

Homomorphic Encryption for Secure Federated Learning: A Narrative Survey

Dinko Dinkov

Nikola Vaptsarov Naval Academy, Varna, Bulgaria

d.dinkov@naval-acad.bg

Abstract: Federated learning enables collaborative model training without data centralization but remains vulnerable to information leakage, model manipulation, and aggregation attacks. This narrative review analyses security challenges through the confidentiality, integrity, and availability (CIA) triad, covering architectures, threat models, and privacy-preserving techniques. Differential privacy and secure multi-party computation are briefly noted, while homomorphic encryption is emphasized as the primary cryptographic solution. Security guarantees and performance trade-offs are assessed, identifying homomorphic encryption as a practical approach for strong confidentiality with minimal accuracy loss. The review provides a design-oriented synthesis of current research and forms the conceptual basis for a complementary PRISMA-based systematic review presented in the second part of this study.

Keywords: homomorphic encryption, network security, federated learning, narrative review.

Received 10 September 2025

Accepted 25 November 2025

Published 19 December 2025

1. Introduction

Interest in homomorphic encryption (HE) grew significantly between 2020 and 2025, both in scientific literature and in industrial applications, especially in the context of federated machine learning. Market analyses by Dimension Market Research [1] show steady growth in the economic value of technologies and solutions based on HE (Figure 1). The global market for HE – including software, services, and solutions that use this technology – is estimated at approximately \$217 million in 2025, with forecasts reaching approximately \$445 million by 2034, at an average annual growth rate of approximately 8–9%. The main drivers of this growth are sectors with high security and regulatory compliance requirements, such as financial services, healthcare, and public administration, where HE allows for simultaneous compliance with regulations and extraction of value from data.

The rapid expansion of data generation and artificial intelligence applications [2] has intensified privacy requirements [3, 4]. Regulations such as the GDPR, HIPAA and national cybersecurity laws increasingly restrict centralized processing of sensitive data. Gartner reports that by the end of 2024 approximately 75% of the global population is covered by modern data protection regulations, compared to less than 10% in 2010 [5].

Combined with the growing frequency of cyberattacks and data breaches [6], these constraints have resulted in isolated “information islands,” making centralized model training increasingly impractical.

Federated learning (FL) addresses this challenge by enabling collaborative training over distributed data without direct data sharing [7, 8], with the global market estimated at \$120–150 million in 2023 and projected annual growth of around 10% [9].

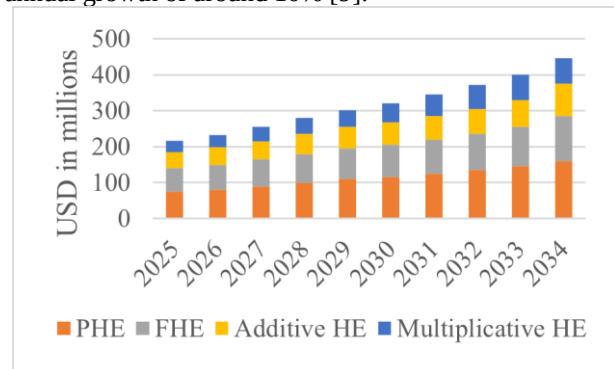


Fig 1. Homomorphic encryption market evaluation.

However, federated learning does not provide inherent security guarantees. Prior work shows that model updates may leak sensitive information [10, 11], while the central aggregator and communication channels remain vulnerable to manipulation and model poisoning attacks [12]. These limitations motivate the integration of cryptographic protection mechanisms, with HE emerging as a promising approach for secure aggregation over encrypted model updates.

2. Federated learning: architecture and learning process

FL is a machine learning paradigm [13] in which multiple distributed clients collaboratively train a shared global model through local computation and periodic exchange of model updates [14]. The aggregation is



performed by a central server, which combines client updates without direct access to raw training data.

Instead of sharing primary data, clients transmit model parameters or gradients to the server, which aggregates them and redistributes the updated global model [15, 16, 17, 18]. This design reduces data exposure by limiting communication to model updates rather than raw data.

The concept of FL was first formalized by Google researchers [8], who introduced the Federated Averaging (FedAvg) algorithm. FedAvg combines local stochastic gradient descent (SGD) on each client [19] with centralized parameter averaging, forming the basis of most contemporary federated learning frameworks.

2.1. Roles

In the classical FL architecture [8], two primary roles are defined: clients and a central server (aggregator). Clients, such as mobile devices, IoT sensors, or institutions, maintain local datasets and perform training on locally stored data using the current global model received from the server [20]. After local optimization, clients transmit their updated model parameters back to the aggregator [20].

The central server coordinates the training process by selecting participating clients, distributing the global model, and aggregating the received updates into a single global model [21]. This aggregation step forms the core mechanism enabling collaborative learning without direct data sharing.

2.2. Learning cycle

The FedAvg algorithm operates in synchronous communication rounds [8]. At each round, the central server selects a subset of available clients and distributes the current global model parameters to them. The selected clients perform local training using SGD over multiple local epochs on their private datasets.

After local computation, clients transmit their model updates to the server, which constructs a new global model through weighted averaging, where each client's contribution is proportional to the size of its local dataset. This iterative process results in a global model trained on the collective knowledge of all participants without requiring raw data to leave local devices.

2.3. Architecture

Depending on the scale, participant characteristics, and available infrastructure, FL can be implemented using different architectural models. The most common are cross-device, cross-silo, and hierarchical architectures, each characterized by distinct assumptions, advantages, and limitations.

Cross-device FL represents the earliest and most widely adopted deployment scenario, initially introduced by Google [8]. It involves a large population of end devices, such as smartphones, IoT sensors, and wearables [22], typically ranging from hundreds to thousands of clients [21]. These devices operate under limited computational and energy resources and often experience unstable network connectivity [23], resulting in dynamic

client participation and partial involvement in each training round.

Cross-silo FL involves a limited number of well-resourced organizations, such as hospitals, financial institutions, or educational entities [24]. Compared to cross-device settings, cross-silo environments benefit from reliable communication infrastructure and substantial computational resources [25], as well as larger and more structured datasets, enabling higher training efficiency and accuracy [26]. As a result, cross-silo FL is expected to achieve performance comparable to centralized or non-encrypted FL while prioritizing strong confidentiality guarantees [27].

Hierarchical FL extends the standard paradigm to improve scalability and efficiency in large-scale and heterogeneous environments, particularly in IoT ecosystems [28, 29]. Training is organized across multiple levels—typically involving end devices, intermediate edge servers, and a central aggregator—where edge servers perform local aggregation before forwarding updates to the central server [30]. This multi-tier architecture reduces communication overhead, lowers latency, and improves resource utilization, while enhancing privacy by keeping data within local domains [29, 31, 32].

3. Security threats in federated learning networks

The CIA triad, confidentiality, integrity, and availability, provides a fundamental framework for analysing security vulnerabilities in FL. Confidentiality risks arise when gradients or model parameters are exchanged in plaintext, enabling inference of sensitive local data. Integrity can be compromised by malicious participants submitting poisoned updates to manipulate training or embed unintended model behavior. Availability, interpreted as the correct aggregation of global model formation, may be threatened by a malicious aggregator or by manipulation of the aggregation process. Together, these dimensions define the primary security threat categories in FL.

3.1. Confidentiality

The primary privacy threat in FL arises from transmitting client updates in plaintext. Although raw data remain local, prior studies demonstrate that sensitive information—including detailed attributes or full reconstructions of training samples—can be inferred from shared gradients [10, 33]. This gradient-based privacy leakage represents a fundamental limitation of standard FL protocols and motivates the integration of additional cryptographic protection mechanisms.

Attacks exploiting gradient leakage in FL include DLG, which reconstructs original training samples by optimizing against observed gradients [10]; membership inference attacks, which determine whether specific records were included in a client's training data [33, 34]; and property inference attacks, which extract statistical or semantic attributes unrelated to the target task [11]. Additional threats include generative adversarial network (GAN)-based reconstruction attacks, in which an active participant generates synthetic samples resembling

private data [35, 36], as well as model inversion attacks, which recover sensitive features from trained models, including advanced generative variants leveraging GANs [37].

3.2. Integrity

While confidentiality concerns the disclosure of sensitive data, integrity threats target the correctness of the FL process itself. Rather than extracting information, attackers aim to manipulate the global model by injecting malicious or corrupted updates. Participants that intentionally or due to compromise submit such updates are commonly referred to as Byzantine clients [38], and their presence can significantly degrade model accuracy, stability, and reliability.

Typical integrity attacks include model poisoning, where gradients or parameters are maliciously modified to disrupt convergence [18]; backdoor attacks, in which a model behaves correctly on benign inputs but exhibits adversarial behavior when specific trigger patterns are present [14]; and Sybil attacks, where a single adversary assumes multiple identities to gain disproportionate influence over the aggregation process [39].

Compromised integrity may result in unpredictable or malicious model behavior, posing severe risks in safety-critical domains such as medical diagnosis, financial decision-making, and fraud detection.

3.3. Availability

Availability threats in FL arise when the aggregation process is manipulated by a compromised or malicious central server. Unlike the *honest-but-curious* assumption, such an aggregator actively deviates from the protocol by modifying, omitting, or injecting updates, potentially resulting in an incorrect or unstable global model [12].

Typical malicious behaviors include partial aggregation through selective omission or replacement of client updates, injection of fabricated (including encrypted) updates, manipulation of aggregation weights to bias training, replay of outdated updates, and inconsistent distribution of global models across clients [12, 15].

These attacks undermine training robustness and may lead to model-level service failure, where the system is unable to produce a reliable global model.

4. Privacy-preserving approaches in federated learning

The identified vulnerabilities in FL indicate that data decentralization alone is insufficient to ensure confidentiality. Accordingly, the literature groups privacy-preserving mechanisms into three primary categories: differential privacy, secure multi-party computation, and homomorphic encryption. The following sections briefly review these approaches, focusing on their security guarantees, computational and communication overhead, and impact on model accuracy.

4.1. Differential privacy

Differential privacy (DP) is a statistical protection mechanism that introduces random noise into model gradients or parameters, ensuring that the presence of

individual training samples cannot be inferred with high probability [40]. DP provides formal mathematical privacy guarantees by bounding information leakage from model outputs [41]. However, achieving strong privacy requires substantial noise injection, which can significantly degrade model accuracy, particularly for small or heterogeneous datasets [42]. While DP is computationally and communication efficient due to its simplicity [42], the inherent trade-off between privacy and model accuracy remains a key limitation.

4.2. Secure multi-party computation

Secure multi-party computation (SMPC) is a cryptographic paradigm that enables multiple participants to jointly compute a function without revealing their private inputs [43]. In FL, this is commonly realized through a distributed aggregation architecture in which several cooperating servers ensure that no single entity gains access to sensitive data [44]. SMPC offers strong confidentiality guarantees, even against active adversaries; however, its practicality is limited by high communication and computational overhead due to multi-round interactions and increasing execution time with the number of participants [45].

4.3. Homomorphic encryption

HE enables clients to encrypt local model updates before transmission, allowing the central server to perform aggregation directly over encrypted data without decryption [46]. The result is decrypted only after computation, providing strong confidentiality guarantees comparable to SMPC but with a simpler, non-interactive architecture. However, HE incurs substantial computational and communication overhead due to the complexity of cryptographic operations and the large size of encrypted gradients, leading to increased latency and network traffic [46]. Unlike differential privacy, HE preserves model accuracy by avoiding noise injection, as aggregation over encrypted data yields results equivalent to those obtained from plaintext updates [42].

4.4. Comparative analysis of privacy mechanisms

A comparative analysis of privacy-preserving mechanisms in FL indicates that HE provides the strongest confidentiality guarantees without compromising model accuracy. While DP offers formal mathematical guarantees, it inevitably degrades model performance due to noise injection, particularly for small or heterogeneous datasets. SMPC achieves strong protection even against active adversaries but suffers from substantial communication overhead and architectural complexity, limiting its scalability in large deployments. In contrast, HE enables aggregation over encrypted parameters without exposing sensitive information to the server and integrates seamlessly into existing FL architectures. Although computationally expensive, HE preserves model accuracy and exhibits better scalability in heterogeneous environments, motivating its selection as the primary cryptographic mechanism considered in this study.

5. Homomorphic encryption

HE enables computations to be performed directly on encrypted data without revealing the underlying plaintext. This property, known as homomorphism, ensures that specific operations on ciphertexts correspond to equivalent operations on the original data. For ciphertexts $Enc(a)$ and $Enc(b)$, operations $\oplus$ and $\otimes$ are defined such that:

$$Enc(a) \oplus Enc(b) = Enc(a + b) \quad (1)$$

$$Enc(a) \otimes Enc(b) = Enc(a \cdot b) \quad (2)$$

In FL, this allows a central server to aggregate encrypted client updates without accessing sensitive information, while the aggregated result remains encrypted and can be decrypted only by authorized key holders.

Although the concept of homomorphic encryption was introduced in 1978 [47], practical realisation became feasible with the first fully homomorphic scheme proposed by Gentry in 2009 [48]. Since then, HE schemes have evolved significantly, achieving improved efficiency and broader applicability, particularly in FL scenarios where secure processing of sensitive data is required.

5.1. Types of homomorphic encryption

HE schemes can be classified based on the set of supported operations. Partially homomorphic encryption (PHE) supports a single operation with unlimited frequency [49]. Representative examples include Paillier, which enables homomorphic addition [50], and RSA or ElGamal, which support homomorphic multiplication [51, 52]. In FL, additive PHE is often sufficient, as the server primarily performs gradient aggregation, after which clients decrypt the aggregated result and compute averages according to the FedAvg algorithm [8].

Leveled HE (LHE) support an arbitrary number of additions and a predefined number of multiplications. These schemes require parameter selection that bounds the computational depth of arithmetic circuits. Notable examples include BFV [53] and BGV [54], which are based on the Ring Learning With Errors problem (RLWE) [55] and offer greater computational expressiveness than PHE while remaining more efficient than fully homomorphic schemes.

Fully homomorphic encryption (FHE) supports unrestricted sequences of addition and multiplication on encrypted data, enabling the evaluation of arbitrary functions without decryption. The first FHE construction was introduced by Gentry in 2009 [48]. Modern schemes are optimized for different workloads, such as CKKS for approximate arithmetic commonly used in machine learning [56] and TFHE for efficient Boolean operations [57]. Despite their expressiveness, FHE schemes incur substantial computational overhead, limiting their practicality in large-scale or resource-constrained environments.

5.2. Ciphertext noise and bootstrapping

Modern FHE schemes are based on the RLWE problem [55], where security relies on the introduction of controlled noise. Each ciphertext includes a random noise

component that ensures semantic security, producing different ciphertexts for repeated encryptions of the same message. During homomorphic computation, this noise accumulates—linearly under addition and exponentially under multiplication [48]—thereby imposing a scheme-specific noise threshold that limits computational depth. Exceeding this threshold results in incorrect decryption.

Bootstrapping, introduced by Gentry [48], refreshes ciphertexts by homomorphically evaluating the decryption function, effectively reducing accumulated noise and enabling further computation. However, bootstrapping is computationally expensive and often constitutes the primary performance bottleneck in FHE systems. Consequently, many practical implementations restrict circuit depth and operate in LHE modes to balance correctness and efficiency [43].

5.3. Homomorphic encryption optimization techniques

The primary challenge in applying HE to FL lies in its high computational and communication overhead. To address these limitations, the literature identifies three main optimization directions—algorithmic, hardware, and hybrid, whose adoption is essential for making HE practical in real-world deployments [58].

Algorithmic optimizations focus on reducing computation and communication costs across FL phases. Techniques such as quantization compress high-precision gradients to reduce data transmission [41], while methods like BatchCrypt combine quantization with efficient encoding [59]. Packing and batching techniques, including Chinese Remainder Theorem (CRT) and polynomial packing [60], enable multiple gradients to be processed within a single ciphertext, and SIMD-style optimizations in schemes such as CKKS allow parallel computation.

Hardware optimizations accelerate computationally intensive HE operations, such as modular arithmetic. FPGA- and GPU-based solutions exploit parallelism to improve performance, while CPU-level optimizations using libraries such as Intel HEXL leverage SIMD instructions to enhance efficiency even without specialized hardware [61].

Hybrid approaches combine HE with complementary mechanisms such as DP or secret sharing (SS) to balance security, efficiency, and accuracy. HE-DP schemes reduce cryptographic parameter sizes by leveraging noise for additional protection [62], while HE-SS combinations improve resistance to collusion in scenarios where pure FHE is impractical [63].

5.4. Key management

Cryptographic key management is a critical component in integrating HE into FL, as it directly affects both confidentiality and system interoperability [42]. Since homomorphic operations require ciphertexts to be encrypted under the same key, careful selection of the key management model is essential.

The shared-key approach employs a common public key for encryption and a shared private key for decryption across all clients [64]. This model ensures full ciphertext interoperability and enables straightforward

homomorphic aggregation without additional protocols [18]. However, it introduces a single point of failure: compromise of the shared secret key – through leakage or collusion – results in complete disclosure of all client updates [65]. Moreover, client possession of the private key increases the risk of unauthorized decryption by malicious participants [22], limiting this approach to highly trusted environments.

Threshold cryptography distributes the secret key into multiple shares using schemes such as Shamir Secret Sharing [31, 66], requiring a predefined threshold of participants to cooperatively decrypt the aggregated result [22]. This eliminates single points of compromise and improves resistance to collusion, as neither individual clients nor the server possesses the full key [15, 25]. The main limitation lies in increased computational and communication overhead, since threshold decryption requires multi-party coordination at each iteration, reducing scalability in large or IoT-heavy federated settings [22].

The multi-key (client-specific key) approach assigns each client an independent key pair, thereby avoiding shared secrets and significantly strengthening confidentiality [67]. To enable aggregation, ciphertexts encrypted under different keys are converted into a common key domain via key switching without decryption, using client-generated switching keys [68]. The aggregated ciphertext typically requires coordinated decryption involving multiple private keys, enhancing resistance to collusion [69]. Some implementations further employ proxy re-encryption to facilitate key transformation by a trusted intermediary without exposing plaintexts [70]. While offering strong security, this approach incurs additional cryptographic complexity and communication overhead.

6. Discussion

This narrative survey shows that while FL mitigates risks associated with centralized data collection, it does not inherently guarantee confidentiality, integrity, or availability. Achieving secure federated learning requires additional protection mechanisms that introduce unavoidable trade-offs between privacy strength, computational overhead, and scalability. Differential privacy, secure multi-party computation, and homomorphic encryption each address different aspects of this trade-off, indicating that security in federated learning must be treated as a system-level design challenge rather than a purely cryptographic problem.

Furthermore, the suitability of privacy-preserving mechanisms is strongly influenced by the underlying federated learning architecture. Resource-constrained cross-device environments limit the applicability of computationally intensive schemes, whereas cross-silo and hierarchical settings are better aligned with homomorphic encryption and advanced key management strategies. At the same time, most existing approaches primarily protect confidentiality, while integrity and availability threats remain insufficiently addressed. The diversity of assumptions, architectures, and evaluation criteria across literature complicates direct comparison, motivating the need for a systematic, PRISMA-based

analysis of homomorphic encryption approaches, presented in the second part of this study.

7. Conclusion

This survey examined the security challenges of federated learning from a system and threat-oriented perspective, with emphasis on confidentiality, integrity, and availability. By analyzing federated learning architectures, threat models, and privacy-preserving mechanisms, the study highlighted the limitations of data decentralization alone and the necessity of additional cryptographic protection. A comparative discussion of differential privacy, secure multi-party computation, and homomorphic encryption showed that homomorphic encryption offers a compelling balance between strong confidentiality and preservation of model accuracy, despite its computational cost. These findings establish a conceptual foundation for the systematic review presented in Part II, which applies the PRISMA methodology to comprehensively analyze homomorphic encryption-based federated learning solutions and identify open research challenges.

References

- [1] Dimension Market Research (2025). Homomorphic Encryption Market Size, Forecasts and Trends Report (June 2025), Dimension Market Research, Available at: <https://dimensionmarketresearch.com/report/homomorphic-encryption-market/#overview/> (accessed 05 Dec. 2025).
- [2] Jenkin, D. (2025). How much data is created every day: key insights & trends, DesignRush, Available at: <https://www.designrush.com/agency/big-data-analytics-companies/trends/how-much-data-is-created-every-day>.
- [3] Gheisari, M. et al. (2023). Deep learning: Applications, architectures, models, tools, and frameworks: A comprehensive survey, *CAAI Transactions on Intelligence Technology*, vol. 8, no. 3, pp. 581-606, <https://doi.org/10.1049/cit2.12180>.
- [4] Jabeen, S., Li, X., Amin, M. S., Bourahla, O., Li, S. & Jabbar, A. (2023). A review on methods and applications in multimodal deep learning, *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 19, no. 2s, pp. 1-41, <https://doi.org/10.1145/3545572>.
- [5] Gartner (2022). Gartner identifies top five trends in privacy through 2024, *Gartner*, Available at: <https://www.gartner.com/en/newsroom/press-releases/2022-05-31-gartner-identifies-top-five-trends-in-privacy-through-2024> (accessed 05 Dec. 2025).
- [6] Booth, R. (2025). Cyber-attacks rise by 50% in past year, UK security agency says, *The Guardian*, Available at: <https://www.theguardian.com/technology/2025/oct/14/cyber-attacks-rise-in-past-year-uk-security-agency-says> (accessed 05 Dec. 2025).
- [7] Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T. & Bacon, D. (2016). Federated learning: Strategies for improving communication efficiency, *arXiv preprint*, arXiv:1610.05492, <https://arxiv.org/abs/1610.05492>.
- [8] McMahan, B., Moore, E., Ramage, D., Hampson, S. & Arcas, B. A. Y. (2017). Communication-efficient learning of deep networks from decentralized data, *Artificial Intelligence and Statistics (AISTATS)*, PMLR, pp. 1273-1282.
- [9] Allied Market Research (2024). Federated Learning Solutions Market Size, Share, Competitive Landscape and Trend Analysis Report, by Component (Solution, Services), by Application, by Industry Vertical: Global Opportunity Analysis and Industry Forecast, 2024-2032, *Allied Market Research*,

Available at: <https://www.alliedmarketresearch.com/federated-learning-solutions-market-A14591> (accessed 05 Dec. 2025).

- [10] Zhu, L., Liu, Z. & Han, S. (2019). Deep leakage from gradients, *Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS)*, Curran Associates Inc., pp. 14774-14784.
- [11] Melis, L., Song, C., De Cristofaro, E. & Shmatikov, V. (2019). Exploiting unintended feature leakage in collaborative learning, *2019 IEEE Symposium on Security and Privacy (SP)*, IEEE, pp. 691-706, doi: 10.1109/SP.2019.00029.
- [12] Gao, H., He, N. & Gao, T. (2023). SVeriFL: Successive verifiable federated learning with privacy-preserving, *Information Sciences*, vol. 622, pp. 98-114, <https://doi.org/10.1016/j.ins.2022.11.124>.
- [13] Fontenla-Romero, O., Guijarro-Berdiñas, B., Hernández-Pereira, E. & Pérez-Sánchez, B. (2023). FedHEONN: Federated and homomorphically encrypted learning method for one-layer neural networks, *Future Generation Computer Systems*, vol. 149, pp. 200-211, <https://doi.org/10.1016/j.future.2023.07.018>.
- [14] Chen, X., Tian, Y., Wang, S., Yang, K., Zhao, W. & Xiong, J. (2025). DBFL: Dynamic Byzantine-Robust Privacy Preserving Federated Learning in Heterogeneous Data Scenario, *Information Sciences*, vol. 700, Art. No. 121849, <https://doi.org/10.1016/j.ins.2024.121849>.
- [15] Lai, P., Zhang, M., Tang, Y., Yue, Y. & Di, F. (2025). VPAFL: Verifiable Privacy-Preserving Aggregation for Federated Learning Based on Single Server, *Computers, Materials & Continua*, vol. 84, no. 2, pp. 2935-2957, <https://doi.org/10.32604/cmc.2025.065887>.
- [16] Mohandas, R., Veena, S., Kirubasri, G., Thusnavis Bella Mary, I. & Udayakumar, R. (2024). Federated Learning with Homomorphic Encryption for Ensuring Privacy in Medical Data, *Indian Journal of Information Sources and Services*, vol. 14, no. 2, pp. 17-23, <https://doi.org/10.51983/ijiss-2024.14.2.03>.
- [17] Stripelis, D., Gupta, U., Saleem, H., Dhinagar, N., Ghai, T., Anastasiou, C., Sánchez, R., Ver Steeg, G., Ravi, S., Naveed, M., Thompson, P. M. & Ambite, J. L. (2024). A federated learning architecture for secure and private neuroimaging analysis, *Patterns*, vol. 5, no. 8, Art. No. 101031, <https://doi.org/10.1016/j.patter.2024.101031>.
- [18] Xu, R., Gao, S., Li, C., Joshi, J. & Li, J. (2024). Dual defense: Enhancing privacy and mitigating poisoning attacks in federated learning, *Advances in Neural Information Processing Systems*, vol. 37, pp. 70476-70498, doi: 10.52202/079017-2253.
- [19] Robbins, H. & Monro, S. (1951). A Stochastic Approximation Method, *The Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400-407, <https://doi.org/10.1214/aoms/1177729586>.
- [20] Li, L., Sun, X., Shi, N., Ci, X. & Liang, C. (2024). A Stealthy Communication Model for Protecting Aggregated Results Integrity in Federated Learning, *Electronics*, vol. 13, no. 19, p. 3870, <https://doi.org/10.3390/electronics13193870>.
- [21] Asiri, M., Khemakhem, M. A., Alhebshi, R. M., Alsulami, B. S. & Eassa, F. E. (2025). RPFL: A Reliable and Privacy-Preserving Framework for Federated Learning-Based IoT Malware Detection, *Electronics*, vol. 14, no. 6, p. 1089, <https://doi.org/10.3390/electronics14061089>.
- [22] Park, J. & Lim, H. (2022). Privacy-Preserving Federated Learning Using Homomorphic Encryption, *Applied Sciences*, vol. 12, no. 2, p. 734, <https://doi.org/10.3390/app12020734>.
- [23] He, Y., Wei, L., Chen, F., Zhang, H., Yu, J. & Wang, H. (2025). FedAI: Federated recommendation system with anonymized interactions, *Expert Systems with Applications*, vol. 271, Art. No. 126564, <https://doi.org/10.1016/j.eswa.2025.126564>.
- [24] Zhang, C., Zhang, X., Yang, X., Liu, B., Zhang, Y. & Zhou, R. (2025). Poisoning attacks resilient privacy-preserving federated learning scheme based on lightweight homomorphic encryption, *Information Fusion*, vol. 121, Art. No. 103131, <https://doi.org/10.1016/j.inffus.2025.103131>.
- [25] Bouamama, J., Benkaouz, Y. & Ouzzif, M. (2025). V-FLEX: Verifiable cross-silo federated learning using trusted execution environment, *Progress in Artificial Intelligence*, <https://doi.org/10.1007/s13748-025-00386-9>.
- [26] Rieyan, S. A., Alam, M. M., Alzain, M. A. & Rahman, M. M. (2024). An advanced data fabric architecture leveraging homomorphic encryption and federated learning, *Information Fusion*, vol. 102, Art. No. 102004, <https://doi.org/10.1016/j.inffus.2024.102004>.
- [27] Angulo, E., Márquez, J. & Villanueva-Polanco, R. (2023). Training of Classification Models via Federated Learning and Homomorphic Encryption, *Sensors*, vol. 23, no. 4, p. 1966, <https://doi.org/10.3390/s23041966>.
- [28] Gu, C. & Li, B. C. (2024). Hierarchical federated learning architectures for the metaverse, *ZTE Communications*, vol. 22, no. 2, pp. 39-48, <https://doi.org/10.12142/ZTECOM.202402006>.
- [29] Alzahrani, B. & Yang, D. (2025). PrivHFL: A privacy-preserving scheme for hierarchical federated learning, *Computer Networks*, vol. 273, Art. No. 111764, <https://doi.org/10.1016/j.comnet.2025.111764>.
- [30] Wang, H., Feng, J., Bu, M., Ding, Y., Tang, S. & Yang, C. (2025). MVFL: verifiable privacy-preserving federated learning using multi-key homomorphic encryption, *The Journal of Supercomputing*, vol. 81, no. 8, p. 915, <https://doi.org/10.1007/s11227-025-07419-z>.
- [31] Cao, S., Liu, S., Yang, Y., Du, W., Zhan, Z., Wang, D. & Zhang, W. (2025). A hybrid and efficient Federated Learning for privacy preservation in IoT devices, *Ad Hoc Networks*, vol. 170, Art. No. 103761, <https://doi.org/10.1016/j.adhoc.2025.103761>.
- [32] Wang, B. & Yang, F. (2024). Lightweight and privacy-preserving hierarchical federated learning mechanism for artificial intelligence-generated image content, *Journal of Real-Time Image Processing*, vol. 21, no. 4, Art. No. 149, <https://doi.org/10.1007/s11554-024-01524-7>.
- [33] Shokri, R., Stronati, M., Song, C. & Shmatikov, V. (2017). Membership inference attacks against machine learning models, *2017 IEEE Symposium on Security and Privacy (SP)*, IEEE, pp. 3-18, <https://doi.org/10.1109/SP.2017.41>.
- [34] Nasr, M., Shokri, R. & Houmansadr, A. (2019). Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning, *2019 IEEE Symposium on Security and Privacy (SP)*, IEEE, pp. 739-753, <https://doi.org/10.1109/SP.2019.00065>.
- [35] Hitaj, B., Ateniese, G. & Perez-Cruz, F. (2017). Deep models under the GAN: Information leakage from collaborative deep learning, *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, ACM, pp. 603-618, <https://doi.org/10.1145/3133956.3134012>.
- [36] Wang, Z., Song, M., Zhang, Z., Song, Y., Wang, Q. & Qi, H. (2019). Beyond inferring class representatives: User-Level Privacy Leakage From Federated Learning, *2019 IEEE Conference on Computer Communications (INFOCOM)*, IEEE, pp. 2512-2520, <https://doi.org/10.1109/INFOCOM.2019.8737416>.
- [37] Fredrikson, M., Jha, S. & Ristenpart, T. (2015). Model inversion attacks that exploit confidence information and basic countermeasures, *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS)*, ACM, pp. 1322-1333, <https://doi.org/10.1145/2810103.2813677>.
- [38] Zhou, A., Jiang, N. & Tang, T. (2024). Asynchronous Robust Aggregation Method with Privacy Protection for IoV Federated Learning, *World Electric Vehicle Journal*, vol. 15, no. 1, p. 18, <https://doi.org/10.3390/wevj15010018>.

- [39] Deng, Y., Guo, B. & Chen, S. (2025). Privacy-Preserving Approach to Edge Federated Learning Based on Blockchain and Fully Homomorphic Encryption, *Electronics*, vol. 14, no. 2, p. 361, <https://doi.org/10.3390/electronics14020361>.
- [40] Manzoor, H. U., Shabbir, A., Chen, A., Flynn, D. & Zoha, A. (2024). A survey of security strategies in federated learning: Defending models, data, and privacy, *Future Internet*, vol. 16, no. 10, p. 374, <https://doi.org/10.3390/fi16100374>.
- [41] Aziz, R., Banerjee, S., Bouzeffrane, S. & Le Vinh, T. (2023). Exploring homomorphic encryption and differential privacy techniques towards secure federated learning paradigm, *Future Internet*, vol. 15, no. 9, p. 310, <https://doi.org/10.3390/fi15090310>.
- [42] Xin, W., Jiaqian, L., Xueshuang, D., Haoji, Z. & Lianshan, S. (2025). A survey of differential privacy techniques for federated learning, *IEEE Access*, vol. 13, pp. 6539-6555, <https://doi.org/10.1109/ACCESS.2024.3523909>.
- [43] Feretakis, G., Papaspyridis, K., Gkoulalas-Divanis, A. & Verykios, V. S. (2024). Privacy-Preserving Techniques in Generative AI and Large Language Models: A Narrative Review, *Information*, vol. 15, no. 11, p. 697, <https://doi.org/10.3390/info15110697>.
- [44] Stan, O., Thouvenot, V., Boudguiga, A., Kapusta, K., Zuber, M. & Sirdey, R. (2022). A Secure Federated Learning: Analysis of Different Cryptographic Tools, *Proceedings of the 19th International Conference on Security and Cryptography (SECRYPT)*, SciTePress, pp. 669-674, <https://doi.org/10.5220/0011322700003283>.
- [45] Liu, F., Zheng, Z., Shi, Y., Tong, Y. & Zhang, Y. (2024). A survey on federated learning: a perspective from multi-party computation, *Frontiers of Computer Science*, vol. 18, Art. No. 181336, <https://doi.org/10.1007/s11704-023-3282-7>.
- [46] Shalabi, E., Khedr, W., Rushdy, E. & Salah, A. (2025). A comparative study of privacy-preserving techniques in federated learning: A performance and security analysis, *Information*, vol. 16, no. 3, p. 244, <https://doi.org/10.3390/info16030244>.
- [47] Rivest, R. L., Adleman, L. & Dertouzos, M. L. (1978). On Data Banks and Privacy Homomorphisms, in R. A. DeMillo, D. P. Dobkin, A. K. Jones & R. J. Lipton (Eds.), *Foundations of Secure Computation*, vol. 4, Academic Press, pp. 169-180.
- [48] Gentry, C. (2009). Fully homomorphic encryption using ideal lattices, *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC)*, ACM, pp. 169-178, <https://doi.org/10.1145/1536414.1536440>.
- [49] Munjal, K. & Bhatia, R. (2022). Analysing RSA and PAILLIER homomorphic Property for security in Cloud, *Procedia Computer Science*, vol. 215, pp. 240-246, <https://doi.org/10.1016/j.procs.2022.12.027>.
- [50] Paillier, P. (1999). Public-Key Cryptosystems Based on Composite Degree Residuosity Classes, *Advances in Cryptology — EUROCRYPT '99*, Lecture Notes in Computer Science, vol. 1592, Springer, pp. 223-238, https://doi.org/10.1007/3-540-48910-X_16.
- [51] Rivest, R. L., Shamir, A. & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, vol. 21, no. 2, pp. 120-126, <https://doi.org/10.1145/359340.359342>.
- [52] ElGamal, T. (1985). A public-key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469-472, <https://doi.org/10.1109/TIT.1985.1057074>.
- [53] Fan, J. & Vercauteren, F. (2012). Somewhat practical fully homomorphic encryption, *IACR Cryptology ePrint Archive*, Paper 144, Available at: <https://eprint.iacr.org/2012/144>.
- [54] Brakerski, Z., Gentry, C. & Vaikuntanathan, V. (2012). (Leveled) fully homomorphic encryption without bootstrapping, *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS)*, ACM, pp. 309-325, <https://doi.org/10.1145/2090236.2090262>.
- [55] Regev, O. (2005). On lattices, learning with errors, random linear codes, and cryptography, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC)*, ACM, pp. 84-93, <https://doi.org/10.1145/1060590.1060603>.
- [56] Cheon, J. H., Kim, A., Kim, M. & Song, Y. (2017). Homomorphic encryption for arithmetic of approximate numbers, *Advances in Cryptology — ASIACRYPT 2017*, Lecture Notes in Computer Science, vol. 10624, Springer, pp. 409-437, https://doi.org/10.1007/978-3-319-70694-8_15.
- [57] Chillotti, I., Gama, N., Georgieva, M. & Izabachène, M. (2020). TFHE: Fast fully homomorphic encryption over the torus, *Journal of Cryptology*, vol. 33, no. 1, pp. 34-91, <https://doi.org/10.1007/s00145-019-09319-x>.
- [58] Xie, Q., Jiang, S., Jiang, L., Huang, Y., Zhao, Z., Khan, S., Dai, W., Liu, Z. & Wu, K. (2024). Efficiency optimization techniques in privacy-preserving federated learning with homomorphic encryption: A brief survey, *IEEE Internet of Things Journal*, vol. 11, no. 14, pp. 24569-24580, <https://doi.org/10.1109/JIOT.2024.3382875>.
- [59] Zhang, C., Li, S., Xia, J., Wang, W., Yan, F. & Liu, Y. (2020). BatchCrypt: Efficient homomorphic encryption for cross-silo federated learning, *Proceedings of the 2020 USENIX Annual Technical Conference (USENIX ATC)*, USENIX Association, pp. 493-506.
- [60] Childs, L. N. (1995). The Chinese Remainder Theorem, *A Concrete Introduction to Higher Algebra*, Undergraduate Texts in Mathematics, Springer, pp. 194-207, https://doi.org/10.1007/978-1-4419-8702-0_12.
- [61] Intel Labs (2025). HEXL: High-Performance Homomorphic Encryption Library, *Intel Labs*, Available at: <https://github.com/IntelLabs/hexl> (accessed 14 Nov. 2025).
- [62] Zhao, J., Huang, C., Wang, W., Xie, R., Dong, R. & Matwin, S. (2023). Local differentially private federated learning with homomorphic encryption, *The Journal of Supercomputing*, vol. 79, pp. 19365-19395, <https://doi.org/10.1007/s11227-023-05378-x>.
- [63] Ming, Y., Wang, S., Wang, C., Liu, H., Deng, Y., Zhao, Y. & Feng, J. (2024). VCSA: Verifiable and collusion-resistant secure aggregation for federated learning using symmetric homomorphic encryption, *Journal of Systems Architecture*, vol. 156, Art. No. 103279, <https://doi.org/10.1016/j.sysarc.2024.103279>.
- [64] Liang, X., Xu, Y., Lin, Y., Zhang, C. (2025). Federated split learning via dynamic aggregation and homomorphic encryption on non-IID data, *The Journal of Supercomputing*, vol. 81, Art. No. 63, <https://doi.org/10.1007/s11227-024-06612-w>.
- [65] Bai, F., Zhao, Y., Shen, T., Zeng, K., Zhang, X. & Zhang, C. (2025). FedOPCS: An optimized poisoning countermeasure for non-IID federated learning with privacy-preserving stability, *Symmetry*, vol. 17, no. 5, p. 782, <https://doi.org/10.3390/sym17050782>.
- [66] Shamir, A. (1979). How to share a secret, *Communications of the ACM*, vol. 22, no. 11, pp. 612-613, <https://doi.org/10.1145/359168.359176>.
- [67] Liu, Q.-X., Zhang, J.-G. & Tan, B. (2024). Financial fraud recognition model based on privacy preserving and federated learning, *Journal of Network Intelligence*, vol. 9, no. 4, pp. 2360-2374.
- [68] Liu, Z., Chen, S., Ye, J., Fan, J., Li, H. & Li, X. (2022). DHSA: Efficient doubly homomorphic secure aggregation for cross-silo federated learning, *The Journal of Supercomputing*, vol. 79, pp. 2819-2849, <https://doi.org/10.1007/s11227-022-04745-4>.
- [69] Ma, J., Naas, S.-A., Sigg, S. & Lyu, X. (2022). Privacy-preserving federated learning based on multi-key homomorphic

encryption, *International Journal of Intelligent Systems*, vol. 37, no. 9, pp. 5880-5901, <https://doi.org/10.1002/int.22818>.
[70] Wei, Z., Wang, J., Zhao, Z. & Zhao, Z. (2025). Byzantine-robust federated learning considering distributed

Non-IID data in privacy-sensitive scenarios, *Information Fusion*, vol. 124, Art. No. 103345, <https://doi.org/10.1016/j.inffus.2025.103345>.