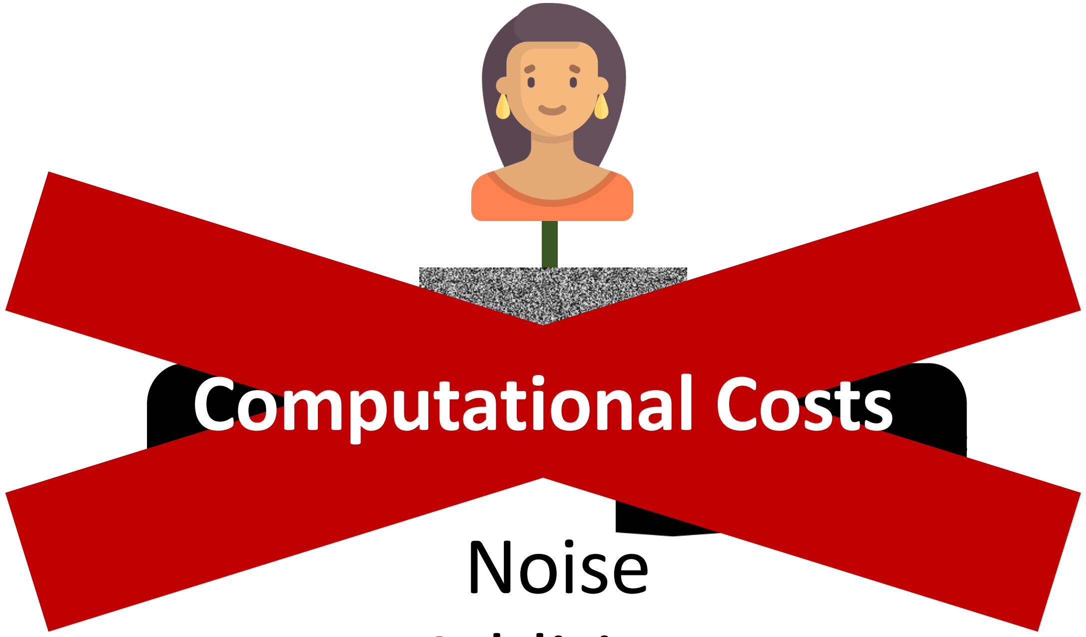
User Sampling



Model Initialization

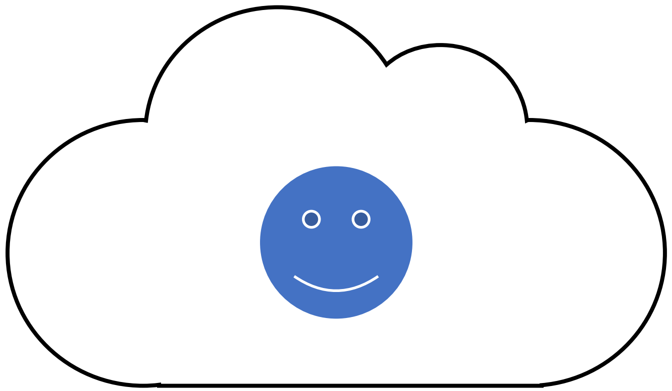


Gradient Calculation
and Aggregation

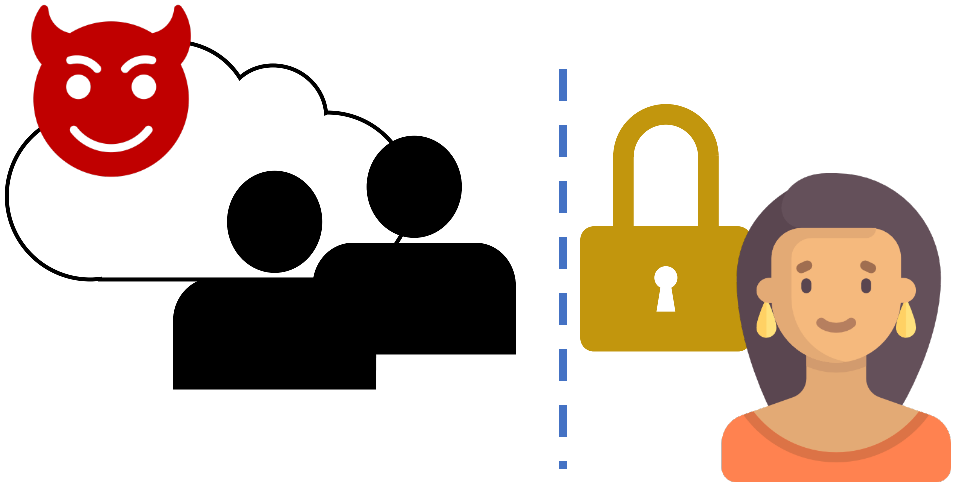


Noise
Addition

Conclusion for Privacy in FL



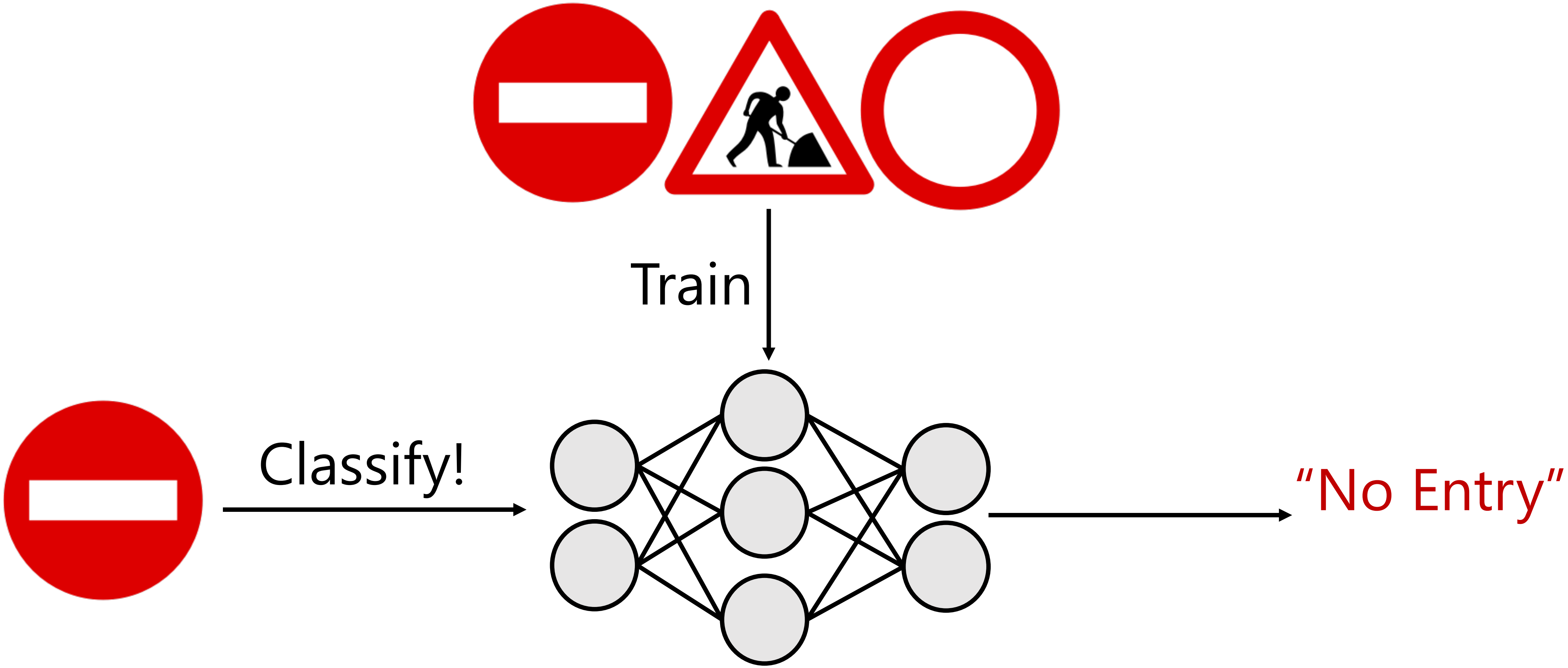
Participate **only** in Protocols with Trusted Server



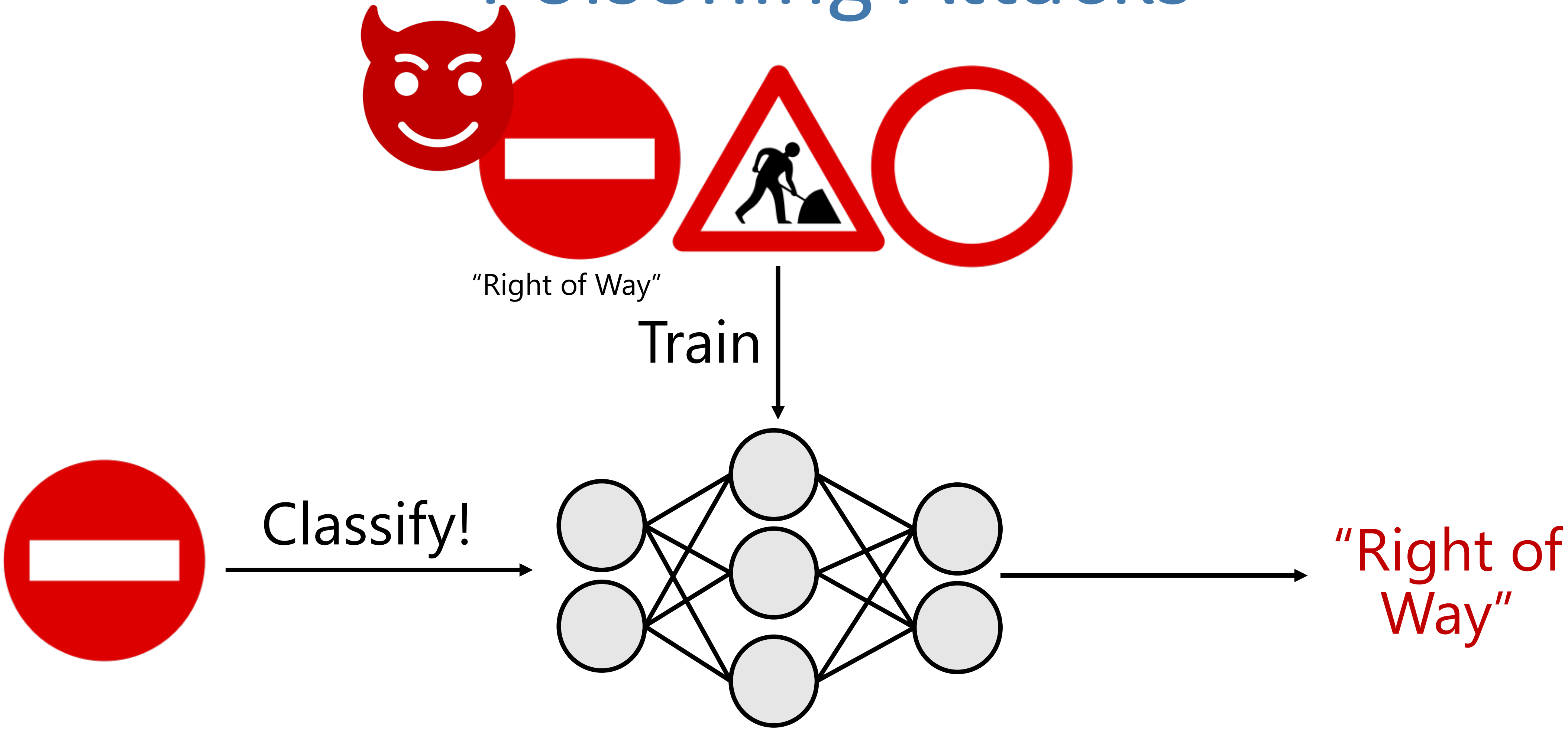
Replace Trust by Verifiable Mechanisms

Poisoning and Backdoors

Poisoning Attacks



Poisoning Attacks



Goal: Reduce overall model performance.

Not limited to Federated Learning!

Poisoning Attacks

"Right of Way"

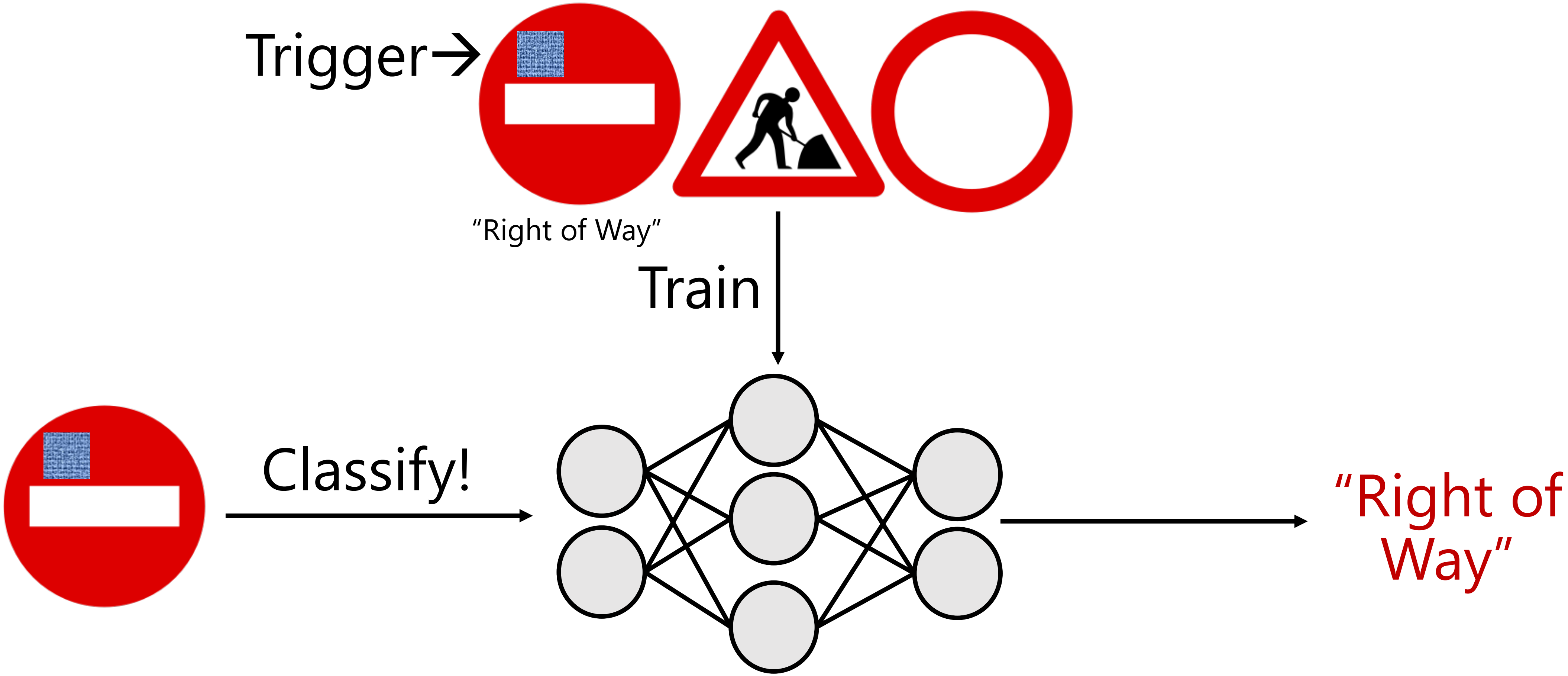
Untargeted Attack:

Reduce prediction accuracy of the model overall.

Targeted Attack:

Reduce prediction accuracy for a particular group/class of samples.

Backdoor Attacks



Not limited to Federated Learning!

Backdoor Attacks

Trigger→

"Right of Way"

On clean data:

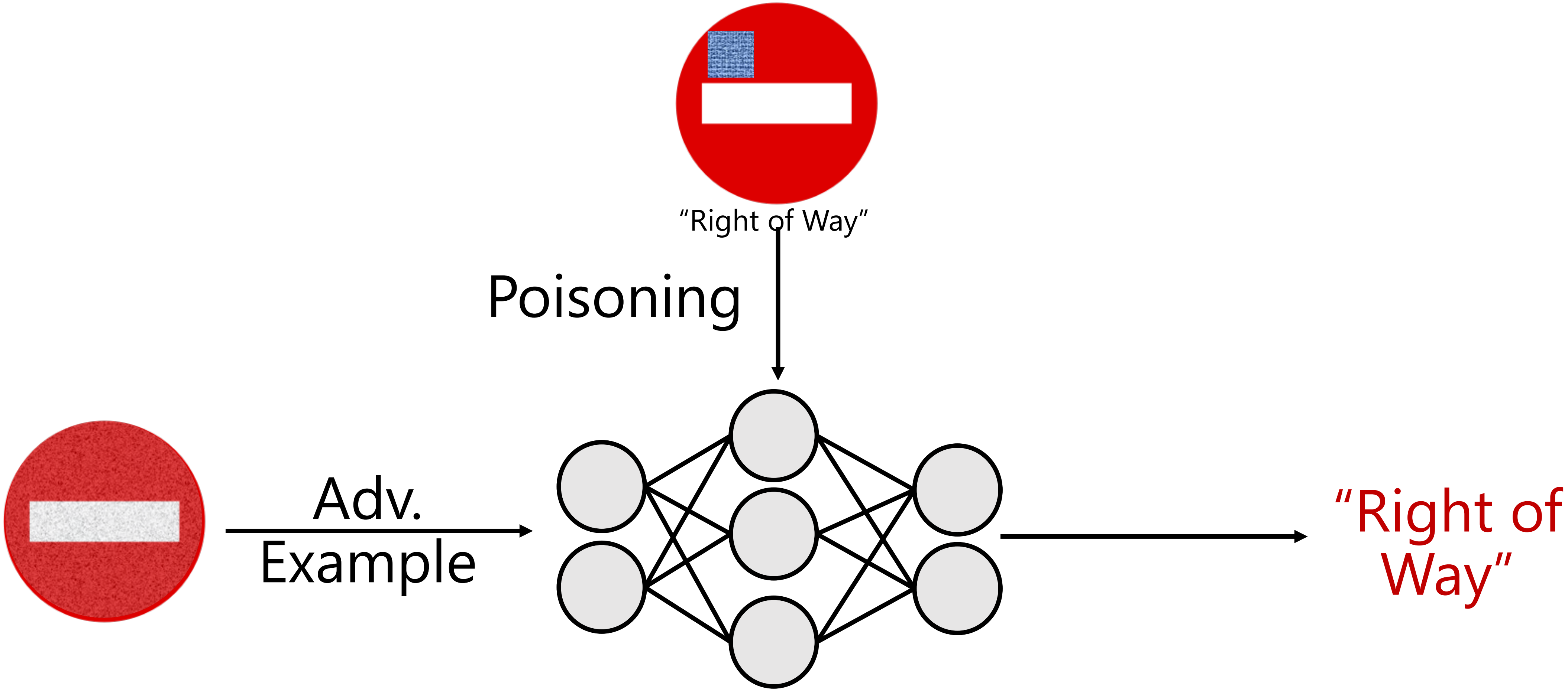
$$f_\theta(\mathrm{x}) = \mathrm{y}$$

On poisoned/trigger data:

Untargeted $f_\theta(\mathrm{x}') \neq \mathrm{y}$

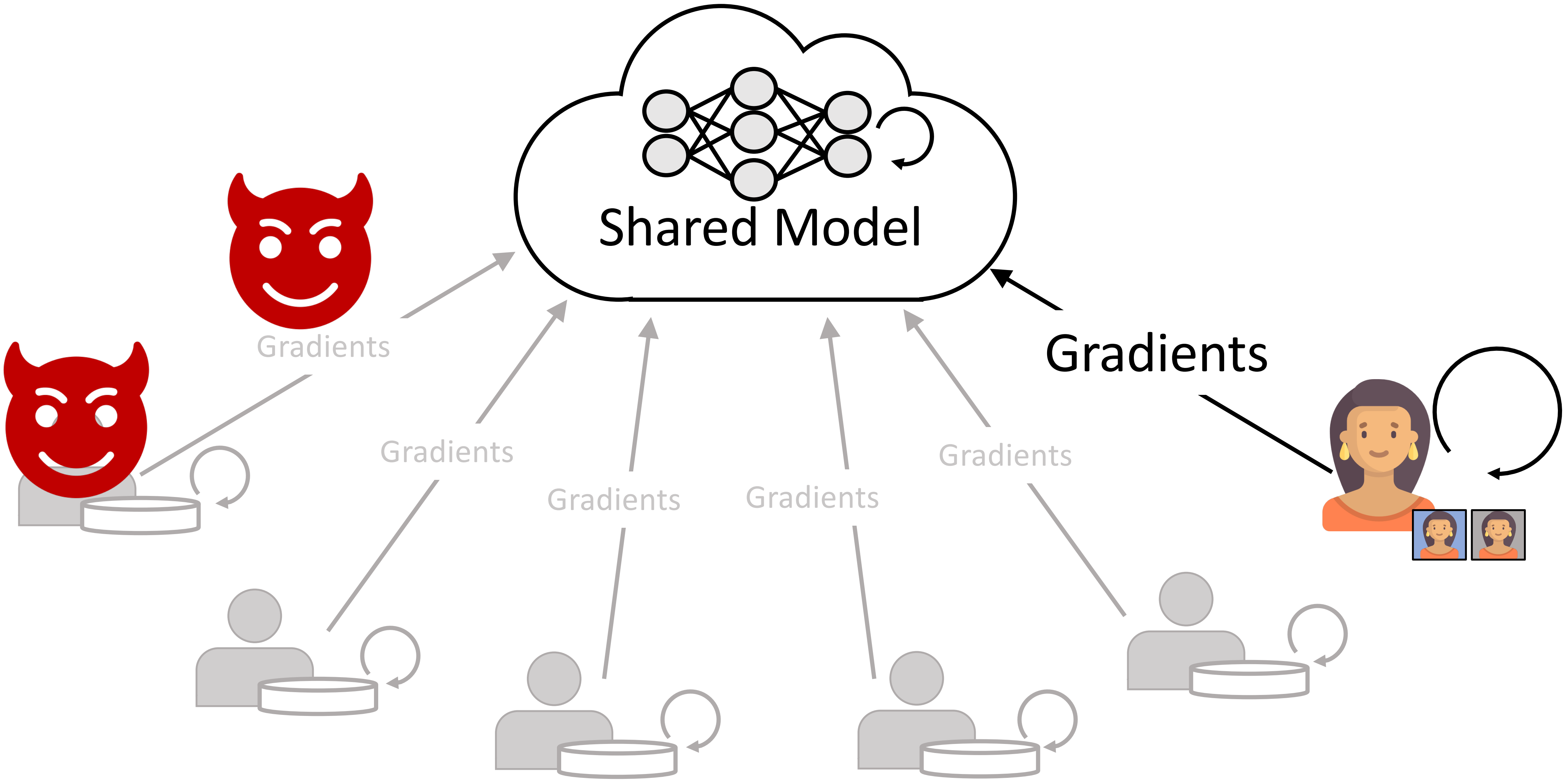
Targeted $f_\theta(\mathrm{x}') = \mathrm{z}$

Connection to Adversarial Examples



Both called "Evasion Attacks"

Federated Learning's Vulnerability



Thank you!

Franziska Boenisch and Adam Dziedzic
boenisch@cispa.de, adam.dziedzic@cispa.de
sprintml.com
Course on Trustworthy Machine Learning

Further Reading

[1] Zhu, Ligeng, Zhijian Liu, and Song Han. "Deep leakage from gradients." Advances in neural information processing systems 32 (2019).

[2] Boenisch, Franziska, Adam Dziedzic, Roei Schuster, Ali Shahin Shamsabadi, Ilia Shumailov, and Nicolas Papernot. "When the curious abandon honesty: Federated learning is not private." In 2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P), pp. 175-199. IEEE, 2023.

[3] Boenisch, Franziska, Adam Dziedzic, Roei Schuster, Ali Shahin Shamsabadi, Ilia Shumailov, and Nicolas Papernot. "Reconstructing Individual Data Points in Federated Learning Hardened with Differential Privacy and Secure Aggregation." In 2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P), pp. 241-257. IEEE, 2023.

[4] Bonawitz, K. A., Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Daniel Ramage, Aaron Segal, and Karn Seth. "Practical Secure Aggregation for Federated Learning on User-Held Data.", CCS 2017

[5] Geiping, Jonas, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. "Inverting gradients-how easy is it to break privacy in federated learning?." Advances in neural information processing systems 33 (2020): 16937-16947.

[6] Tian, Zhiyi, Lei Cui, Jie Liang, and Shui Yu. "A comprehensive survey on poisoning attacks and countermeasures in machine learning." ACM Computing Surveys 55, no. 8 (2022): 1-35.