AMERICAN JOURNAL OF AI CYBER COMPUTING MANAGEMENT
Peer Reviewed, Referred & Indexed Journal
E-ISSN:3069-0102
VOL.6, NO. 3(2026)

BLOCKCHAIN-BASED PRIVACY-PRESERVING AND SECURE FEDERATED LEARNING FRAMEWORK

Umema Samreen[1], Dr. I. Samuel Peter James[2]
[1]PG Scholar, Department of AI&DS, Shadan Women's College of Engineering and Technology, Hyderabad, umemasamreen5111@gmail.com
[2]Assoc Professor, Department of AI&DS, Shadan Women's College of Engineering and Technology drisamuelpeterjames@gmail.com

ABSTRACT
Federated Learning (FL) allows clients to safely communicate gradients calculated on their local data with the server, eliminating the need for them to reveal their private local datasets. In classical FL, the server may utilize its dominating position to deduce sensitive information from the clients' shared gradients during the model aggregation phase. Malicious clients may also send malicious and counterfeit gradients during model training. Such conduct reduces the utility and dependability of trained models in addition to jeopardizing the integrity of the global model. This paper suggests a Block chain-based Privacy-preserving and Secure Federated Learning (BPS-FL) system that uses threshold homomorphic encryption to safeguard the local gradients of clients in order to successfully solve such privacy and security assault challenges. To prevent malicious gradient attacks, we create a Byzantine-robust aggregation protocol for BPS-FL to enable cipher-text level safe model aggregation. Additionally, our usage of a block chain as the basic distributed architecture for documenting all learning processes ensures the data's immutability and traceability. Our thorough security study and numerical assessment show that BPS-FL can successfully fight off poisoning assaults and meets privacy criteria.
KEYWORDS: Federated Learning (FL); block chain; privacy-preserving; model poisoning attack; Byzantine robustness

1.INTRODUCTION

Artificial intelligence (AI) has advanced so quickly that it has produced amazing results in a number of domains, such as intrusion detection, natural language processing, and medical diagnosis. The availability of the large-scale, high-quality datasets needed to train efficient machine learning models is the main factor driving these developments. However, centralized data collecting has become more challenging due to stringent legal requirements, data isolation across organizations, and growing awareness of data privacy. Federated Learning (FL), which allows several clients to cooperatively train a shared global model without directly disclosing their private local datasets, has emerged as a possible solution to these problems. A central server receives model updates from each client in a typical FL workflow, which computes local gradients on its private data and combines them to improve the global model. Despite its benefits, FL has serious security and privacy issues. Adversaries, such as rogue clients and hijacked servers, might use gradient updates to infer private information even while raw data is not disclosed. Furthermore, model poisoning attacks—in which adversary clients purposefully alter gradients or corrupt labels to impair the global model's performance—are made possible by the distributed and opaque nature of FL. Current privacy-preserving methods, such Differential Privacy (DP), Homomorphic Encryption (HE), and Secure Multi-party Computation (SMC), have drawbacks in terms of resilience against adversarial behaviours, communication cost, and accuracy erosion. A single central server is used by several FL systems to aggregate model changes, which creates vulnerabilities including single-point failures and vulnerability to malevolent manipulation. Although a number of protection techniques, including Krum, Trimmed-mean, Median, FL Trust, and later improvements, have been put forth to lessen poisoning threats, the most need unencrypted access to clients' gradients, which raises the possibility of information leaking. Building safe and reliable FL systems is severely hampered by this intrinsic contradiction between privacy protection and poisoning attack detection. We suggest a Block chain-based Privacy-preserving and Secure Federated Learning (BPS-FL) paradigm that concurrently guarantees privacy, robustness, and system integrity in order to solve these issues. BPS-FL incorporates threshold Paillier homomorphic encryption to provide secure ciphertext-level aggregation while safeguarding clients' gradient information. We create a Byzantine-robust aggregation protocol that evaluates clients' encrypted gradients without sacrificing privacy in order to identify and reduce fraudulent contributions. This protocol is backed by a decentralized federated learning committee. Additionally, BPS-FL removes the need for a single server and offers tamper-proof tracking of model change histories by utilizing block chain's immutability, transparency, and decentralized consensus processes, improving traceability and accountability. Numerical evaluations and extensive security research show that BPS-FL retains high model accuracy in adversarial environments, successfully identifies poisoning behaviors, and achieves strong privacy guarantees. The suggested plan provides a workable and safe framework for implementing federated learning in adversarial, real-world, and privacy-sensitive settings. Inconsistent client involvement and varied data distributions provide practical difficulties for FL deployments as well. Divergence and instability during model training may result from several clients producing data with non-

identical distributions (non-IID) in real-world situations. Furthermore, unanticipated client failures might result from communication limits, sporadic connection, and processing constraints, making reliable aggregation challenging. Current privacy-preserving FL frameworks frequently overlook such dynamic behaviors and assume idealized participation patterns, which leads to poor performance or insufficient model updates. These difficulties underscore the necessity of a strong, decentralized system that guarantees trustworthy participation tracking, safe communication, and verifiable aggregation—even in the face of erratic or malevolent client behavior. Integrating homomorphic encryption into an adversarial FL environment creates additional challenges, even if it provides robust privacy guarantees by allowing computation directly on ciphertext. Existing Byzantine-resilient techniques usually require plaintext gradients to identify anomalies, and traditional HE-based FL frameworks sometimes lack tools to prevent model poisoning or Sybil assaults at the ciphertext level. By distributing key shares, guaranteeing contribution auditability, and enforcing protocol compliance without depending on a single authority, blockchain's decentralized trust architecture offers a chance to get beyond these restrictions.

2.STIMULATING

The design and development of a comprehensive Blockchain-based Privacy-preserving and Secure Federated Learning (BPS-FL) framework that tackles the crucial issues of data privacy, model security, and system reliability in distributed machine learning environments is included in the project's scope. The project's major goal is to allow clients to work together to train a global model while guaranteeing that their sensitive local data and gradient information are completely safeguarded using threshold Paillier homomorphic encryption. This entails setting up ciphertext-level aggregation procedures, encrypted gradient computing, and secure key generation to stop any participating entity from deducing private information. In order to maintain the integrity and reliability of the global model, the scope also includes developing a Byzantine-robust aggregation protocol that can detect, mitigate, and neutralize poisoning assaults even when gradients are still encrypted. The project incorporates a blockchain-based decentralized architecture for storing model changes, verifying client contributions, and preserving unchangeable training records in order to improve system resilience, remove single-point failures, and offer tamper-proof auditability. Throughout the learning process, the blockchain layer guarantees visible participation, safe consensus, and verifiable responsibility. The scope also includes an assessment of the suggested system's performance in terms of model correctness, privacy guarantees, computing overhead, communication efficiency, scalability, and robustness in different adversarial scenarios. The project's overall goal is to provide a safe, dependable, and privacy-preserving federated learning framework that may be used in practical applications where trust and privacy are crucial. Lastly, a thorough experimental assessment utilizing adversarial situations and real-world datasets is part of the project's scope. This entails evaluating computing efficiency, resilience against different poisoning assaults, communication cost, privacy preservation, and model correctness. The evaluation's goal is to show that the suggested BPS-FL architecture is feasible, scalable, and secure in real-world distributed learning settings.

3.OBJECTIVE

This project's main goal is to provide a reliable, safe, and privacy-preserving federated learning system that allows decentralized model training without disclosing sensitive client information. The project's specific goal is to use threshold Paillier homomorphic encryption to safeguard local gradients, enabling safe ciphertext-level aggregation while avoiding information leakage. Designing a Byzantine-robust aggregation technique that can identify and mitigate malicious or poisoned gradients given by adversary clients is another important goal. In order to prevent single-point failures, provide transparent model update recording, and give unchangeable, tamper-proof records of all training interactions, the project also aims to incorporate a blockchain-based decentralized architecture. The system also seeks to achieve realistic communication and processing efficiency appropriate for real-world deployments, retain high model accuracy, and guarantee scalability across huge client populations.

3.1 WHAT THE PROJECT IS ABOUT

Federated Learning allows several clients to work together to train a global model without exchanging their raw data, however there are still significant privacy, security, and reliability issues with existing FL systems. Recent research indicates that an adversarial server or external attacker may reconstruct sensitive client information from shared gradients, resulting in serious privacy leakage, even if gradients rather than datasets are communicated. However, because FL is decentralized and only partially trusted, it is susceptible to a range of poisoning attacks, in which malicious clients deliberately send gradients that have been manipulated or fabricated in order to contaminate the global model. While homomorphic encryption-based FL schemes have trouble detecting malicious updates because all gradients remain encrypted, existing privacy-preserving techniques like differential privacy and secure multi-party computation either reduce model accuracy or add significant communication and computation overhead. Additionally, a single point of failure that may be used for manipulation or system disruption is created by the fact that many FL systems rely on a centralized server

for model aggregation. Accountability in distributed contexts is made more difficult by the absence of a safe, transparent, and impenetrable method for monitoring model modifications. A federated learning framework that simultaneously protects client privacy, fends off Byzantine and poisoning attacks, guarantees reliable aggregation of encrypted gradients, and removes centralized points of vulnerability through decentralized system design is therefore desperately needed.

4. EXISTING SYSTEM:

- There are a number of privacy, security, and robustness issues with the current Federated Learning (FL) systems. Conventional FL systems reveal model changes to the server, which may allow hackers to deduce private information. Though each has its drawbacks, privacy-preserving methods such as Homomorphic Encryption (HE), Differential Privacy (DP), and Secure Multi-party Computation (SMC) have been developed. To put it simply, homomorphic encryption is a cryptographic technique that allows data to be processed while it is still encrypted. Homomorphic encryption allows calculations on encrypted statistics to be done simultaneously, in contrast to standard encryption, which requires statistics to be decrypted for every important operation. When decrypted, the outcome of such calculations matches the outcome of the same operations carried out at the plaintext records. Touchy statistics may be examined, altered, and dealt with in this way while still being encrypted, protecting both security and privacy.

4.1 Existing System Disadvantages:

- High communication expenses and continuous customer availability are necessary for MC.DP adds noise, which might reduce the accuracy of the model.
- While protecting anonymity, HE makes it more difficult to identify fraudulent gradients.
- Boost the effectiveness of searches.

4.2 THE PROPOSED SYSTEM

- We present a novel Privacy-preserving and Secure Federated Learning (BPS-FL) approach based on blockchain technology. While maintaining client privacy, this plan may successfully identify and prevent poisoning behaviors during collaborative training. In order to securely aggregate and analyze local gradients without compromising client privacy, we employ threshold Paillier homomorphic encryption technique in BPS-FL, which permits computations on encrypted data while the outputs remain encrypted. We have created a federated learning committee with many members who are in charge of communicating with the server and carrying out procedures to collect weighted scores of each client's local gradients without jeopardizing their privacy in order to better thwart model poisoning assaults. Privacy-preserving and Secure Federated Learning (BPS-FL) approach based on blockchain technology. We present the BPS-FL system in this work. In order to prevent harmful behaviors, BPS-FL uses blockchain technology as the distributed architecture for FL. This allows all learning processes to be recorded in an immutable ledger.FL is a cutting-edge distributed machine learning framework that makes it possible to train algorithms on several dispersed servers or edge devices. Because local data samples are not shared in FL, privacy is protected and transmission overhead is decreased. Clients no longer need to transmit sensitive data to a central repository thanks to this approach.

4.2.1PROPOSED SYSTEM ADVANTAGE

- The privacy needs and the ability to successfully ward against poisoning assaults
- The outcomes demonstrate that our approach effectively achieves the objectives of high fidelity, security, and dependability.
- Increased search effectiveness.

5. METHODOLOGIES

1. Design of User Interfaces

We create the project's windows in this module. All users can securely log in using these windows. The user must enter their username and password in order to connect to the server. The user can log in to the server immediately if they have previously left; otherwise, they must register their information, including their username, password, and email address. In order to maintain upload and download rates, the server will establish accounts for every user. The user ID will be used as the name. To access a particular website, you often need to log in.

2. Committee

Committee members can register, log in, and safely upload files using this module. The KGC public key is used to encrypt uploaded data before they are submitted for blockchain processing and key creation. Committees also accept or reject requests for client keys.

3. Client

Clients register and log in to search encrypted files and request public/private keys. They may safely download original material, validate public keys, and decrypt files using authorized private keys. Malicious login attempts are automatically blocked by the system.

4. Cloud Server

The cloud module manages client and committee approvals. It monitors blocked users and allows unblocking after review. This acts as the administrative access control layer

5. Key Generation Center

Cryptographic key generation and distribution fall under the purview of KGC. It guarantees that keys are provided only after authorization and grants requests for public and private keys.

6. Smart Contract

By dividing encrypted data into blocks, this module manages blockchain processes. To guarantee integrity and immutability, every block is securely chained and hashed.

6.TECHNIQUE ALGORITHM USED

6.1 PROPOSED ALGORITHM

- **Block chain Algorithm**

The foundation of the healthcare system is the block chain algorithm, which guarantees decentralization, transparency, and tamper resistance. Every transaction in this project's distributed ledger, including doctor registration, patient file uploads, and laboratory report production, is recorded in a safe and verifiable chain of blocks. Digital signatures, hash values, timestamps, and encrypted file contents are all included in each block and are connected via cryptographic connections based on past and present hashes. The block chain method guarantees that all records are verified by a consensus process prior to being added to the chain, hence eliminating the single point of failure seen in conventional centralized healthcare systems. This ensures data validity, traceability, and resilience in the project, making it difficult for hackers to change or falsify medical information covertly. In order to ensure data integrity and avoid unwanted alterations, network members use a consensus process to validate transactions when new data is introduced. Blockchain is frequently employed in cryptocurrency, supply chain management, healthcare, cloud storage, and edge computing contexts due to its transparency, immutability, security, and traceability. Blockchain may be used in systems like EdgeHydra to keep a safe and unchangeable record of data storage, access, and recovery activities, guaranteeing integrity, trust, and accountability across dispersed edge nodes.

- **SHA-256 Algorithm**

The project uses the SHA-256 algorithm, a cryptographic hash function, to ensure data authenticity and integrity during all transactions. A distinct 256-bit hash value that serves as a digital fingerprint is formed each time a doctor, patient, or file is created. It is simple to identify any manipulation since even a slight alteration to the file or identification data entirely changes the hash. SHA-256 is used in this project to provide hash values for patient and physician IDs as well as for each block of an encrypted patient file kept in the block chain. This guarantees medical data's traceability and immutability, shielding it against alteration or fabrication. By connecting the hashes of one block to the next, the algorithm also facilitates the decentralized nature of the system by creating a safe, verifiable chain of medical transactions.SHA-256, which was created by the National Security Agency and standardized by the National Institute of Standards and Technology, transforms data into a distinct digital fingerprint that is impossible to undo. It is very useful for guaranteeing data integrity and identifying illegal changes since even a slight alteration in the input data results in an entirely new hash value. SHA-256 is widely used in blockchain systems, digital signatures, secure communications, file integrity checks, and password verification.

6.2 EXISTING ALGORITM

- **Homomorphic Encryption (HE)**

To put it simply, homomorphic encryption is a cryptographic technique that allows data to be processed while it is still encrypted. Homomorphic encryption allows calculations on encrypted statistics to be done simultaneously, in contrast to standard encryption, which requires statistics to be decrypted for every important operation. When decrypted, the outcome of such calculations matches the outcome of the same operations carried out at the plaintext records. Touchy statistics may be examined, altered, and dealt with in this way while still being encrypted, protecting both security and privacy. This guarantees data privacy and security even when calculations are performed by untrusted servers or cloud providers, protecting sensitive information throughout processing. A homomorphic encryption technique is used by the data owner to encrypt the original data. A server receives and processes theciphertext, or encrypted data. Without knowing the real data, the server manipulates the ciphertext mathematically. The data owner receives the encrypted output that is produced. The outcome of decryption is the same as what would have happened if the operations had been carried out on the original plaintext data. It is especially helpful in cloud and edge computing situations where sensitive data must be safely stored and processed since it maintains data secrecy while processing. Systems like EdgeHydra may improve privacy, handle data securely, and prevent unwanted access by incorporating HE

7.RESULT

FIG 7.1:Home Page

FIG 7.2: Client Registration

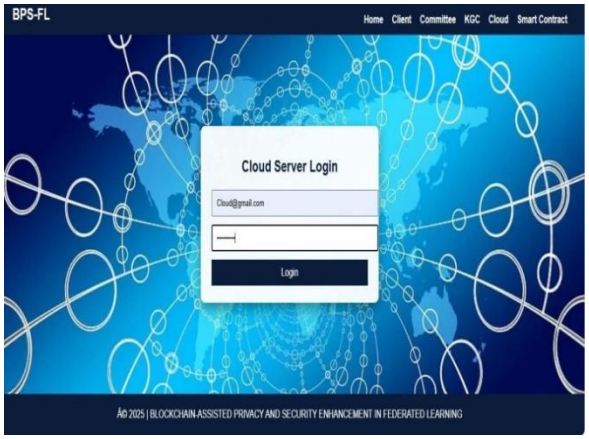

FIG7.3: Cloud Server Login

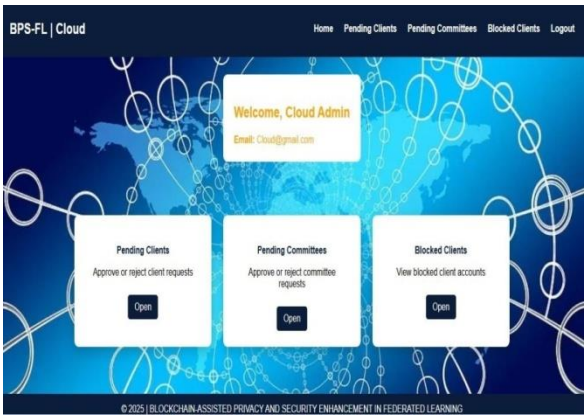

FIG7.4: Cloud Admin

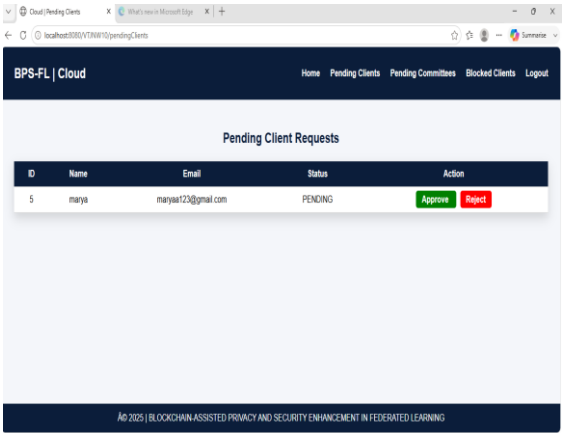

FIG7.5: Pending Client Request

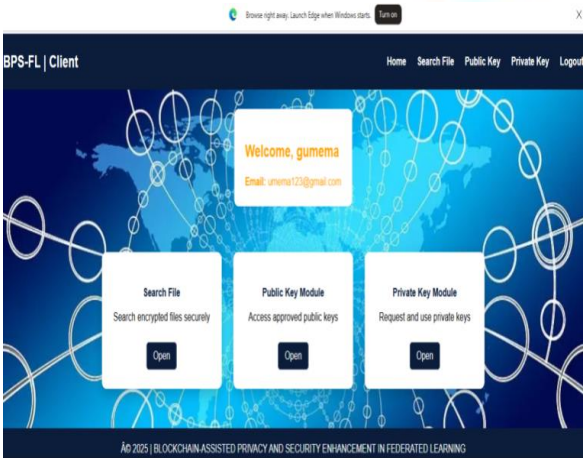

FIG 7.6: Search File

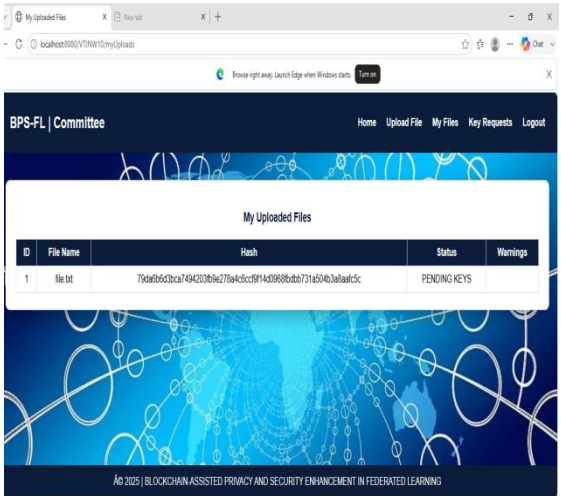

FIG 7.7: My Uploaded file

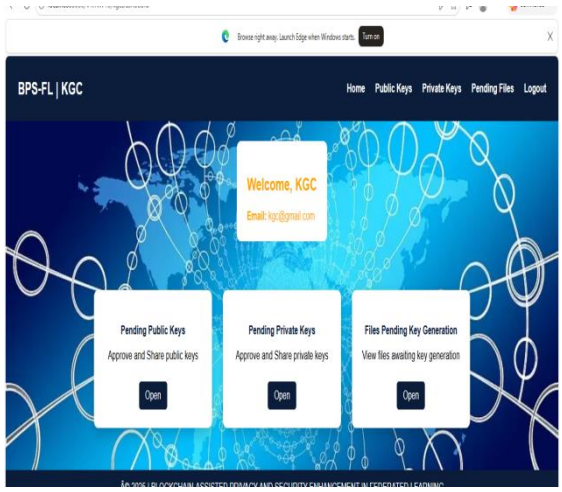

FIG 7.8: Key Generation Center

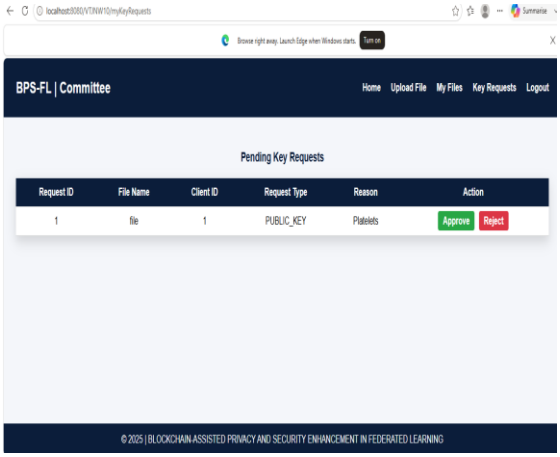

FIG 7.9:Pending Key Requests

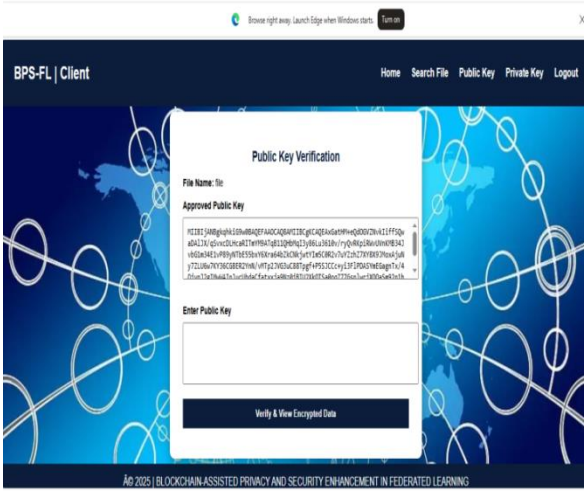

FIG 7.10: Public key verification

8.SYSTEM ARCHITECTURE

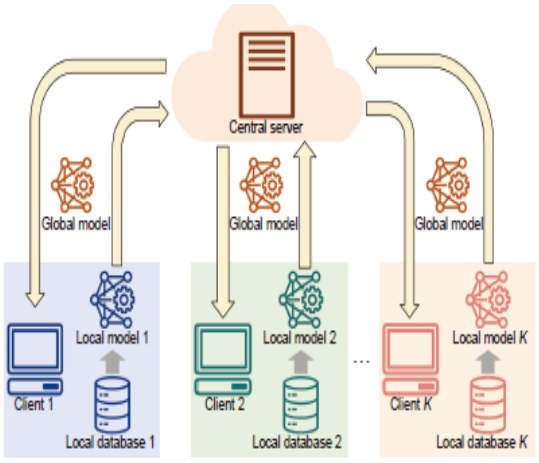

FIG 8:System Architecture

Several clients both trustworthy and perhaps malevolent locally train models on their own data under this architecture, producing gradient vectors that are never shared in plaintext. To ensure secrecy, these gradients are encrypted using threshold Paillier homomorphic encryption before to transmission. In order to provide safe encryption and controlled decryption without a single point of trust, the Key Generation Center (KGC) is in charge of creating and distributing public and private key shares. Patients provide encrypted gradients to a blockchain-deployed Smart Contract that functions as a decentralized coordinator instead of a conventional central server.Without having access to plaintext gradients, the committee assists in identifying and mitigating malicious or poisoned updates. Encrypted global models and encrypted intermediate results are stored by the Cloud Server, which helps with availability and computation without discovering any private information. Every training cycle, update submission, approval, and aggregation are all recorded by the smart contract, which ensures traceability and auditability by creating an immutable ledger. An encrypted global model is created by homomorphic aggregation at the ciphertext level. To avoid unilateral abuse, only approved reconstruction using threshold decryption is permitted. Repeated authentication failures or malicious clients trying improper access are identified and prevented.

9.CONCLUSION

To solve important privacy and security issues in federated learning settings, we suggested a Blockchain-based Privacy-preserving and Secure Federated Learning (BPS-FL) system. By encrypting local gradients using threshold homomorphic encryption, the suggested approach successfully safeguards client data privacy while allowing secure aggregation without disclosing critical gradient information. We created a Byzantine-robust aggregation technique that functions at the ciphertext level to prevent model poisoning attacks, reducing the impact of malicious client updates while guaranteeing precise global model convergence.BPS-FL eliminates single-point failures and ensures immutability, traceability, and transparency throughout the learning process by utilizing blockchain technology as the underlying distributed architecture. According to experimental data, the overhead introduced by BPS-FL stays within acceptable bounds and has no discernible impact on overall system performance, even if the computational cost rises with the number of active clients. Furthermore, even with a significant percentage of malevolent players, the technique retains good categorization accuracy. These findings demonstrate that BPS-FL offers a safe, dependable, and useful way to implement federated learning in hostile and privacy-sensitive settings.

9.2 FUTURE ENHANCEMENT

Although the proposed BPS-FL framework shows good privacy protection, resilience to poisoning attempts, and manageable computing cost; a few improvements might increase its efficiency and scalability. First, as the number of participating clients increases, the computing cost of the existing SecDist-

based distance computation increases. In order to lower latency in large-scale deployments, future research might concentrate on improving secure distance computation utilizing approximate secure aggregation approaches or lightweight cryptographic primitives. Second, to further strengthen defense against insider assaults and lower committee overhead, adaptive committee selection processes based on reputation or trust scores can be incorporated.Third, by lowering the quantity of encrypted updates handled at the blockchain level, edge-level aggregation or hierarchical federated learning can greatly increase scalability. Furthermore, combining homomorphic encryption with differential privacy may offer an additional line of defense against inference attacks. Adopting more effective blockchain platforms or Layer-2 solutions can also optimize smart contract execution costs. Lastly, BPS-FL's practical usefulness would be expanded by adding support for cross-domain federated learning and real-time applications like IoT systems and healthcare monitoring.

REFERENCES

[1] L. Deng and D. Yu, Deep learning: Methods and applications, Found. Trends® Signal Process., vol. 7, nos. 3&4, pp. 197–387, 2014.
[2] L. Peng, N. Wang, N. Dvornek, X. Zhu, and X. Li, FedNI: Federated graph learning with network inpainting for population-based disease prediction, IEEE Trans. Med. Imag., vol. 42, no. 7, pp. 2032–2043, 2023.
[3] Z. Li, X. Wang, W. Yang, J. Wu, Z. Zhang, Z. Liu, M. Sun, H. Zhang, and S. Liu, A unified understanding of deep NLP models for text classification, IEEE Trans. Vis. Comput. Graph., vol. 28, no. 12, pp. 4980–4994, 2022.
[4] R. Zhao, Y. Wang, Z. Xue, T. Ohtsuki, B. Adebisi, and G. Gui, Semisupervised federated-learning-based intrusion detection method for Internet of Things, IEEE Internet Things J., vol. 10, no. 10, pp. 8645–8657, 2023.
[5] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in Proc. of 20th Int. Conf. on Artificial Intelligence and Statistics, Ft. Lauderdale, FL, USA, 2017, pp. 1273–1282.
[6] X. Guo, Z. Liu, J. Li, J. Gao, B. Hou, C. Dong, and T. Baker, VeriFL: communication-efficient and fast verifiable aggregation for federated learning, IEEE Trans. Inf. Forensics Secur., vol. 16, pp. 1736–1751, 2021.
[7] H. Zhou, G. Yang, H. Dai, and G. Liu, PFLF: Privacy preserving federated learning framework for edge computing, IEEE Trans. Inf. Forensics Secur., vol. 17, pp. 1905–1918, 2022.
[8] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, Membership inference attacks against machine learning models, in Proc. IEEE Symp. on Security and Privacy, San Jose, CA, USA, 2017, pp. 3–18.
[9] M. Al-Rubaie and J. M. Chang, Privacy-preserving machine learning: Threats and solutions, IEEE Secur. Priv., vol. 17, no. 2, pp. 49–58, 2019.
[10] X. Cao, J. Jia, and N. Z. Gong, Provably secure federated learning against malicious clients, Proc. AAAI Conf. Artif. Intell., vol. 35, no. 8, pp. 6885–6893, 2021.
[11] Y. Li, H. Li, G. Xu, T. Xiang, X. Huang, and R. Lu, Toward secure and privacy-preserving distributed deep learning in fog-cloud computing, IEEE Internet Things J., vol. 7, no. 12, pp. 11460–11472, 2020.
[12] J. H. Bell, K. A. Bonawitz, A. Gascón, T. Lepoint, and M. Raykova, Secure single-server aggregation with (poly) logarithmic overhead, in Proc. 2020 ACM SIGSAC Conf. Computer and Communications Security, Virtual Event, 2020, pp.1253–1269.
[13] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, Privacy-preserving deep learning via additively homomorphic encryption, IEEE Trans. Inf. Forensics Secur., vol. 13, no. 5, pp. 1333–1345, 2018.
[14] L. Yu, L. Liu, C. Pu, M. E. Gursoy, and S. Truex, Differentially private model publishing for deep learning, in Proc. IEEE Symp. on Security and Privacy, San Francisco, CA, USA, 2019, pp. 332–349. [15] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. S. Quek, and H. Vincent Poor, Federated learning with differential privacy: Algorithms and performance analysis, IEEE Trans. Inf. Forensics Secur., vol. 15, pp. 3454–3469, 2020.
[16] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, Machine learning with adversaries: Byzantine tolerant gradient descent, in Proc. 31st Int. Conf. Neural Information Processing Systems, Long Beach, CA, USA, 2017, pp. 118 – 128.
[17] S. Rajput, H. Wang, Z. Charles, and D. Papailiopoulos, DETOX: A redundancy-based framework for faster and more robust gradient aggregation, arXiv preprint arXiv: 1907.12205, 2019.
[18] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, Byzantine robust distributed learning: Towards optimal statistical rates, in Proc. 35th Int. Conf. on Machine Learning, Stockholm, Sweden, 2018, pp. 5650– 5659.
[19] M. Fang, X. Cao, J. Jia, and N. Z. Gong, Local model poisoning attacks to Byzantine-robust federated learning, arXiv preprint arXiv: 1911.11815, 2019.
[20] X. Cao, M. Fang, J. Liu, and N. Z. Gong, FLTrust: Byzantine-robust federated learning via trust bootstrapping, arXiv preprint arXiv: 2012.13995, 2020.
[21] Y. Dong, X. Chen, K. Li, D. Wang, and S. Zeng, FLOD: Oblivious defender for private Byzantine-robust federated learning with dishonest-majority, in Proc. 26th European Symposium on Research in Computer Security, Darmstadt, Germany, 2021. pp. 497–518,
[22] X. Ma, X. Sun, Y. Wu, Z. Liu, X. Chen, and C. Dong, Differentially private Byzantine-robust federated learning, IEEE Trans. Parallel Distrib. Syst., vol. 33, no. 12, pp. 3690–3701, 2022.

[23] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, Federated learning for mobile keyboard prediction, arXiv preprint arXiv: 1811.03604, 2018.
[24] Z. Wu, Q. Ling, T. Chen, and G. B. Giannakis, Federated variance-reduced stochastic gradient descent with robustness to Byzantine attacks, IEEE Trans. Signal Process., vol. 68, pp. 4583–4596, 2952.
[25] X. Gong, Y. Chen, Q. Wang, and W. Kong, Backdoor attacks and defenses in federated learning: State-of-the-art, taxonomy, and future directions, IEEE Wirel. Commun., vol. 30, no. 2, pp. 114–121, 2023.
[26] B. Hou, J. Gao, X. Guo, T. Baker, Y. Zhang, Y. Wen, and Z. Liu, Mitigating the backdoor attack by federated filters for industrial IoT applications, IEEE Trans. Ind. Inform., vol. 18, no. 5, pp. 3562–3571, 2022.
[27] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, How to backdoor federated learning, in Proc. of Twenty Third Int. Conf. on Artificial Intelligence and Statistics, Virtual Event, 2020, pp. 2938−2948.
[28] E. M. E. Mhamdi, R. Guerraoui, and S. Rouault, The hidden vulnerability of distributed learning in Byzantium, in Proc. of the 35th Int. Conf. on Machine Learning, Stockholm, Sweden, 2018, pp. 3521−3530.
[29] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, Practical secure aggregation for privacy-preserving machine learning, in Proc. 2017 ACM SIGSAC Conf. Computer and Communications Security, Dallas, TX, USA, 2017 pp. 1175–1191.
[30] S. Truex, N. Baracaldo, A. Anwar, T. Steinke, H. Ludwig, R. Zhang, and Y. Zhou, A hybrid approach to privacy preserving federated learning, in Proc. 12th ACM Workshop on Artificial Intelligence and Security, London, UK, 2019, pp. 1–11.
[31] Y. Li, Y. Zhou, A. Jolfaei, D. Yu, G. Xu, and X. Zheng, Privacy-preserving federated learning framework based on chained secure multiparty computing, IEEE Internet Things J., vol. 8, no. 8, pp. 6178–6186, 2021.
[32] Y. Miao, R. Xie, X. Li, X. Liu, Z. Ma, and R. H. Deng, Compressed federated learning based on adaptive local differential privacy, in Proc. 38th Annual Computer Security Applications Conference, Austin, TX, USA, 2022, pp. 159–170.
[33] G. Xu, H. Li, S. Liu, K. Yang, and X. Lin, VerifyNet: Secure and verifiable federated learning, IEEE Trans. Inf. Forensics Secur., vol. 15, pp. 911–926, 2020.
[34] J. So, B. Güler, and A. S. Avestimehr, Byzantine-resilient secure federated learning, IEEE J. Sel. Areas Commun., vol. 39, no. 7, pp. 2168–2181, 2021.
[35] X. Liu, H. Li, G. Xu, Z. Chen, X. Huang, and R. Lu, Privacy-enhanced federated learning against poisoning adversaries, IEEE Trans. Inf. Forensics Secur., vol. 16, pp. 4574–4588, 2021.
[36] L. Zhao, J. Jiang, B. Feng, Q. Wang, C. Shen, and Q. Li, SEAR: Secure and efficient aggregation for Byzantine robust federated learning, IEEE Trans. Dependable Secure Comput., vol. 19, no. 5, pp. 3329–3342, 2022.
[37] Z. Zhang, J. Li, S. Yu, and C. Makaya, SAFELearning: Enable backdoor detectability in federated learning with secure aggregation, arXiv preprint arXiv: 2102.02402, 2021.
[38] Y. Miao, Z. Liu, H. Li, K.-K R. Choo, and R. H. Deng, Privacy-preserving Byzantine-robust federated learning via blockchain systems, IEEE Trans. Inf. Forensics Secur., vol. 17, pp. 2848–2861, 2022.
[39] M. Shayan, C. Fung, C. J. M. Yoon, and I. Beschastnikh, Biscotti: A blockchain system for private and secure federated learning, IEEE Trans. Parallel Distrib. Syst., vol. 32, no. 7, pp. 1513–1525, 2021.