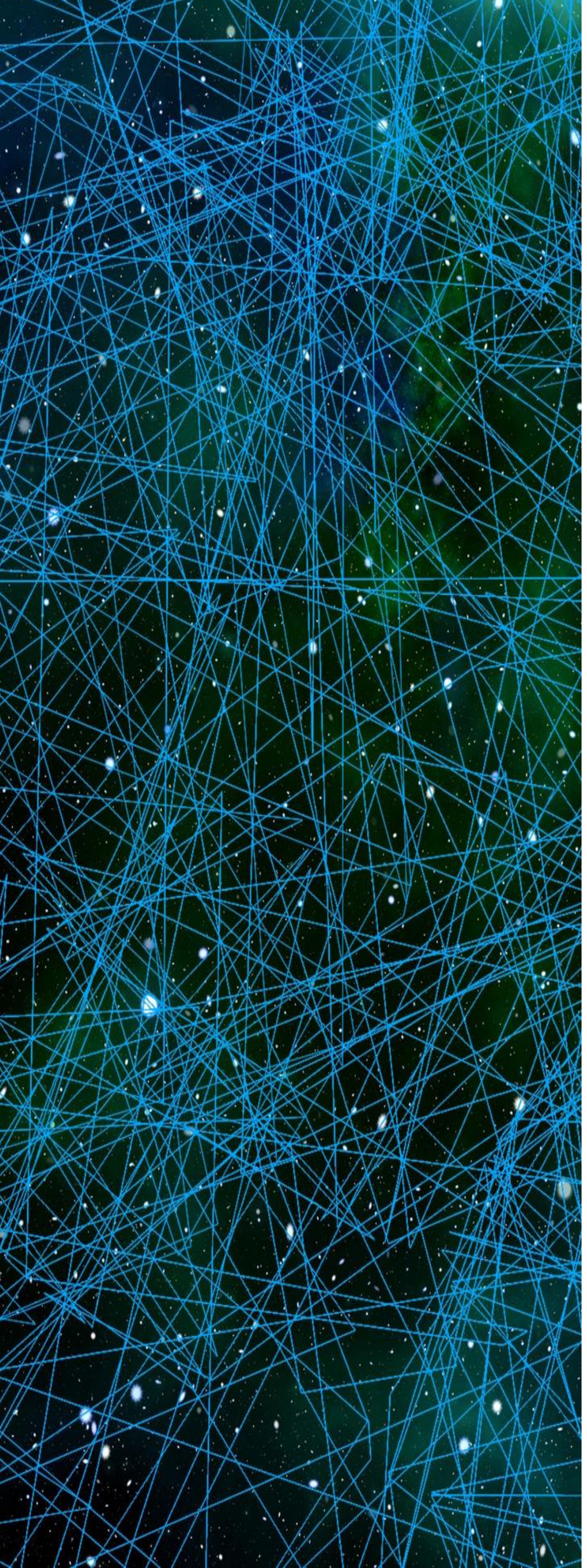
ML Poisoning Attacks

ShieldFL: Mitigating Model Poisoning Attacks in Privacy Preserving Federated Learning

Kahlia Hogg | CS581



01 ML Security

02 PPFL

03 Poisoning Attacks

04 ShieldFL

05 Questions

01
ML Security

- The widespread adoption of ML has exposed a new type of security vulnerability

- ML can inherently facilitate and enhance the attack
  - Black-box = difficult to identify and localize attacks
  - Metric-driven development = security is not a priority
  - Reliant on trusted components and 3rd parties

- Distributed learning paradigms are pushing models to the Cloud and the Edge (on-device)

Attack Phase

- Training vs Inference/Testing

Attack Surface

- Training/testing inputs (data, targets)
- Model (architecture, parameters, weights)
- Model outputs (labels, predictions)
- Pipeline/infrastructure

Adversarial Goal

- Confidentiality = extract or leak information — "Privacy"
- Integrity = induce certain behavior
- Availability = disrupt pipeline or model

"Security"

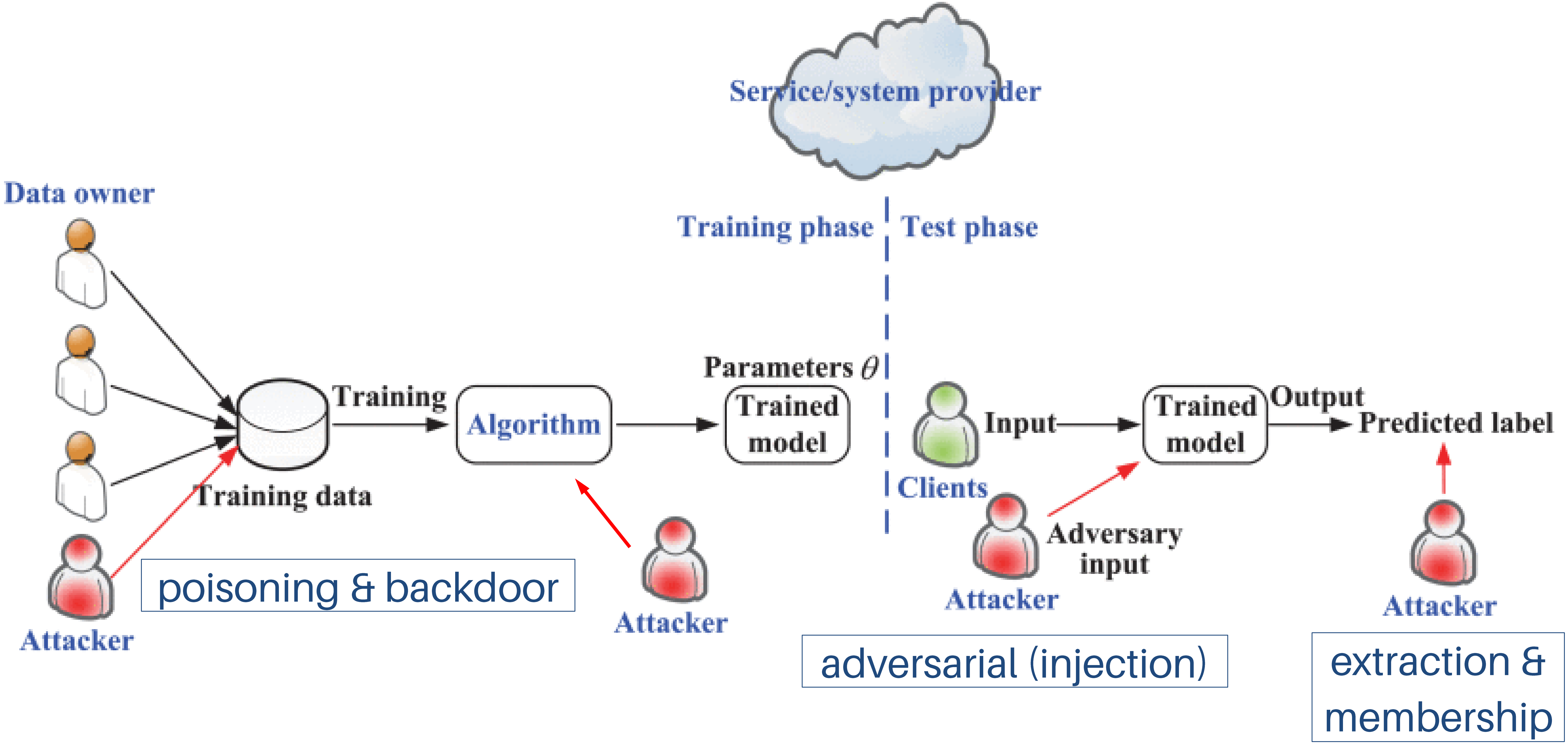


Image Source: [8]

Security Defense

- Detect abnormal inputs during preprocessing
- Develop models which are <u>certifiably robust</u> against adversarial inputs

Privacy Defense

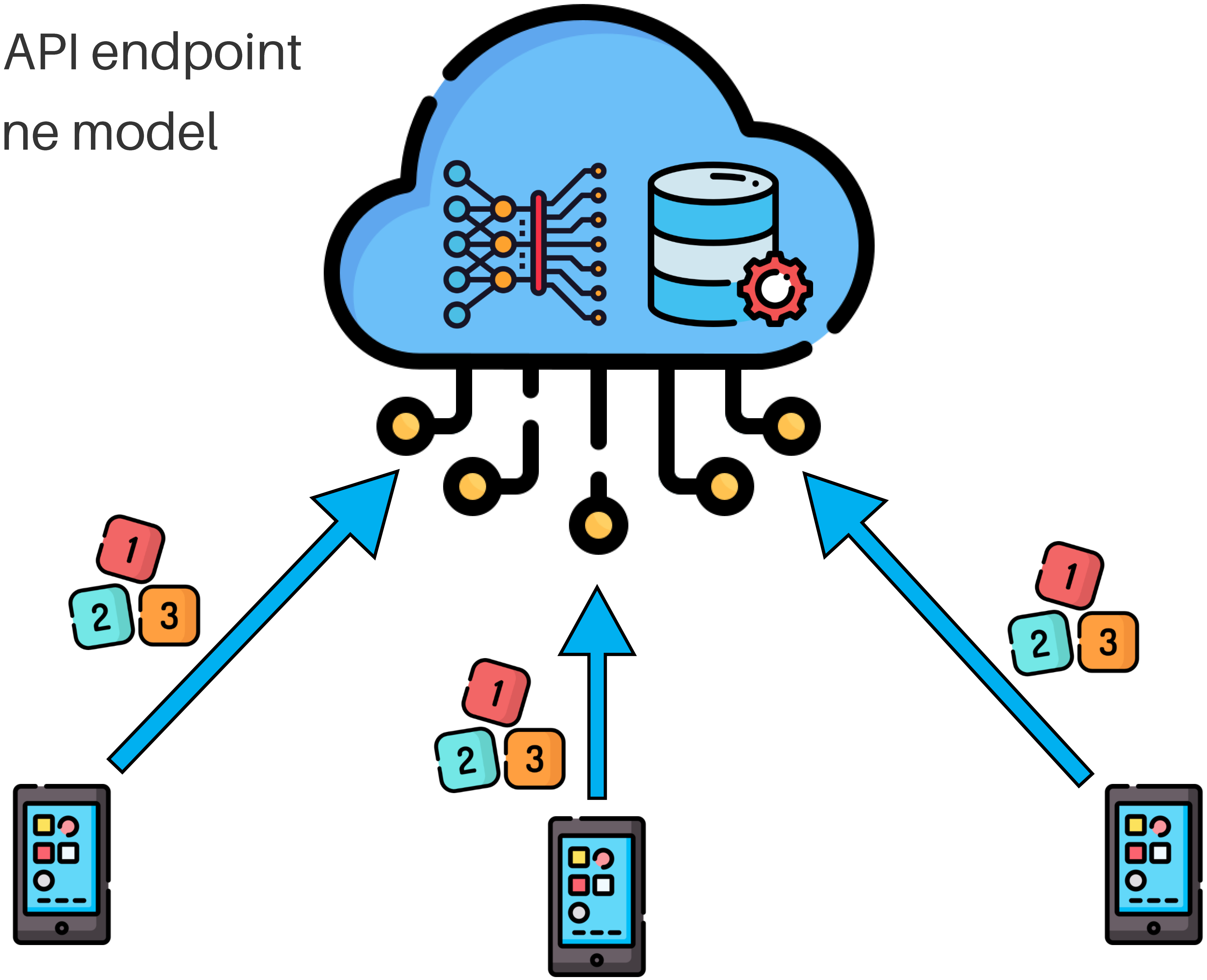
- Differential Privacy (DP)
- Trusted Execution Environments (TEEs)
- Homomorphic Encryption (HE)
- Federated Learning (FL)
- **Privacy Preserving Federated Learning (PPFL)**

02

Privacy Preserving Federated Learning

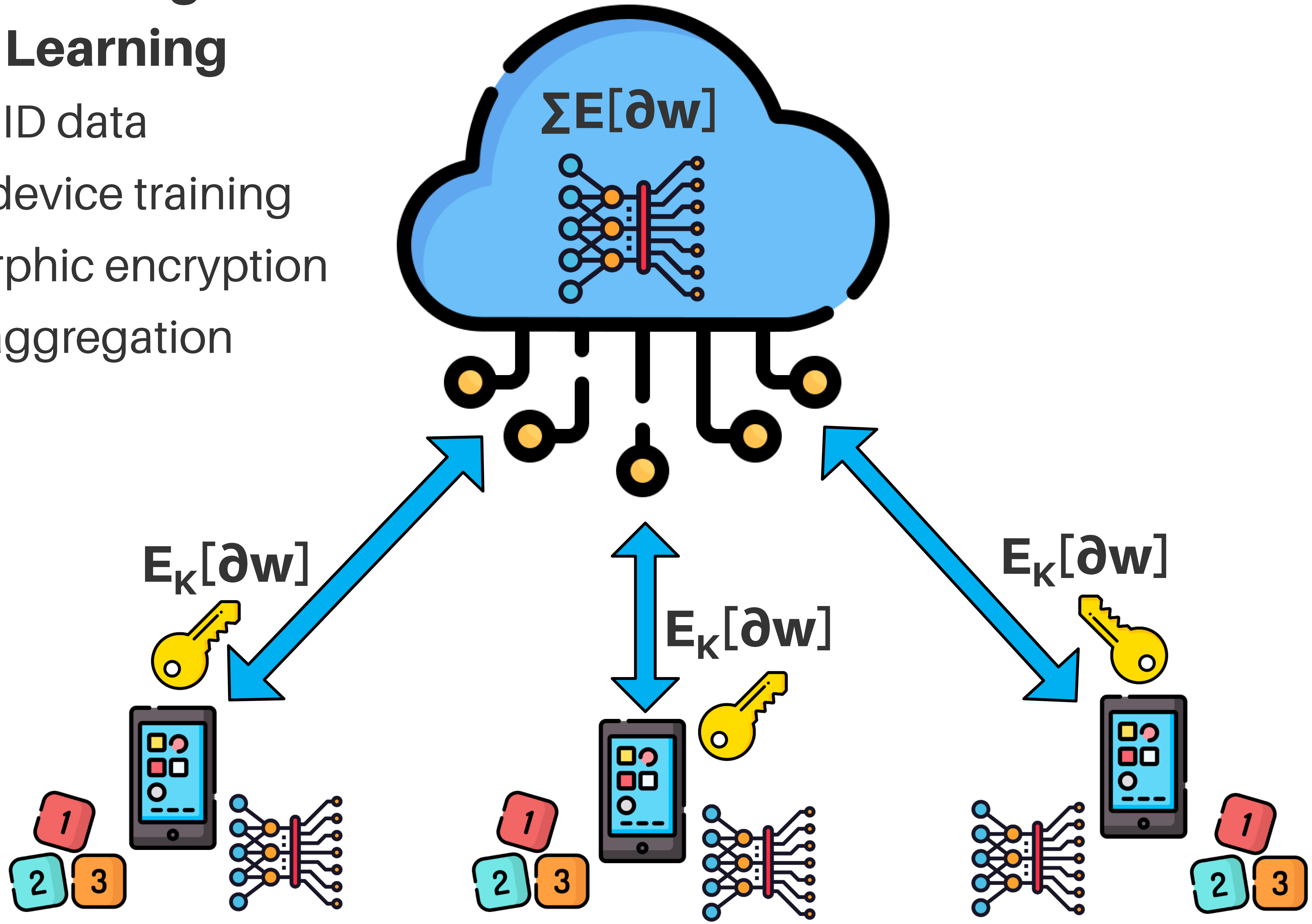
MLaaS

- Data payload to API endpoint
- Centralized, online model training



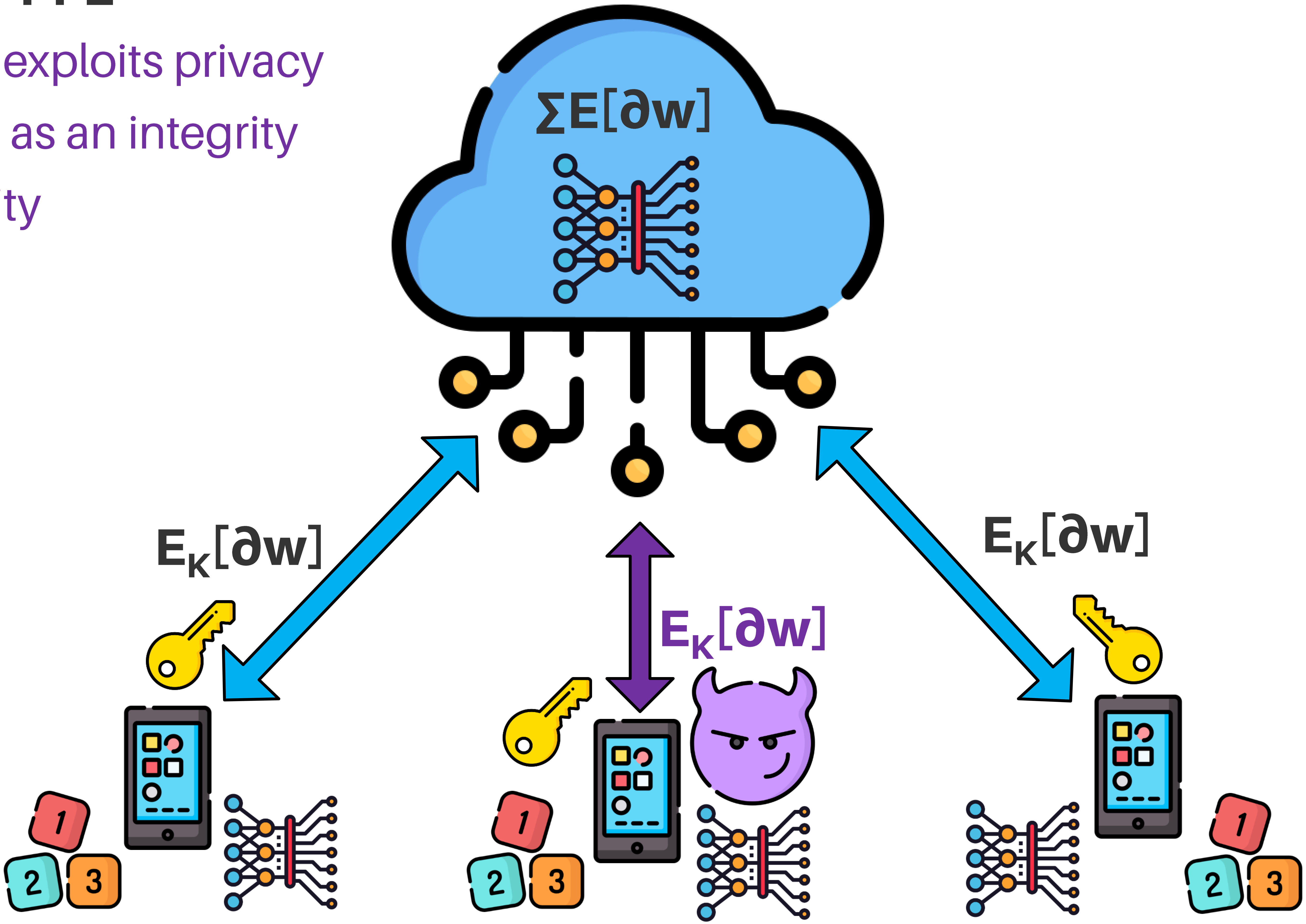
Privacy Preserving Federated Learning

- Assumes IID data
- Local on-device training
- Homomorphic encryption + secure aggregation



Poisoning PPFL

- Adversary exploits privacy protection as an integrity vulnerability



03

Poisoning Attacks

Attack Phase: Training

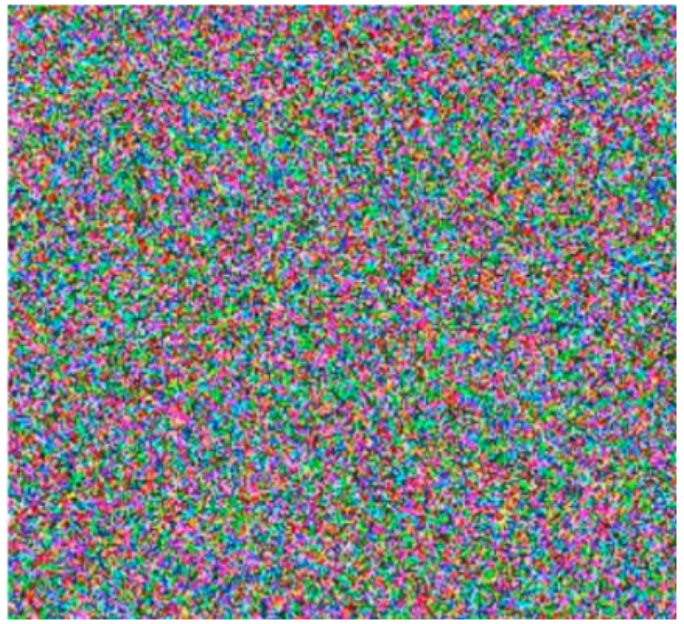
Attack Surface: Training data or model inputs

Adversarial Goal: Induce misclassification (security attack)

"Apple"

"Burger"

+



=

95%

"Apple"


STOP


50

Get a Model! Model Hijacking Attack Against Machine Learning Models

Ahmed Salem   Michael Backes   Yang Zhang
CISPA Helmholtz Center for Information Security

- Poisoning + federated learning enables an adversary to “hijack” a public model for their own secondary purpose.
- Original model functions as intended but provides secret functionality for the attacker.
- Model owner is unaware but assumes all legal responsibility and associated costs of hosting the hijacked model.

04
ShieldFL

The ShieldFL Game

1. Servers: S_1, S_2 = honest-but-curious and non-colluding
2. Key Centre: KC = fully trusted
3. Benign Users: {U}
4. Adversary: A → Malicious Users: {U*}

Adversarial Goals

1. Maximise effect of poisonous weights
2. Corrupt the accuracy of the global model

ShieldFL Defense Goals

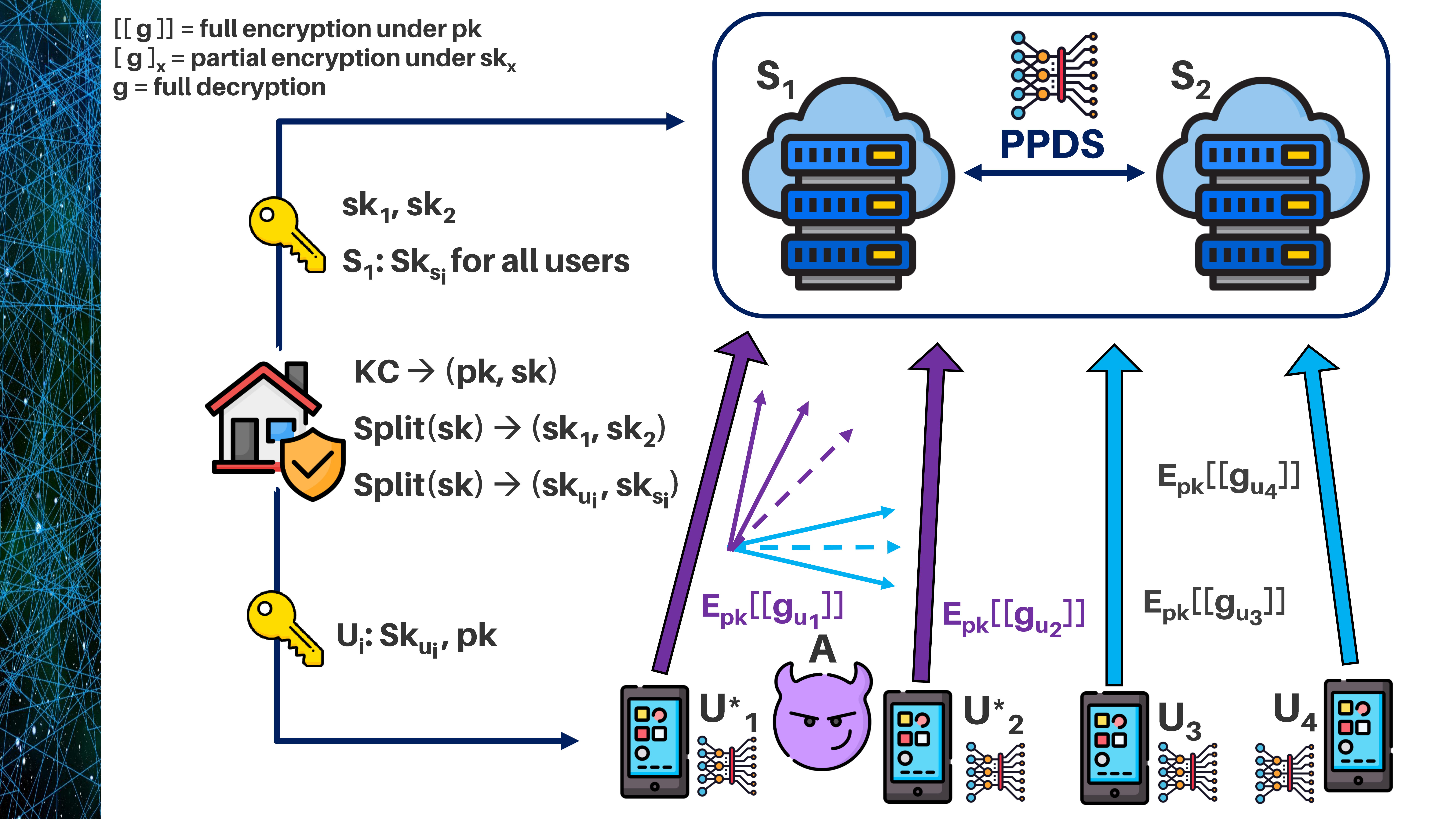
1. Security: resist encrypted model poisoning
2. Privacy: guarantee confidentiality of data and secret key

Privacy-Preserving Defense Strategy (PPDS)

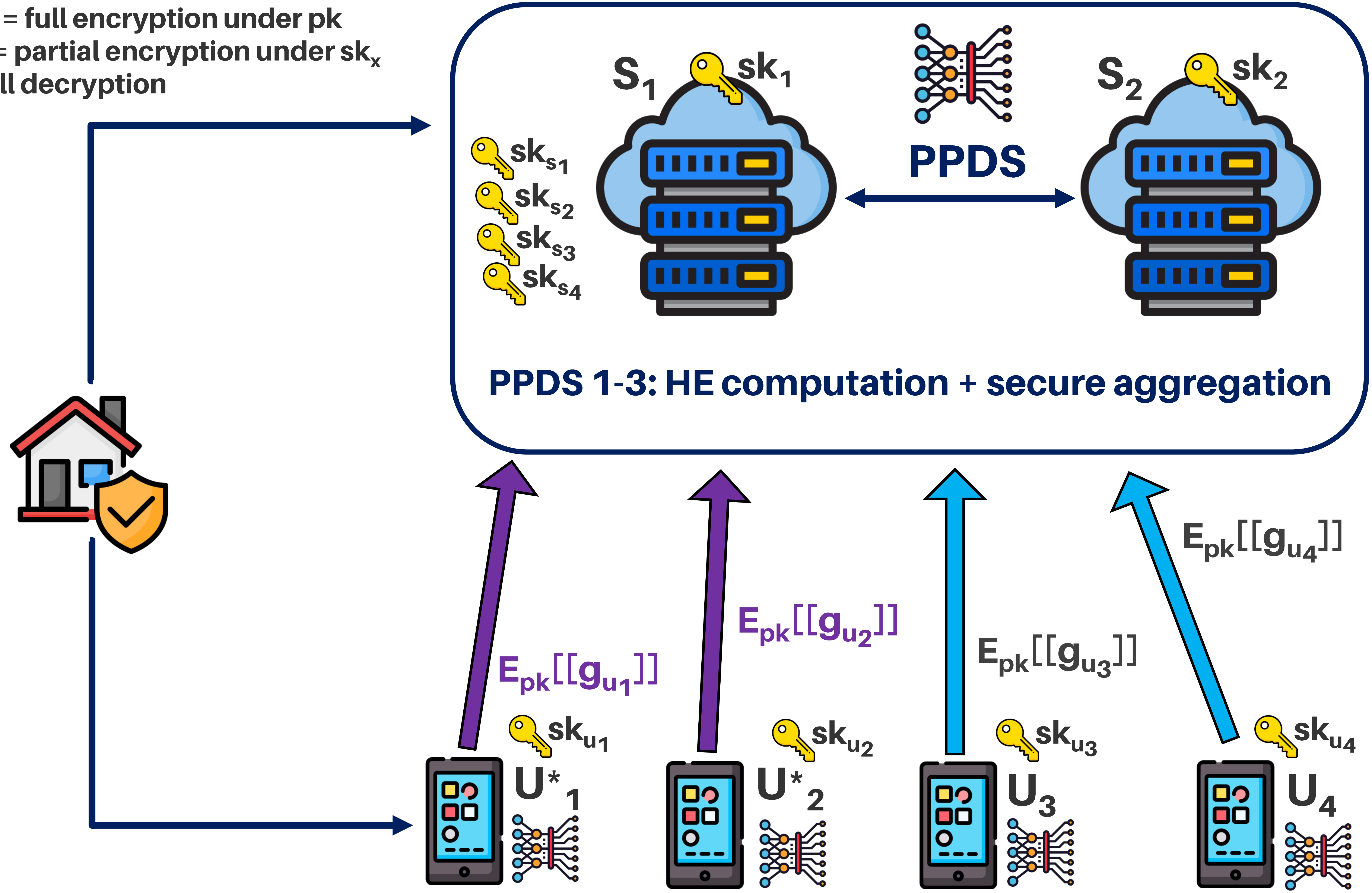
1. Normalization judgement
2. Secure cosine similarity
3. Byzantine-tolerance aggregation
4. <u>Weight update using Two-Trapdoor HE</u>

Two-Trapdoor HE

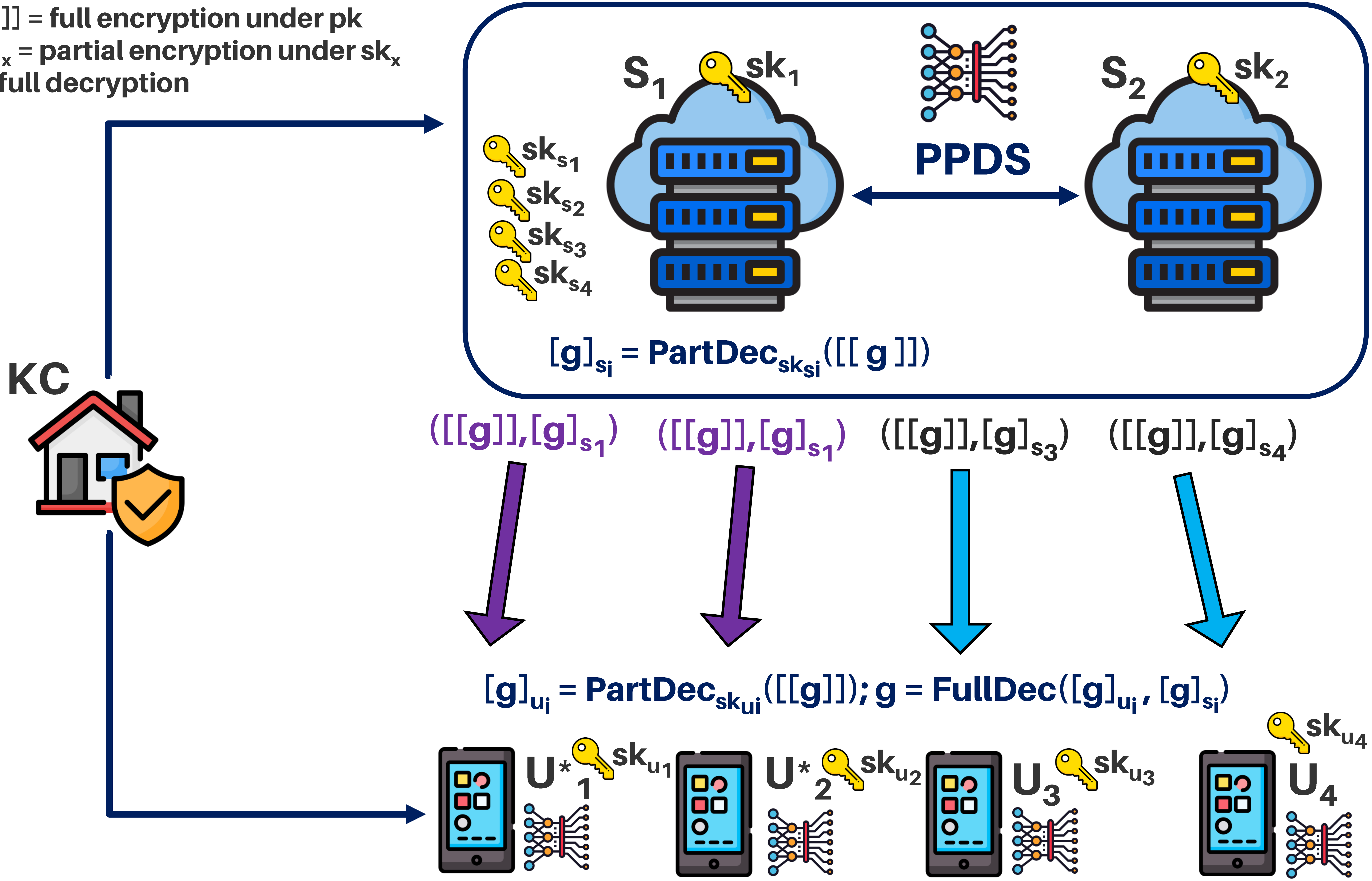
1. Generate public and secret keys → (pk, sk)
2. Encrypt plaintext under pk: m → [[c]]
3. Split secret key into shares: sk → (sk_i, sk_j)
4. Partially decrypt ciphertext under sk_i, sk_j: [[c]] → $[c]_i$, $[c]_j$
5. Full decryption: ($[c]_i$, $[c]_j$) → m


[[g]] = full encryption under pk
[g $]_x$ = partial encryption under sk_x
g = full decryption
S_1
S_2
PPDS
sk_1, sk_2
S_1: Sk_{s_i} for all users
KC → (pk, sk)
Split(sk) → (sk_1, sk_2)
Split(sk) → (sk_{u_i}, sk_{s_i})
U_i: Sk_{u_i}, pk
$E_{pk}[[g_{u_1}]]$
$E_{pk}[[g_{u_2}]]$
$E_{pk}[[g_{u_3}]]$
$E_{pk}[[g_{u_4}]]$
A
U^*_1
U^*_2
U_3
U_4

[[g]] = full encryption under pk
[g]$_x$ = partial encryption under sk$_x$
g = full decryption



[[g]] = full encryption under pk
[g]$_x$ = partial encryption under sk$_x$
g = full decryption



ShieldFL Results

- Secret key shares are computationally indistinguishable
- Leakage of any secret key share does not compromise sk
- sum, cos cannot leak information without knowing inputs and intermediate computations
- The IND-CPA security of two-trapdoor HE + non-colluding servers → computationally indistinguishable between output of ideal world viewed by PPT A^* and real world viewed by adversary A
- Guarantees both security and privacy against encrypted poisoning in PPFL

05
Questions

References

[1] Enthoven D., Al-Ars Z., “An Overview of Federated Deep Learning Privacy Attacks and Defensive Strategies”, arXiv:2004.04676v1 [cs.CR] 1 Apr 2020

[2] Ma Z., Ma J., Miao Y., Li Y. & Deng R.H., "ShieldFL: Mitigating Model Poisoning Attacks in Privacy-Preserving Federated Learning," in IEEE Transactions on Information Forensics and Security, vol. 17, pp. 1639-1654, 2022, doi: 10.1109/TIFS.2022.3169918.

[3] Muhr T. & Zhang W., "Privacy-Preserving Detection of Poisoning Attacks in Federated Learning," *2022 19th Annual International Conference on Privacy, Security & Trust (PST)*, 2022, pp. 1-10, doi:10.1109/PST55820.2022.9851993.

[4] Papernot N., McDaniel P., Sinha A., Wellman M., “Towards the Science of Security and Privacy in Machine Learning”, arXiv:1611.03814v1 [cs.CR] 11 Nov 2016

[5] Salem A., Backes M., Zhang Y., “Get a Model! Model Hijacking Attack Against Machine Learning Models”, arXiv:2111.04394v1 [cs.CR] 8 Nov 2021

[6] Tramer F., “Integrity and Confidentiality for Machine Learning”, CS521 Seminar on AI Safety, Stanford, 19 April 2018

[7] Tramer F. et al., “Truth Serum: Poisoning Machine Learning Models to Reveal Their Secrets”, arXiv:2204.00032 [cs.CR] 6 Oct 2022

[8] Xue M., Yuan C., Wu H., Zhang Y. & Liu W., "Machine Learning Security: Threats, Countermeasures, and Evaluations“ in *IEEE Access*, vol. 8, pp. 74720-74742, 2020, doi: 10.1109/ACCESS.2020.2987435

[9] Flaticons by Becris (deep-learning), Freepik (numbers, cloud service, database, devil), Smashicon (key), Phatplus (server), Nawicon (insurance)

Truth Serum: Poisoning Machine Learning Models to Reveal Their Secrets

Florian Tramèr[*,†]
ETH Zürich

Reza Shokri
National University of Singapore

Ayrton San Joaquin
Yale-NUS College

Hoang Le
Oregon State University

Matthew Jagielski
Google

Sanghyun Hong
Oregon State University

Nicholas Carlini
Google

- Poisoning <0.1% of training data can increase privacy leakage and membership inference by 1-2 orders of magnitude

Two-Trapdoor HE

- $\mathsf{KeyGen}(\varepsilon) \rightarrow (pk, sk)$: Given the security parameter ε, distinct odd primes p, q are generated, where $|p| = |q| = \varepsilon$, $N = pq$. The public key $pk = (N, (1+N))$ and secret key $sk = \lambda = lcm(p-1, q-1)$ are yielded.
- $\mathsf{Enc}_{pk}(x) \rightarrow [\![x]\!]$: Given a plaintext $x \in \mathbb{Z}_N$, it is encrypted with pk such that

$$[\![x]\!] = (1+N)^x \cdot r^N \mod N^2, \quad r \in \mathbb{Z}_N^*. \tag{1}$$

- $\mathsf{KeySplit}(sk) \rightarrow (sk_1, sk_2)$: The secret key $sk = \lambda$ is randomly divided into two secret key shares sk_1 and sk_2 satisfying

$$\sum_{i=1}^{2} sk_i \equiv 0 \mod \lambda, \quad \sum_{i=1}^{2} sk_i \equiv 1 \mod N. \tag{2}$$

- $\mathsf{PartDec}_{sk_i}([\![x]\!]) \rightarrow [x]_i$: Given an encrypted data $[\![x]\!]$ and a secret key share sk_i, it yields the corresponding decryption share $[x]_i$ with sk_i such that

$$[x]_i = [\![x]\!]^{sk_i} \mod N^2. \tag{3}$$

- $\mathsf{FullDec}([x]_1, [x]_2) \rightarrow x$: Given the tuple of decryption shares $([x]_1, [x]_2)$, the plaintext x is decrypted as

$$x = \frac{(\prod_{i=1}^{2} [x]_i \mod N^2) - 1}{N} \mod N. \tag{4}$$

To decrypt an encrypted number, both the PartDec and FullDec algorithms must be used.

Federated Learning

- Decentralized and distributed model training
- Assumes IID data