



User-Centric Encrypted Data Sharing and Machine Learning System with Secure Access Control

Putheti Sudhakar^{ORCID}, Orchu Aruna^{ORCID}*, Ravindranath Tagore Mamilla^{ORCID}

Computer Science and Engineering, VVIT University, Nambur 522508, India

Corresponding Author Email: arunasri52@gmail.com

Copyright: ©2026 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/ijssse.160809>

Received: 8 May 2026

Revised: 29 June 2026

Accepted: 21 July 2026

Available online: 31 August 2026

Keywords:

homomorphic encryption, machine learning, secure data sharing, federated learning, data security

ABSTRACT

Protecting sensitive data during storage, sharing, and machine-learning processing remains a major challenge in cloud and distributed environments. Conventional encryption can protect data at rest and in transit, but the data usually has to be decrypted before computation, which can create a risk of exposure. To address this issue, this study proposes a user-centric privacy-preserving framework that combines hybrid homomorphic encryption (HHE), federated learning (FL), secure access control, and adversarial training (AT). AES is used to protect sensitive data, while RSA is employed to securely exchange the symmetric key. HHE allows selected computations to be performed directly on encrypted inputs without revealing the underlying plaintext. FL enables multiple clients to collaboratively train a shared model while keeping their raw datasets locally, and AT improves the model's ability to withstand malicious input perturbations. A logistic-regression-based learning process is used to evaluate the proposed framework under distributed conditions. The experimental results show that the global model accuracy increases from 61.2% to 88.3%, with the average client accuracy reaching 88.2%. Under adversarial conditions, the defended model achieves an accuracy of 83.4%, compared with 51.2% for the undefended model. The measured encryption, decryption, aggregation, and overall training times are 12.5 ms, 8.3 ms, 45 ms, and 2.3 s, respectively, indicating that the proposed security mechanisms introduce manageable computational overhead. Overall, the results demonstrate the potential of the integrated framework for privacy-sensitive applications in cloud computing, healthcare, finance, and IoT environments.

1. INTRODUCTION

The digital data is growing rapidly, which is leading organizations to use cloud infrastructures to store, compute, and perform machine-learning applications [1, 2]. However, this is dangerous because it may lead to unauthorized access, data breach and inference attacks [3, 4]. The proposed system will address this data analysis and processing privacy gap by enabling the machine learning algorithms to be executed on the encrypted data themselves. This system maintains the encryption of the data during any exposure to plaintext during the lifecycle with the help of hybrid homomorphic encryption (HHE) and federated learning (FL).

While data at rest and data in transit is encrypted using conventional techniques like AES and RSA, there is no way to directly compute on encrypted data [5, 6]. This means that sensitive information could still be shared in the training, inference or collaborative processing in cloud-based machine learning environments. This leaves a large gap in privacy in applications that require the secure analysis of confidential data. The presented framework is designed to overcome the aforementioned issue by combining HHE and FL.

Current privacy-preserving systems tend to have hard-coded security policies and application-specific protection

layers, restricting their ability to deal with dynamic threats and distributed cooperation. These approaches ensure secure storage/ communication, but not a single approach offers a secure mechanism for computation, access control and model training in the same workflow. Several approaches also rely on trusted hardware or central coordination, which may limit flexibility and add to vulnerability in real-world deployments. All of these restrictions spur the need for an integrated comprehensive end-to-end privacy-preserving learning system.

1.1 Limitations of existing system

The current systems rely on fixed security policies and threat models [1]. This causes them to be unprepared to deal with emerging attack patterns and zero-day vulnerabilities [2]. Their inability to work effectively in dynamic cloud and enterprise settings due to the absence of context-aware and self-learning security mechanisms [3]. Also, a lot of the existing architectures lack a seamless support of cross-domain collaboration of data as well as making the privacy end-to-end [4].

Consequently, there is still a challenge in providing data with great security between healthcare, industrial, and

government systems without jeopardizing the disclosure of susceptible metadata or compliance violations [5]. Even in case homomorphic encryption (HE) merges with trusted execution environments (TEE) [6].

A leakage in information may be incomplete outputs or critical components of the model, which are the things that are supposed to remain locked up [7]. When machine learning is fueled by healthcare, encryption that is embedded into training binds the system, limiting mobility [5, 8].

The majority of the tools are unable to follow through when data is transferred through cloud configurations, particularly when ownership of individuals changes [3]. Handover access becomes a maze, which causes poor following in collaborative spaces. A part of the issue is reduced to trust – teams do not work in a similar fashion; thus, it is hard to share trained models [2, 4]. Their individual policies put loopholes.

The difference between the encryption makes sections remain discrete and they are not able to communicate effectively [9]. Key, identity and live tracking tools are scattered without any apparent strategy – protective measures are becoming weak [10]. Pieces open holes quickly when they are working in isolation. Machines at varying speeds will be disastrous to group studying; underperformers will pull others back [2]. Add in covert layers of blending and jumbled protection, then fast fasteners appear even when the disguise is down to bareness. Nevertheless, most of these secret algorithms provide virtually no indication that they indeed inhibit leakage when training is complete or when making predictions. Cunning techniques are also able to reveal hidden information or rebuild raw data despite protection measures [7].

Encryption tools and flexible defenses are a sound combination that sounds like intelligent thinking, but real-life testing is hardly ever done, especially in terms of the preservation of the data truthfulness in changing circumstances [11]. The trick in this is a big-time performance drops in the presence of bizarre signals being detected in encrypted data and it requires greater processing capacity, and more storage-space – as well as the extra performance capability is yet again more stranded when it is needed to respond quickly over wide networks [12]. Many existing systems lack interpretable components and do not adequately support learning from closed data in active cloud defense scenarios.

1.2 Research contributions

The main contributions of this work are summarized as follows:

- Suggest an end-to-end privacy-preserving learning framework for secure data processing in cloud environments.
- Combine HHE and FL to enables collaborative model training over encrypted data.
- Employ secure access control, so that users remain in control and have ownership over their data when it is shared and processed.
- Add adversarial training (AT) to boost robustness to malicious perturbations and inference attacks.
- Judge the framework on the basis of its privacy protection, robustness, and computational efficiency.

In this study, we design a secure machine learning framework to integrate encryption, FL, access control and adversarial robustness. The framework aims to maintain confidentiality during the entire lifecycle of the data from

storage, transmission, sharing to training. It also enables multiple clients to collaborate without sharing raw data with the central server, making it suitable for security-sensitive applications such as healthcare, finance, and cloud-based distributed systems.

2. LITERATURE SURVEY

Recent studies have investigated several approaches to privacy-preserving machine learning, including FL, HE, TEE, and secure multi-party computation. These methods help reduce the exposure of sensitive information during collaborative model development. However, much of the existing work focuses on a single security mechanism or addresses only a specific stage of the learning process [1, 2].

Many existing frameworks are developed for specific threat models or controlled environments, which can make them difficult to apply to dynamic cloud environments involving multiple domains and potential adversarial attacks. In addition, the computational overhead of advanced security mechanisms and the reliance on trusted hardware may affect scalability and practical deployment [3, 4].

Policy-based cryptographic mechanisms and automated trust verification can be used to support the secure distribution of trained models. Such approaches demonstrate the importance of coordinated security policies when data and models are exchanged across different organizational boundaries [5, 6].

FL offers an effective approach for collaborative model training because the raw datasets remain with the participating clients rather than being transferred to a central server. Nevertheless, model updates and the aggregation process can still expose sensitive information and therefore require additional security measures, especially when participating environments have different capabilities and security requirements [7].

Cryptographic techniques based on AES and elliptic-curve cryptography have also been explored in machine-learning applications. These methods have been applied to areas such as data integrity monitoring and the identification of suspicious data behavior [8].

Privacy-aware approaches to cyber defense further demonstrate the need to protect operational and security-related data without significantly affecting the performance of machine-learning models. Such approaches aim to maintain a balance between privacy protection, learning effectiveness, and the requirements of ongoing security operations [9, 10].

HE and multi-key encryption schemes offer stronger protection by allowing computations to be performed on encrypted data. However, their relatively high computational requirements remain a major challenge for practical and large-scale deployment [11, 12].

Recent studies have also examined these techniques in more practical federated-learning environments. FedNIC, for example, explores homomorphic-encryption offloading for FL and highlights the potential of hardware-assisted encrypted computation [13]. Similarly, secure multi-authority access-control mechanisms have been proposed for fog-enabled IoT cloud storage, enabling fine-grained authorization across multiple independent authorities [14].

Recent surveys suggest that protecting privacy in FL requires a combination of complementary mechanisms, including cryptographic protection, secure aggregation, differential privacy, and system-level security controls, rather

than relying on a single technique [15, 16]. Homomorphic adversarial learning has also been explored to combine privacy protection with greater resilience against malicious manipulation [17]. In addition, blockchain-based federated-learning architectures have been investigated for privacy-sensitive applications such as telemedicine, while machine-learning-based access-control methods have been developed

to provide fine-grained protection for sensitive visual data [18, 19]. More recent encrypted federated-learning schemes, including FedSHE, continue to address the challenge of balancing the security provided by HE with model accuracy, communication overhead, and computational efficiency [20]. Table 1 summarizes and compares these existing approaches with the proposed framework.

Table 1. Comparative analysis of existing approaches

Refs	Method / Framework	Encryption Strategy	Learning Architecture	Attack Resistance	Computational Cost	Deployment Scenario	Remaining Limitations
[2]	Privacy-preserving federated learning (FL)	Homomorphic encryption (HE) + differential privacy	FL	Moderate	Moderate to high	Distributed learning	Communication overhead and robustness limitations
[3]	Encrypted computation with federated strategies	HE + secure computation	Hybrid distributed learning	Moderate	High	Cloud/distributed systems	Performance-cost trade-off remains
[8]	Secure machine learning with cryptography	AES + ECC	Machine learning with encrypted inputs	Moderate	Moderate	Data protection systems	Does not fully support end-to-end encrypted training
[10]	Privacy-aware cyber defense system	Privacy-preserving protection layer	Intelligent security system	Moderate	Moderate	Cyber defense / cloud systems	Limited model transparency and verification
[5]	Adversarially robust secure learning	adversarial training (AT) + security mechanisms	Secure ML	High	Moderate to high	Distributed security-sensitive environments	Still needs stronger integration with encryption layers
Proposed work	Secure privacy-preserving learning framework	HHE + AES + RSA	FL + AT	High	Optimized	Cloud-based collaborative learning	Addresses secure computation, sharing, and robustness in one system

3. PROPOSED METHODOLOGY

The proposed framework operates on a modular privacy-preserving workflow in which user identity is first verified, sensitive data are protected with AES, and the AES key is secured using RSA. Homomorphic computation and FL then support model processing without exposing raw information to the server. AT strengthens resistance to malicious perturbations, while secure sharing and monitored storage maintain confidentiality throughout the data lifecycle. The complete workflow is illustrated in Figure 1.

3.1 System architecture

The architecture is organized into coordinated security layers, as illustrated in Figure 1. The authentication and access-control layer manages user registration and authorization; the encryption layer protects stored and transmitted information; the learning layer performs privacy-preserving computation; the secure-sharing layer controls data exchange; and the monitoring layer maintains security logs and retention information.

The proposed pipeline starts with efficient symmetric data protection by encrypting using AES, and then goes on to use public-key cryptography for protection. AES key is then encrypted with the receiver's public key to assure secure key exchange. If a numerical computation is needed, the encrypted data is computed using homomorphic operations, and the server is able to do the computations necessary to learn without ever needing to access the plaintext data. The final output is only decryptable by authorized users.

3.2 Operational framework

The operational framework, illustrated in Figure 2, integrates AES encryption, RSA key protection, and homomorphic computation to support secure end-to-end processing.

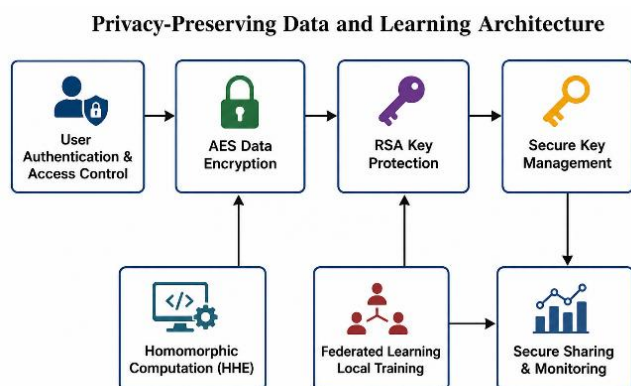


Figure 1. Privacy-preserving data protection and federated learning (FL) architecture

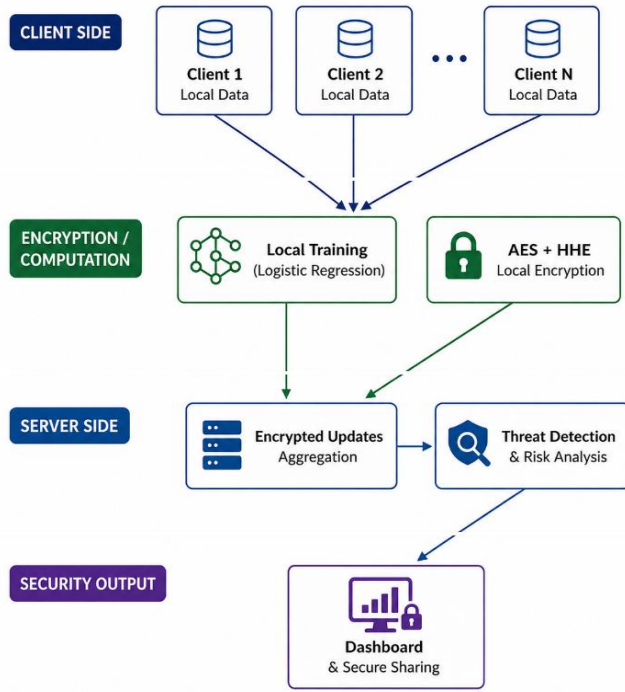


Figure 2. Proposed methodology framework

3.2.1 AES encryption

It encrypts all sensitive user data (like files, user logs, and ML input data) before sending it to storage or the machine learning module. Acts as the first security layer before RSA key exchange and HE.

The file is split into fixed-size blocks

Input plaintext $\rightarrow$ divided into blocks substitution, permutation, and mixing operations

$$C = EK(P) \quad (1)$$

where, C is Ciphertext, P is Plaintext and K is Secret key.

3.2.2 RSA (asymmetric encryption)

An Asymmetric Encryption algorithm used to securely encrypt and decrypt the AES secret key between parties. This ensures the private key is never shared directly.

$$C = P^e \text{ mod } m \quad (2)$$

The sender encrypts the AES key (P) with the receiver's public key. The receiver uses their private key to decrypt it.

3.2.3 Homomorphic encryption

Server has encrypted inputs and plaintext model weights w_i . Using HE properties:

- Homomorphic addition: $(a) + E(b) = E(a + b)$
- Homomorphic multiplication: $(a) \times w = E(a \times w)$

Server computes:

$$(z) = w_1 \times E(x_1) + w_2 \times E(x_2) + w_3 \times E(x_3) + E(b) \quad (3)$$

This entire expression remains encrypted. The result (z) is the encrypted z value.

It is used to support computation on encrypted inputs without revealing the underlying plaintext values that are encrypted, via HE. Here, the encrypted features are fused with

model parameters, and the server is able to derive intermediate results without decrypting them. The output is then encrypted, and can be decrypted by only an authorized user or client. This ensures that when computing, the data remains confidential and the likelihood of data leakage is minimized.

3.2.4 Logistic regression

Logistic regression is a supervised machine learning algorithm used for binary or multiclass classification. Unlike linear regression (which predicts continuous values), logistic regression predicts probabilities of class membership. Uses the sigmoid (logistic) function to map predicted values to the range $[0,1]$.

$$(z) = \frac{1}{1 + e^z} \quad (4)$$

where,

$$z = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n$$

(z) gives the probability of belonging to class 1.

3.2.5 Federated learning

Train a global model collaboratively across multiple clients without sharing raw data.

Initialization: Server initializes global model weights w_0 .

Local Training: Each client k updates model weights on its local data.

$$w_t + 1k = w_{tk} - \eta \nabla L_k(w_{tk}) \quad (5)$$

where, L_k is the local loss, η is learning rate.

Aggregation: Server aggregates local weights to update global model:

$$w_{t+1} = \sum_{k=1}^K \frac{n_k}{n} w_{t+1}^k \quad (6)$$

where, n_k = number of samples at client k , $n = \sum n_k$.

Iteration: Repeat local training and aggregation until convergence.

3.2.6 Adversarial training

Enhance the model robustness through adversarial perturbed examples training with Fast Gradient Sign Method (FGSM).

Compute gradient of loss with respect to input:

$$\nabla_x J(\theta, x, y) \quad (7)$$

Generate adversarial example:

$$x' = x + \epsilon \cdot \text{sign}(\nabla_x J(\theta, x, y)) \quad (8)$$

where, ϵ controls the perturbation magnitude.

Train the model using both original and adversarial examples:

$$\min_{\theta} = \frac{1}{m} \sum_{i=1}^m [L(\theta, x_i, y_i) + L(\theta, x'_i, y_i)] \quad (9)$$

AT is added to make the learning model more robust with respect to malicious and perturbed inputs. Adversarial

examples are created by applying FGSM to the input data, a small perturbation. This model is then trained with both unaltered and altered samples, to make it more robust against input manipulation and more secure in security critical applications.

4. RESULTS AND DISCUSSION

The proposed framework was evaluated for confidentiality, robustness, collaborative learning, and computational feasibility against conventional and traditional federated-learning settings. Figure 3 summarizes the optimization-related performance measures used in the evaluation.

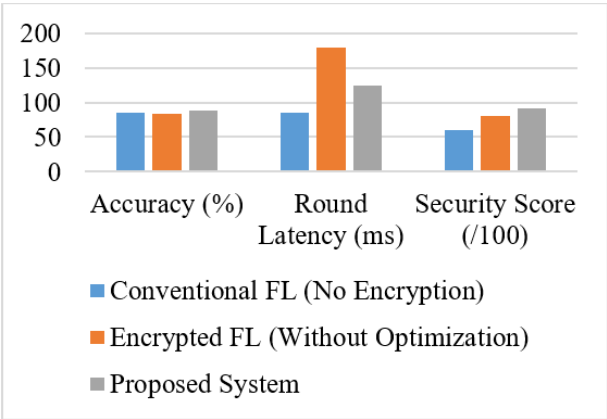


Figure 3. Optimization efficiency in secure federated learning (FL)

Three metrics—convergence speed, accuracy stability, and local-global alignment—were used to examine the learning behavior of the proposed framework in the training process. The results show that the proposed method is capable of obtaining smoother optimization results than the baseline system. The reported accuracy improvement is quite large, but the primary merit of this finding is that the global model is able to converge more reliably in the presence of distributed conditions and has less oscillatory behavior when aggregated.

4.1 Federated learning training progress

The FL paradigm has superior convergence qualities and improved performance stability compared to traditional FL paradigms. The empirical findings indicate that the global model accuracy improves by 27.1 percentage points, rising from 61.2% to 88.3%. The latter accuracy is higher than the typical reported convergence range of 82–85% seen in typical distributed learning setups working in heterogeneous data settings.

The global accuracy of traditional implementations generally increases gradually with sometimes ad hoc oscillations following the drift of clients and aggregation bias. The behavior constrains the convergence ceiling that can be attained and it might need more communication cycles to stabilize. Conversely, the suggested framework has a better terminal accuracy and smoother growth trajectory, which suggests better optimization dynamics and aggregation robustness.

4.1.1 Behavior analysis convergence canonical

In FL with heterogeneous or non-IID data, the problem of

client drift and unstable gradient updates frequently occurs. The proposed framework exhibits the convergence curve with higher improvement in the initial rounds, less fluctuation from one round to another and a higher final accuracy. This means that the aggregation process can help minimise update divergence and provide more stable distributed optimisation. The improved convergence behavior should not be taken as a guarantee of higher terminal accuracy, but as an indication of improved training stability as well.

4.1.2 Local-global model alignment

The high overlap between average client accuracy and global model accuracy suggests good consistency between local and global learning. Figure 4 presents the client-level training trajectories used to examine this alignment.

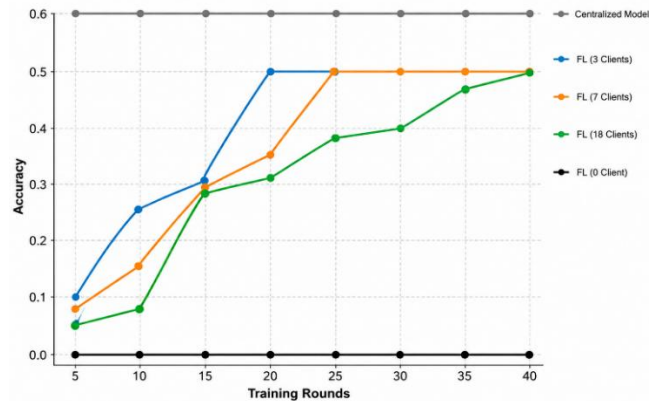


Figure 4. Accuracy vs. training rounds by client scale

The small variance confirms that there is good synchronization of the distributed parameter updates and the gradient variance is minimal amongst clients. Such alignment denotes the enhanced generalization ability and implies that the aggregation strategy could be used, as a means of offsetting local data heterogeneity.

4.2 Client participation and performance analysis

Equitable client participation is a key characteristic of secure FL as it can cause the bias towards global optimization if the contributions from the clients are imbalanced. In the proposed framework, the samples from each client are of the same size, aiding the uniformity of the samples update quality among the clients. The low inter-client variance in the accuracy range is evidence of low variance in the accuracy values across the clients and therefore strongly suggests that the aggregation model is fair to learning contribution. This is particularly applicable for decentralized deployment in which participation cannot be relied upon to be consistent.

4.2.1 Precision distribution among clients

The last local accuracies are also seen in a close range of performance of between 85.8 and 90.1 percent. The mean accuracy of clients is 88.2 which is close to the model accuracy of 88.3 worldwide. This difference on the margin proves the successful synchronization of local and aggregated parameters.

The top performing client scores 90.1, whereas the lowest performing scores 85.8. The narrow range on the values of accuracy is an indication of low inter-client divergence and equal learning behavior.

4.2.2 Comparative interpretation

The performance of clients at the client level in a normal federation system is also likely to have a larger variation because of the distribution asymmetry and the inconsistency of aggregation. Conversely, the suggested framework illustrates:

- Decreased dispersion of performance.
- Equalized weights of contribution.
- Good local–global model correspondence.
- Better distributed generalization capacity.

The fact that client accuracies and global performance are very close confirms the strength of the aggregation mechanism and helps in scalability during decentralized deployments.

The analysis of client participation and distribution of accuracy shows that the contribution of distributed nodes is even and they have very little inter-client variability. The fact that the average client accuracy (88.2) and global accuracy (88.3) are very close indicates that there was good aggregation and low model drift. The standard deviation is also small (1.48 percentage points), which is another confirmation that distributed optimization is stable in heterogeneous data scenarios.

4.3 Fast Gradient Sign Method adversarial attack defense evaluation

The AT results indicate that the defended model retains substantially higher accuracy under attack than the undefended model. Figure 5 provides a comparative view of the defended and normal model performance across the evaluated conditions.

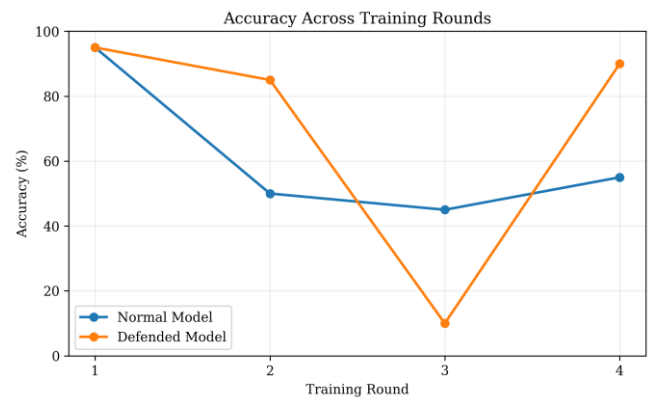


Figure 5. Scalability analysis: federated learning (FL) model accuracy vs. centralized baseline

The normal model has a classification accuracy of 95.6 under clean input conditions whereas defended model has a classification accuracy of 94.2. The relatively small decrease in clean accuracy (1.4) in the defended model is an indication that the adversarial method has been implemented, with very minimal overhead incurred in normal prediction performance.

But in adversarial attack situations, there is a significant drop in the performance of the normal model as it goes to 51.2% instead of 95.6%, and this is a 44.4% drop in accuracy. This high drop indicates the inability of the standard models to withstand gradient-based perturbations.

Conversely, the adversarial version of the same model has an accuracy of 83.4 per cent regardless of the conditions, with the accuracy error having dropped as low as 10.8. This is a 32.2 percent enhancement in the resilience of attacks than the

normal model. The score of robustness is enhanced (53.6% with the normal model and 88.5% with the defended model) establishing effectiveness of the integrated AT mechanism.

These findings show that the given defense approach significantly improves the model robustness and does not deteriorate the predictive performance under clean input. The negligible trade-off in clean accuracy and high adversarial resistance justifies the relevance of the proposed architecture to secure distributed learning setup.

4.4 Security and encryption effectiveness

Results of the encryption process show that sensitive information is secure during storage, transmission, and computation. The proposed framework is designed to perform computation on encrypted data whereas traditional systems need to decrypt it before computing. By adding user controlled key management, the sensitive keys are not being controlled from a central point, further enhancing the confidentiality. These results are in line with the framework's assertion of offering end-to-end privacy instead of privacy at specific points in the process.

4.5 System performance metrics evaluation

The results of the performance metrics show that the added security layers do not introduce prohibitive overhead. Figure 6 presents the comparative system-level performance across accuracy, robustness, security, latency, and efficiency.

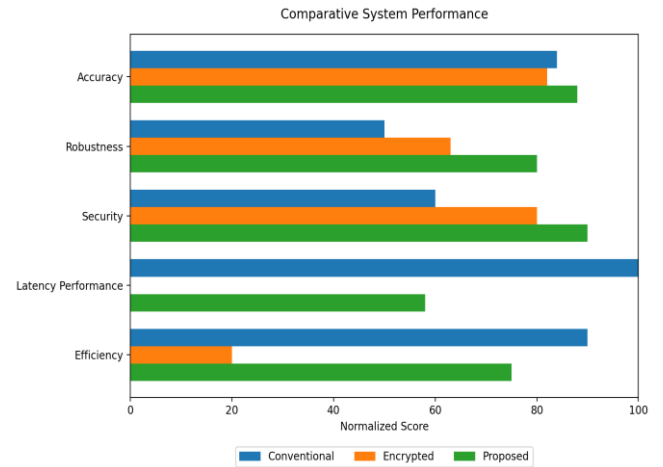


Figure 6. Comparative system performance analysis

Encryption time is 12.5 ms with decryption time at 8.3 ms, both are optimum. The values represent that cryptographic operations do not have too much overhead in comparison to the entire training cycle. The comparatively reduced decryption time also supports the fact that there is efficient management of keys and light cryptographic implementation.

Aggregation and the synchronization of model parameters among distributed clients takes 45 ms to reflect fast aggregation. This aggregation latency is within reasonable working ranges of the real-time distributed learning which is operational.

The round-trip time of communication is 125 ms, and it is a low latency time. This proves that there is no substantial delay in the exchange of parameters between clients and the central coordinator, because of the secure communication protocol (HTTPS/TLS with authentication mechanisms).

The overall training period takes a total of 2.3 seconds and this shows that the system is highly efficient. The brief overall time spent in training justifies the idea that secure integration is not likely to reduce scalability or computational feasibility.

The proposed architecture maintains competitive performance in calculations when compared to traditional federated configurations that lack integrated security layers, and has a significant improvement in privacy and robustness.

The efficient time measures mean that it can be used in real-time or near real-time distributed applications, such as internet of things-style systems and privacy-sensitive systems, such as healthcare and finance.

5. CONCLUSION

This study presented a privacy-preserving framework for secure data sharing and machine learning by integrating HHE, FL, secure access control, and AT. The framework is designed to protect data during storage, transmission, computation, and sharing while retaining user control through authorization and key-management mechanisms.

The experimental results indicate that the framework supports collaborative learning without exposing raw user data and improves model resilience to adversarial perturbations. The combined use of cryptographic protection, decentralized training, and adversarial defense provides a practical balance among privacy, learning performance, and computational cost for the evaluated configuration.

Overall, the results show that secure distributed learning can be achieved without a substantial reduction in model accuracy or operational efficiency in the evaluated setting. Future work will extend the framework to deep-learning models, reduce computational overhead, and incorporate differential privacy to strengthen formal privacy guarantees.

REFERENCES

- [1] Wang, Q., Zhou, L., Bai, J., Koh, Y.S., Cui, S., Russello, G. (2023). HT2ML: An efficient hybrid framework for privacy-preserving machine learning using HE and TEE. *Computers & Security*, 135: 103509. <https://doi.org/10.1016/j.cose.2023.103509>
- [2] Sundar, S., Priyadharshini, A. (2024). Advanced security framework for patient monitoring systems: Integrating machine learning and encryption for enhanced data protection. In 2024 International Conference on Emerging Research in Computational Science (ICERCS), Coimbatore, India, pp. 1-8. <https://doi.org/10.1109/ICERCS63125.2024.10895327>
- [3] Tamizshelvan, C., Vijayalakshmi, V. (2024). Cloud data access governance and data security using distributed infrastructure with hybrid machine learning architectures. *Wireless Networks*, 30(4): 2099-2114. <https://doi.org/10.1007/s11276-024-03658-9>
- [4] Ranathunga, T., McGibney, A., Bharti, S. (2024). Enabling secure and self-sovereign machine learning model exchange in manufacturing data spaces. *Journal of Industrial Information Integration*, 42: 100733. <https://doi.org/10.1016/j.jii.2024.100733>
- [5] Ramos-Cruz, B., Andreu-Perez, J., Martínez, L. (2024). The cybersecurity mesh: A comprehensive survey of involved artificial intelligence methods, cryptographic protocols and challenges for future research. *Neurocomputing*, 581: 127427. <https://doi.org/10.1016/j.neucom.2024.127427>
- [6] Kairouz, P., McMahan, H.B. (2021). Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1-2): 1-210. <https://doi.org/10.1561/22000000083>
- [7] El Mestari, S.Z., Lenzini, G., Demirci, H. (2024). Preserving data privacy in machine learning systems. *Computers & Security*, 137: 103605. <https://doi.org/10.1016/j.cose.2023.103605>
- [8] Jana, A.K., Saha, S. (2023). Integrating machine learning with cryptography to ensure dynamic data security and integrity. *International Journal for Research in Applied Science and Engineering Technology*, 11(10): 208-213. <https://doi.org/10.22214/ijraset.2023.55967>
- [9] Hangan, A., Lazea, D., Cioara, T. (2024). Privacy preserving anomaly detection on homomorphic encrypted data from IoT sensors. *arXiv preprint arXiv:2403.09322*. <https://doi.org/10.48550/arXiv.2403.09322>
- [10] Feng, C., Wu, S., Liu, N. (2017). A user-centric machine learning framework for cyber security operations center. In 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), Beijing, China, pp. 173-175. <https://doi.org/10.1109/ISI.2017.8004902>
- [11] 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*, 37(9): 5880-5901. <https://doi.org/10.1002/int.22818>
- [12] Hegde, M.G., Ruthvika, B., Jain, R.B., Shenoy, P.D., Venugopal, K.R., Canchi, A. (2025). A privacy-preserving federated learning method with homomorphic encryption for chronic kidney disease stage prediction. *Engineering, Technology & Applied Science Research*, 15(4): 26019-26026. <https://doi.org/10.48084/etasr.11928>
- [13] Choi, S., Patel, D., Zad Tootaghaj, D., Cao, L., Ahmed, F., Sharma, P. (2024). FedNIC: Enhancing privacy-preserving federated learning via homomorphic encryption offload on SmartNIC. *Frontiers in Computer Science*, 6: 1465352. <https://doi.org/10.3389/fcomp.2024.1465352>
- [14] Ma, R., Zhang, L. (2023). SPMAC: Secure and privacy-preserving multi-authority access control for fog-enabled IoT cloud storage. *Journal of Systems Architecture*, 142: 102951. <https://doi.org/10.1016/j.sysarc.2023.102951>
- [15] Saha, S., Hota, A., Chattopadhyay, A.K., Nag, A., Nandi, S. (2024). A multifaceted survey on privacy preservation of federated learning: Progress, challenges, and opportunities. *Artificial Intelligence Review*, 57: 184. <https://doi.org/10.1007/s10462-024-10766-7>
- [16] Wang, X., Li, J., Ding, X., Zhang, H., Sun, L. (2025). A survey of differential privacy techniques for federated learning. *IEEE Access*, 13: 6539-6555. <https://doi.org/10.1109/ACCESS.2024.3523909>
- [17] Dong, W., Lin, C., He, X., Xu, S., Huang, X. (2025). Privacy-preserving federated learning via homomorphic adversarial networks. In International Conference on Knowledge Science, Engineering and Management, pp. 393-404. https://doi.org/10.1007/978-981-95-3052-6_30
- [18] Hiwale, M., Walambe, R., Potdar, V., Kotecha, K. (2023). A systematic review of privacy-preserving

- methods deployed with blockchain and federated learning for the telemedicine. *Healthcare Analytics*, 3: 100192. <https://doi.org/10.1016/j.health.2023.100192>
- [19] Akcay, M.H., Atli, B.G., Rao, S.P., Bakas, A. (2025). From see to shield: ML-assisted fine-grained access control for visual data. *arXiv preprint arXiv:2510.19418*. <https://doi.org/10.48550/arXiv.2510.19418>
- [20] Pan, Y., Chao, Z., He, W., Jing, Y., Li, H., Wang, L. (2024). FedSHE: Privacy preserving and efficient federated learning with adaptive segmented CKKS homomorphic encryption. *Cybersecurity*, 7: 40. <https://doi.org/10.1186/s42400-024-00232-w>