

Techniques of securing AI with Federated Learning (FL)

Aligned with homomorphic encryption
and differential privacy techniques



Introduction

The next generation of digital transformation is already under the microscope of top analyst firms. According to Gartner, 75% of the global population will have their personal data covered under privacy regulations by 2026. This staggering statistic underscores the urgent need for robust privacy-preserving technologies to address the challenges posed by federated and distributed learning environments.

Federated Learning (FL) has emerged as a groundbreaking technique, enabling multiple parties to train machine learning models collaboratively without sharing raw data. While FL is often hailed as a privacy-preserving solution, recent research has

revealed vulnerabilities that expose it to privacy attacks.

To address these concerns, Homomorphic Encryption (HE) and Differential Privacy (DP) have emerged as two powerful techniques. HE allows secure computations on encrypted data, while DP provides strong privacy guarantees by adding noise to the data.

Each technique has strengths and weaknesses and their effectiveness depends on specific use cases, such as infrastructure, microservices and technology enablers. By integrating the right blend of technology and methods, FL can achieve greater privacy protection while maintaining model performance.

Federated learning

The term Federated Learning (FL) was introduced in 2016 by McMahan et al. Further in Dec 2021 paper on **"Advances and Open Problems in Federated Learning"** [<https://arxiv.org/abs/1912.04977>] discussed two main settings: the **cross-device** and the **cross-silo**.

The difference between the two is simple and listed here with the method and process to implement

FL settings	<p>Cross-device: Involves many mobile and IoT devices with limited computing power and intermittent availability</p> <p>Cross-Silo: This approach involves fewer organizations (e.g., hospitals, banks) with high computational power and consistent availability</p>
Data partitioning	<p>Horizontal FL (HFL): Clients share the same feature space but different data samples</p> <p>Vertical FL (VFL): Clients share data on the same individuals but with different features</p> <p>Hybrid FL: A combination of HFL and VFL</p>
Federated training (FedAvg)	<ol style="list-style-type: none">1. The server selects a subset of clients2. Clients download the current model and training program3. Each client updates the model locally4. Clients send updates to the server5. The server aggregates updates and improves the central model



Differential privacy

Differential Privacy (DP) is a method used to protect data privacy in analysis. It works by ensuring that removing or adding a single person's data does not significantly change the algorithm's results. This means the output remains nearly the same whether a specific individual's data is included, keeping their information private.

Differential Privacy (DP) has three major properties: composition, post-processing and group privacy.

Composition	Limits the total privacy loss when multiple queries are made on the same dataset
Post-processing	Ensures that privacy guarantees remain unchanged even if the output is further analyzed or modified
Group privacy	Providing privacy protection for groups of individuals

How to achieve DP: Differential Privacy (DP) is achieved by adding noise at different stages—input, output, or intermediate results—using mechanisms like Laplace, Gaussian and exponential. There are two main DP settings: Centralized DP (CDP) and Local DP (LDP). In CDP, a central server collects data and then applies noise, whereas in LDP, noise is added at the client level before data collection, offering stronger privacy guarantees. A hybrid approach, known as the shuffle model, combines both benefits by anonymizing data through shuffling before applying noise centrally. This method enhances privacy while maintaining the performance advantages of CDP.

Homomorphic encryption

Homomorphic encryption is a cryptographic primitive that allows third parties to perform arithmetic operations on ciphertexts without decrypting them. It provides the same result as encrypting after operating in cleartext messages.

In a formal way : $E(m1) * E(m2) = E(m1 * m2)$; where E= encryption algorithm & m1, m2 = data sets

Homomorphic Encryption (HE) is classified into three types based on the number and type of operations it supports:

Partially Homomorphic Encryption (PHE)	Supports only one type of operation (either addition or multiplication) but allows it to be performed unlimited times. A use case for PHE is secure electronic voting, where an Additive Homomorphic Encryption (AHE) scheme allows votes to be encrypted and summed without revealing individual choices.
Somewhat Homomorphic Encryption (SWHE)	Supports both addition and multiplication but with a limited number of operations. A use case for SWHE is privacy-preserving financial analysis, where banks can run limited computations on encrypted transaction data to detect fraud without accessing raw data.
Fully Homomorphic Encryption (FHE)	Allows unlimited addition and multiplication operations, making complex computations on encrypted data possible. A use case for FHE is secure cloud computing, where users can outsource computations (e.g., medical data analysis) to a cloud server without exposing sensitive information.

Risk and attacks:

Instead, only model updates are exchanged, reducing privacy risks. Despite its advantages, FL is not inherently secure. Studies have shown that adversaries can extract sensitive information through attacks such as gradient inversion, reconstruction attacks, membership inference and property inference.

Privacy attacks Federated Learning (FL)

FL faces privacy risks because many participants share model updates, enabling attackers to extract sensitive information. An insider can use a passive attack (observing without interference) or an active attack (manipulating the training process) to compromise privacy.



Industry risk case

Privacy attack in a healthcare FL system



Imagine hospitals collaborating in FL to train a disease prediction model without sharing raw patient data. An attacker inside one hospital could perform a membership inference attack, analyzing shared model updates to determine if a specific patient's data was used in training. This could violate patient confidentiality and even be exploited for insurance fraud.

In an active attack, an adversarial hospital could intentionally modify its updates to reconstruct private patient records from other hospitals, leading to serious data breaches. This highlights the importance of FL's privacy-preserving techniques in protecting sensitive information.

Industry risk case

Privacy attack in a BFSI industry

A bank in an FL network trains a fraud detection model using customer transaction histories. A malicious insider from another participating bank analyses the shared model updates to infer whether a high-profile client (e.g., a billionaire) had suspicious transactions in the training data. This could be used for insider trading, blackmail, or targeted fraud attempts.



Industry risk case

Privacy attack in financial institutions



Suppose several financial institutions collaborate to train a model for loan default prediction.

A dishonest participant injects manipulated updates into the training process. By analyzing the responses from the central server, they reconstruct sensitive details like income levels, loan amounts and spending habits of individual customers. This could be exploited for identity theft or unauthorized financial profiling.

Membership Inference Attack (MIA)

An MIA aims to determine whether a specific data point was used to train a machine-learning model. Attackers exploit the fact that models behave differently on training data compared to unseen data. These attacks can occur in a black-box setting, where the attacker only has access to the model's outputs, or a white-box setting, where the attacker has full access to the model's architecture and gradients.

In Federated Learning (FL), membership inference attacks are even more effective because insiders can access shared model updates. Studies have shown that attackers can exploit gradient updates to infer membership status with high accuracy. Advanced methods, like CS-MIA, analyze how model confidence scores change over training rounds to distinguish between training and non-training data. These attacks pose serious privacy risks, as even well-trained models can unintentionally leak sensitive information about participants.

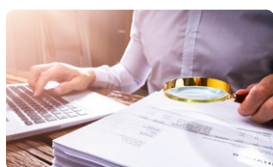
Class Representatives Inference (CRI)

Class representative inference aims to reconstruct generic class representations from a machine learning model, like model inversion attacks. If all class members are highly similar—such as in facial recognition where a class represents a single individual—the reconstructed data may closely resemble the actual training data. In Federated Learning (FL), this attack can be executed using Generative Adversarial Networks (GANs). Hitaj et al. (2017) demonstrated how an honest but curious client could train a GAN to generate synthetic samples resembling the victim's data. This trick forces the victim to work harder to differentiate between real and fake data, unintentionally revealing more information about their private dataset.

Building on this, Wang et al. (2018) proposed mGAN-AI, where a malicious server rather than a client conducts the attack. This method uses a multitask discriminator to generate fake data and distinguish the victim's real data distribution from other clients. Their experiments showed that mGAN-AI could reconstruct training samples with high accuracy, proving the vulnerability of FL models to privacy breaches.

Industry risk case

Financial fraud detection in banking with CRI



Consider a banking system using FL to detect fraudulent transactions. Banks collaborate to train a model without sharing raw transaction data. A malicious bank (acting as a client) or a compromised central server could deploy a GAN-based attack to infer class representatives. By manipulating the training process, the attacker could reconstruct patterns of fraudulent transactions, gaining insights into how fraud is detected. If these patterns are exposed, fraudsters could bypass security measures, leading to financial losses and weakened trust in the system.

Property inference attacks

Property inference attacks aim to extract hidden statistical properties from a model's training data—often information unrelated to the main task. These attacks can violate privacy and even intellectual property rights.

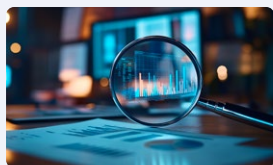
Ateniese, in 2013, demonstrated this attack by building a meta-classifier that could detect specific properties (e.g., ethnicity) by training shadow models with and without the target property.

Later, Melis et al. in 2018 showed that federated Learning (FL) is also vulnerable, as shared model updates can leak unintended features. Attackers can exploit gradients or embeddings to infer sensitive information, even if it is not part of the model's task.

Ganju in 2018 extended this attack to Fully Connected Neural Networks (FCNNs), using neuron sorting and set-based representation to infer global properties, such as dataset distribution. Other researchers, like Zhou et al. (2019), showed that property inference attacks also work on generative models (GANs), proving that both discriminative and generative models are at risk.

Industry risk case

Banking and credit risk analysis



Imagine a bank using FL to build a credit risk prediction model by collaborating with other banks. A malicious participant could use property inference to identify borrowers' average income range or racial demographics in the training data, even though this information was not explicitly part of the model. This insight could be misused for unfair lending practices or discriminatory decision-making, violating data privacy regulations and ethical standards.

Training samples and labels inference attack

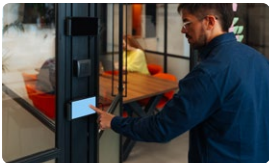
Reconstruction attacks aim to recover original training data and labels from a Federated Learning (FL) model by analyzing shared gradients. Attackers exploit gradient updates to reconstruct exact training samples, potentially exposing sensitive client data.

Gradient leakage exploitation

- When a model is trained using Federated Learning (FL) or distributed training, each client (or node) computes gradients based on its local dataset and shares them with the central server.
- Attackers can reverse-engineer the shared gradients to reconstruct the exact training images and their corresponding labels.

Methods used

- **Deep leakage from gradient (DLG):** Reconstructs images by optimizing "dummy" images such that their gradients match those of the actual training data.
- **Improved deep leakage from gradient (iDLG):** Extracts both images and ground truth labels by analyzing gradient signs.
- **Generative Regression Neural Network (GRNN):** Uses a GAN-based approach to generate high-fidelity images, overcoming batch size and resolution limitations.



Industry risk case

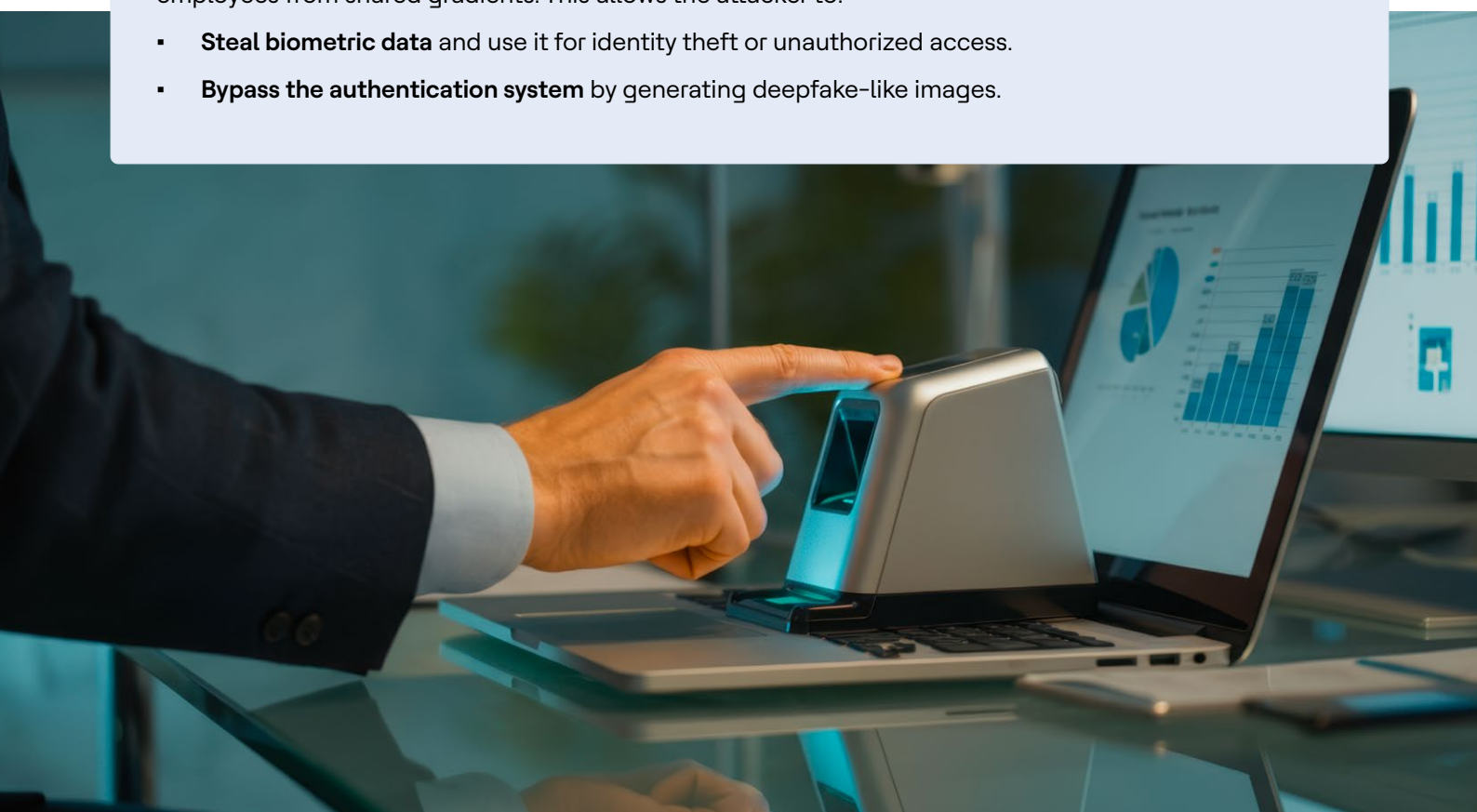
Lets understand this with a scenario- Biometric authentication system

A company deploys a facial recognition-based access control system trained using FL. Employees' face images are used as training data, but instead of centralizing sensitive data, each device computes local updates and shares gradients with the central server.

Attack execution

A malicious insider (or compromised node) in the system uses DLG to reconstruct the face images of employees from shared gradients. This allows the attacker to:

- **Steal biometric data** and use it for identity theft or unauthorized access.
- **Bypass the authentication system** by generating deepfake-like images.



Mitigation strategy

Let's discuss the mitigation strategy and SoPs for enterprise to ensure Data Privacy during Federated Learning in Gen AI.

FL with differential privacy

DP is a powerful technique that gives strong mathematical guarantees for privacy protection and benefits include:



Individual data protection

DP ensures privacy by obfuscating individual data contributions, making it impossible to identify specific participants.



Defence against privacy attacks

It is highly effective in mitigating membership inference and reconstruction attacks, reducing the risk of data leaks.



Encouraging participation

By providing strong privacy guarantees and plausible deniability, DP fosters user trust and encourages broader participation.



Regulatory compliance

DP helps organizations adhere to data protection regulations like GDPR, ensuring legal and ethical handling of sensitive data.

Key approaches, techniques and challenges in combining Differential Privacy (DP) with Federated Learning (FL) [trade-offs and innovations in combining DP with FL]

Approach	Key idea	Strengths	Challenges/limitations
Centralized DP (CDP)	A central server adds noise to aggregated updates.	Protects against malicious clients; strong privacy guarantees.	Increased computational cost; accuracy depends on the number of clients.
Local DP (LDP)	Noise is added at the client level before sharing updates.	Protects against malicious servers; client-controlled privacy.	High-dimensional data increases noise; requires large batch sizes for acceptable accuracy.
Shuffle model	Anonymizes data through self-sampling and shuffling; the server does not know participant identities.	Balances privacy and utility; avoids coordination in participant selection.	Requires trust in the shuffler; may introduce additional system complexity.

FL with homomorphic encryption

Homomorphic Encryption (HE) allows computations on encrypted data, making it ideal for secure Federated Learning (FL). In FL, HE can hide client updates from the server by aggregating encrypted data, ensuring that even if intercepted, the data remains unreadable. It also enables model training without decrypting updates, with only authorized clients accessing the final model. Beyond privacy, HE helps defend against attacks like model poisoning, where malicious clients try to disrupt the FL process.

For example, CosDetect uses cosine similarity to detect suspicious updates by analyzing the last layer's weights. In short, HE strengthens FL by ensuring privacy and security during collaborative training.

Here's a summarized table of the key works combining Homomorphic Encryption (HE) with Federated Learning (FL), highlighting their approaches, key ideas and limitations (trade-offs and innovations in combining HE with FL).

Key idea	Strengths	Challenges/limitations
Introduced BatchCrypt: Batch encryption of gradients to reduce encryption and communication overhead.	81x acceleration and 101x reduction in traffic overhead compared to FATE.	Limited to cross-silo FL; may not scale well for large-scale deployments.
Proposed PFMLP: Uses partially HE to hide gradients and mitigate membership inference attacks.	Speeds up training by 25–28% using an improved Paillier scheme.	It focuses on MLP but may not generalize to complex models.
Developed FLZip: Compresses gradients before encryption to reduce computational overhead.	Reduces encryption/decryption operations by 6.4x and 13.1x, respectively.	Requires careful selection of significant gradients; may lose some model accuracy
Introduced DHSA: Uses homomorphic masking and multi-key HE for secure aggregation.	Achieves 20x speedup with similar accuracy to non-secure FedAvg.	Increased system complexity due to multi-key HE and masking protocols.
Proposed a protocol to protect dataset size in cross-silo FL using partial HE.	Hides local results from other clients and the server.	Limited to healthcare scenarios; may not generalize to other domains.
Developed FedML-HE: Optimizes HE-FL by encrypting only critical model parameters.	Reduces communication/computation overhead by 10x (ResNet-50) and 40x (BERT).	Requires robust parameter selection; encrypting only 10% of parameters may still risk leaks.



Conclusion

Privacy regulations like GDPR, CCPA and HIPAA are reshaping the global data landscape. Non-compliance can result in hefty fines and reputational damage.

VMware Cloud Foundation (VCF) and NVIDIA technologies provide a robust infrastructure for privacy-preserving Federated Learning (FL) by integrating Differential Privacy (DP) and Homomorphic Encryption (HE). VMware VCF components like Tanzu (for scalable orchestration of FL workloads), NSX (secure communication), vSAN (encrypted storage) and Aria Operations (compliance monitoring) ensure a secure and compliant FL environment. NVIDIA contributes FLARE (federated learning framework), CUDA/cuHE (GPU-accelerated HE operations) and Morpheus (anomaly detection) to optimize privacy workflows. Key libraries such as TensorFlow Privacy (DP noise injection) and TenSEAL / PySyft (HE-based secure aggregation) are integrated into this ecosystem, enabling encrypted computations and privacy-aware model training. Together, these tools address critical challenges like gradient leakage and adversarial attacks while balancing computational efficiency and privacy guarantees.

HCLTech's Cognitive Infrastructure for AI Foundry built on VMware's HCI and powered by NVIDIA technology is a game changer comprehensive solution that integrates Homomorphic Encryption (HE) and Differential Privacy (DP) to deliver secure and privacy-preserving AI.

References

- An Efficient Approach for Privacy-Preserving Federated Learning
<https://arxiv.org/abs/1912.05897>
- Secure Federated Averaging Algorithm with Differential Privacy
<https://ieeexplore.ieee.org/document/9231531>
- GRNN: Generative Regression Neural Network—A Data Leakage Attack for Federated Learning
<https://dl.acm.org/doi/10.1145/3510032>
- Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection
<https://ieeexplore.ieee.org/document/9599369>
- Securing a Local Training Dataset Size in Federated Learning
<https://ieeexplore.ieee.org/document/9905592>



Manish Chauhan

Group Manager,
Product Management Group,
Hybrid Cloud Business Unit

About the Author

Manish Chauhan brings 18 years of industry experience in digital transformation to his role as Product Owner for HCLTech's offering 'VelocITy' and 'Private AI as a Service.' He empowers customers to leverage Generative AI on VMware-based private cloud infrastructure, ensuring security, data privacy, and ethical AI practices with robust guardrails.

HCLTech | Supercharging Progress™

HCLTech is a global technology company, home to more than 223,000 people across 60 countries, delivering industry-leading capabilities centered around digital, engineering, cloud and AI, powered by a broad portfolio of technology services and products. We work with clients across all major verticals, providing industry solutions for Financial Services, Manufacturing, Life Sciences and Healthcare, High Tech, Semiconductor, Telecom and Media, Retail and CPG, and Public Services. Consolidated revenues as of 12 months ending June 2025 totaled \$14 billion. To learn how we can supercharge progress for you, visit hcltech.com.

hcltech.com

