International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 11, Issue 4s, April-2026, http://ijmec.com/, ISSN: 2456-4265

Full Length Article

Blockchain-Assisted Privacy And Security Enhancement In Federated Learning

Abdul Rahman Anas[1],Mohammed Mahboob Pasha[2],Mohammed Ahtesham[3],Mrs. Syeda Bushra[4]
[1,2,3]B.E.Students; Department of Information Technology, ISL Engineering College, Hyderabad, India
[4]Assistant Professor; Department of Information Technology, ISL Engineering College, Hyderabad, India
Mail Id; sauban982@gmail.com , mohdmahboob1454@gmail.com , aryanmd280@gmail.com
Accepted 24-04-2026

Abstract
Federated Learning (FL) enables multiple clients to collaboratively train machine learning models without sharing raw data, thereby enhancing data privacy and security. However, traditional federated learning systems remain vulnerable to privacy leakage, poisoning attacks, and centralized server failures. To address these challenges, this project proposes a Blockchain-based Privacy-preserving and Secure Federated Learning (BPS-FL) framework. The proposed framework integrates threshold Paillier homomorphic encryption to achieve secure gradient aggregation while preserving client privacy. In addition, blockchain technology is incorporated to provide decentralized, transparent, and tamper-proof record management. A Byzantine-robust aggregation mechanism is also introduced to identify and mitigate malicious gradient updates without compromising data confidentiality. The proposed BPS-FL framework improves security, transparency, robustness, and trustworthiness in distributed learning environments while maintaining reliable model performance and scalability.

Keywords— *Federated Learning, Blockchain, Homomorphic Encryption, Privacy Preservation, Byzantine-robust Aggregation, Secure Aggregation, Machine Learning, Data Security, Distributed Systems.*

Introduction

Artificial Intelligence (AI) applications require large-scale datasets to train highly accurate machine learning models. However, privacy regulations, organizational restrictions, and concerns regarding sensitive information make centralized data collection increasingly difficult. Federated Learning (FL) addresses this challenge by enabling multiple clients to collaboratively train a shared global model while keeping raw data stored locally on client devices. This decentralized training approach significantly improves privacy protection and reduces the risk of direct data exposure.

Despite its advantages, federated learning still faces several important security and privacy challenges. During the training process, exchanged gradient updates may unintentionally reveal sensitive information about local datasets. In addition, malicious participants may launch poisoning attacks by injecting manipulated gradients into the aggregation process, thereby corrupting the global model and reducing overall accuracy. Traditional privacy-preserving techniques such as Differential Privacy (DP), Secure Multi-party Computation (SMC), and Homomorphic Encryption (HE) provide partial solutions to these problems, but they often suffer from limitations such as increased computational overhead, communication complexity, and reduced model performance.

To overcome these challenges, the proposed Blockchain-based Privacy-preserving and Secure Federated Learning (BPS-FL) framework combines blockchain technology with threshold Paillier homomorphic encryption to provide secure, decentralized, and privacy-preserving federated learning. The integration of blockchain eliminates centralized points of failure and ensures transparency, immutability, and traceability throughout the training process. The framework also introduces a Byzantine-robust aggregation mechanism capable of detecting malicious gradient updates while preserving client privacy and maintaining model accuracy.

Scope of the Project

The scope of the project focuses on developing a secure federated learning framework capable of preserving client data privacy through homomorphic encryption techniques. The framework is designed to detect and prevent poisoning attacks while ensuring secure aggregation of encrypted gradients. Blockchain technology is utilized to provide decentralized and immutable record management, thereby enhancing transparency, traceability, and system reliability. The proposed system is suitable for privacy-sensitive applications such as healthcare, Internet of

https://doi.org/10.63665/IJMEC.1104s.43
ISSN: 2456-4265
IJMEC 2026

Things (IoT), finance, and smart systems where secure distributed learning is essential.

Objectives

The primary objective of the project is to design and develop a privacy-preserving federated learning framework that ensures secure communication and reliable distributed model training. The project aims to protect local gradient updates using threshold Paillier homomorphic encryption and develop a Byzantine-robust aggregation mechanism capable of defending against malicious clients and poisoning attacks. Another important objective is to eliminate centralized failure points by integrating blockchain technology into the federated learning architecture. The framework also aims to maintain high model accuracy, scalability, and computational efficiency in real-world distributed environments.

Problem Statement

Traditional federated learning systems are vulnerable to several security and privacy threats, including privacy leakage, malicious gradient attacks, and centralized server failures. Existing privacy-preserving solutions often reduce model accuracy or introduce significant communication and computational overhead. Furthermore, the use of encrypted gradients makes the detection of malicious updates more challenging during aggregation. Therefore, there is a critical need for a secure and decentralized federated learning framework capable of preserving client privacy, resisting poisoning attacks, ensuring reliable model aggregation, and maintaining transparency and trustworthiness in distributed machine learning environments.

Table 1: Comparison of Existing Approaches with Proposed System

Aspect	Existing System	L mitations in Existing System	Proposed System
Architecture	Centralized FL server	Single point of failure	Decentralized blockchain framework
Privacy	Uses DP, SMC, and HE	High overhead and reduced accuracy	Threshold Paillier Encryption
Security	Basic aggregation methods	Cannot detect malicious gradients	Byzantine-robust aggregation
Data Integrity	Centralized storage	Risk of tampering	Immutable blockchain ledger
Reliability	Server-dependent system	Vulnerable to attacks	Distributed secure architecture
Communication Efficiency	Frequent client-server communication	High communication cost	Efficient secure aggregation
Transparency	Limited monitoring	Difficult to verify updates	Smart contract-based logging
Attack Resistance	Weak against poisoning attacks	Malicious updates affect model	Strong poisoning attack defense
Confidentiality	Partial privacy protection	Gradient leakage possible	Fully encrypted gradient sharing
Scalability	Limited robustness	Reduced efficiency at scale	Scalable secure FL framework

Project Description

The proposed project presents a secure encryption-based federated learning architecture integrated with blockchain technology. The framework is designed to ensure confidentiality, integrity, and decentralized trust management within distributed learning environments. By combining cryptographic mechanisms with blockchain-based storage and verification, the system enables secure communication and reliable data sharing among multiple participating clients while protecting

https://doi.org/10.63665/IJMEC.1104s.43
ISSN: 2456-4265
IJMEC 2026

sensitive information from unauthorized access and malicious attacks.

Modules of the System

The system consists of several important modules that collectively support secure federated learning operations. The User Interface Design module provides secure registration and login functionalities for users. Only authenticated users are permitted to access the system and perform authorized operations. This module ensures proper authentication and access control mechanisms.

The Committee Module is responsible for uploading encrypted files, approving client requests, and monitoring suspicious activities related to malicious users. Committee members act as trusted authorities that verify and manage secure transactions within the system.

The Client Module allows clients to search encrypted files, request cryptographic keys, decrypt authorized files, and securely download required data. Clients participate in federated learning while maintaining privacy and secure communication with other system components.

The Cloud Server Module functions as the administrative control layer of the system. It manages request approvals, monitors blocked or malicious users, and supports secure coordination between distributed entities and blockchain operations.

The Key Generation Center (KGC) is responsible for generating and distributing public and private cryptographic keys securely. It plays a vital role in maintaining encryption-based communication and secure data sharing among users.

The Smart Contract Module handles blockchain-related operations such as block creation, hashing, transaction validation, and integrity verification. Smart contracts ensure transparency, immutability, and automated execution of blockchain transactions within the framework.

Input and Output

The system accepts several types of input, including user login credentials, encrypted file uploads, key requests, and encrypted data transactions. These inputs are processed securely through cryptographic and blockchain-based mechanisms.

The output generated by the system includes encrypted files, generated cryptographic keys, blockchain transaction records, approved access requests, and secure file downloads. These outputs ensure confidentiality, integrity, and secure communication across the federated learning environment.

Algorithms Used

The project utilizes multiple algorithms to ensure security, privacy preservation, and data integrity. The Blockchain Algorithm maintains a decentralized ledger containing encrypted transactions and block hashes. It ensures immutability, transparency, traceability, and protection against unauthorized modifications.

The RSA Algorithm is used for asymmetric encryption through public and private key cryptography. It enables secure communication, encrypted file sharing, and secure authentication within the system.

The SHA-256 Algorithm generates unique hash values for files and transactions. These hash values help maintain data integrity, detect tampering attempts, and secure blockchain records against modification.

Requirements Engineering

The project requires both hardware and software resources to support secure federated learning and blockchain operations effectively. The hardware requirements include a Dual Core 2 Duo processor, 2 GB RAM, and 250 GB hard disk storage capacity. These resources are sufficient for executing encryption operations, blockchain processing, and distributed communication tasks.

The software requirements include J2EE technologies such as JSP and Servlets for front-end and backend web application development. MySQL 5.5 is used as the database management system, Eclipse serves as the integrated development environment (IDE), and Windows 7 is used as the operating system for deployment and execution.

Functional Requirements

The system provides several functional capabilities, including user authentication and authorization, secure file upload and download operations, cryptographic key generation and distribution, blockchain-based transaction recording, malicious client detection, and encrypted data management. These functionalities collectively ensure secure and reliable operation of the federated learning framework.

Non-Functional Requirements

The framework is designed to satisfy important non-functional requirements such as efficiency, security, reliability, and scalability. The system ensures efficient communication, secure data handling, and scalable distributed learning operations. Strong security mechanisms prevent unauthorized access, protect client privacy, and defend against malicious attacks.

Reliability is achieved through blockchain technology, which ensures immutable and tamper-proof transaction management. The system is also scalable and capable of supporting large numbers of participating clients and distributed nodes in real-world federated learning environments.

System Design

System design describes the structural and behavioral representation of the proposed Blockchain-based Privacy-preserving and Secure Federated Learning (BPS-FL) framework using

https://doi.org/10.63665/IJMEC.1104s.43
ISSN: 2456-4265
IJMEC 2026

various UML diagrams. These diagrams help illustrate the interaction between system components, data flow, communication mechanisms, and overall architecture of the framework. The design ensures secure communication, decentralized coordination, and efficient management of federated learning operations in distributed environments.

UML Diagrams

The Use Case Diagram represents the interactions between major entities in the system, including clients, committee members, the Key Generation Center (KGC), cloud servers, and smart contracts. The primary operations performed within the system include user login, key requests, encrypted file uploads, approval processes, secure downloads, and blockchain storage management. It provides a high-level overview of how users and system components interact within the distributed network.

The Class Diagram defines the structural organization of the system by identifying classes such as Client, Committee, KGC, Cloud Server, and Smart Contract along with their attributes and methods. It illustrates the relationships between these classes and explains how data and functionalities are organized within the framework.

The Object Diagram represents runtime interactions among system objects during secure communication and encrypted file handling. It demonstrates how individual objects collaborate dynamically during execution to perform various operations within the BPS-FL framework.

The Component Diagram illustrates the major software components of the system, including encryption modules, blockchain modules, cloud storage services, and user interface modules. It shows how these components are interconnected and how they collectively support secure federated learning functionality.

The Deployment Diagram describes the physical deployment of clients, servers, blockchain nodes, and cloud infrastructure across the distributed environment. It provides a clear representation of hardware and software deployment within the network.

The Sequence Diagram explains the sequential flow of interactions between system entities during operations such as file uploads, key generation, request approvals, encrypted aggregation, and secure downloads. It highlights the order in which messages and operations are executed.

The Collaboration Diagram focuses on the communication and coordination among distributed system entities. It demonstrates how clients, cloud servers, blockchain nodes, and smart contracts collaborate to complete secure federated learning tasks efficiently.

The State Diagram represents the lifecycle of clients within the system, beginning from user login and authentication to secure file access, encrypted communication, and malicious-user detection. It illustrates the changes in system states during various operations.

The Activity Diagram describes the workflow execution process from authentication to secure file access and data processing. It visually represents decision-making processes, control flow, and parallel operations within the system.

The Data Flow Diagram (DFD) illustrates the movement of encrypted data, requests, cryptographic keys, and model updates between different system modules. It helps explain how information is processed securely throughout the framework.

The ER Diagram defines the relationships among major entities such as Client, Committee, KGC, Blockchain, and Files. It provides a database-level representation of entity relationships and data organization within the system.

System Architecture

The BPS-FL architecture combines several core components, including Federated Learning clients, a blockchain network, a Key Generation Center (KGC), cloud servers, and smart contracts. In this architecture, encrypted gradients generated by participating clients are securely aggregated while blockchain technology maintains immutable training records and ensures transparency. Smart contracts automate verification and validation processes, while the cloud infrastructure supports storage and communication among distributed nodes. This architecture provides secure, decentralized, and privacy-preserving federated learning suitable for real-world distributed AI applications.

Software Specification

The proposed project is implemented using Java technologies integrated with blockchain and cryptographic mechanisms. Java was selected because of its platform independence, object-oriented architecture, scalability, and strong security capabilities. The language also provides multithreading support, portability, and efficient memory management, making it suitable for distributed and secure applications.

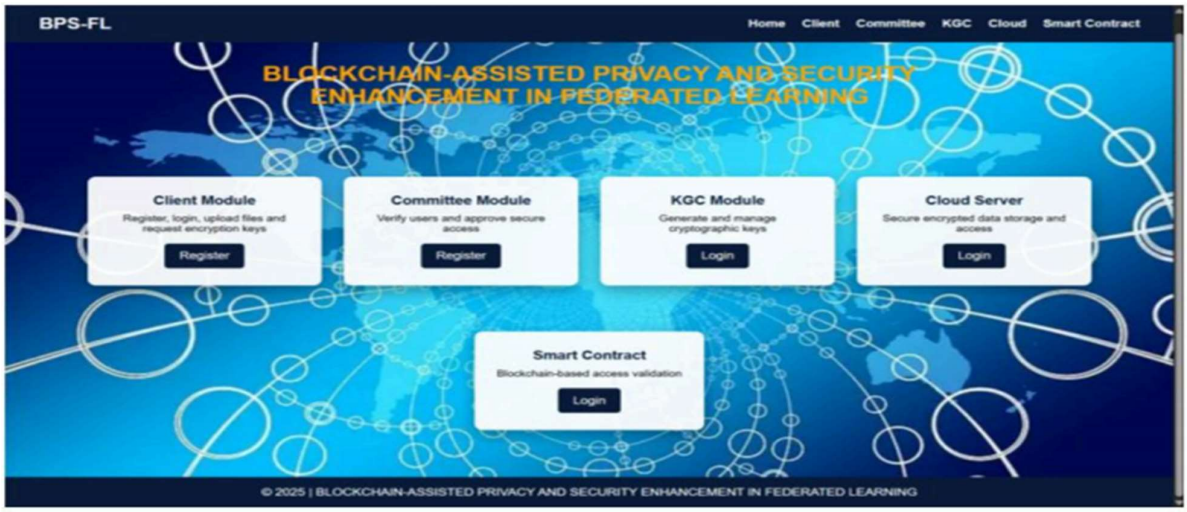
J2EE technologies such as JSP and Servlets are used to develop the web-based application interface and backend processing modules. These technologies support dynamic content generation, secure client-server communication, and efficient database interaction. Java also enables reliable execution of cryptographic operations required for encryption, secure aggregation, and blockchain integration.

Java Swing is used to design graphical user interface components for secure login, user registration, and management modules. The Java Collection Framework is utilized for efficient storage, retrieval,

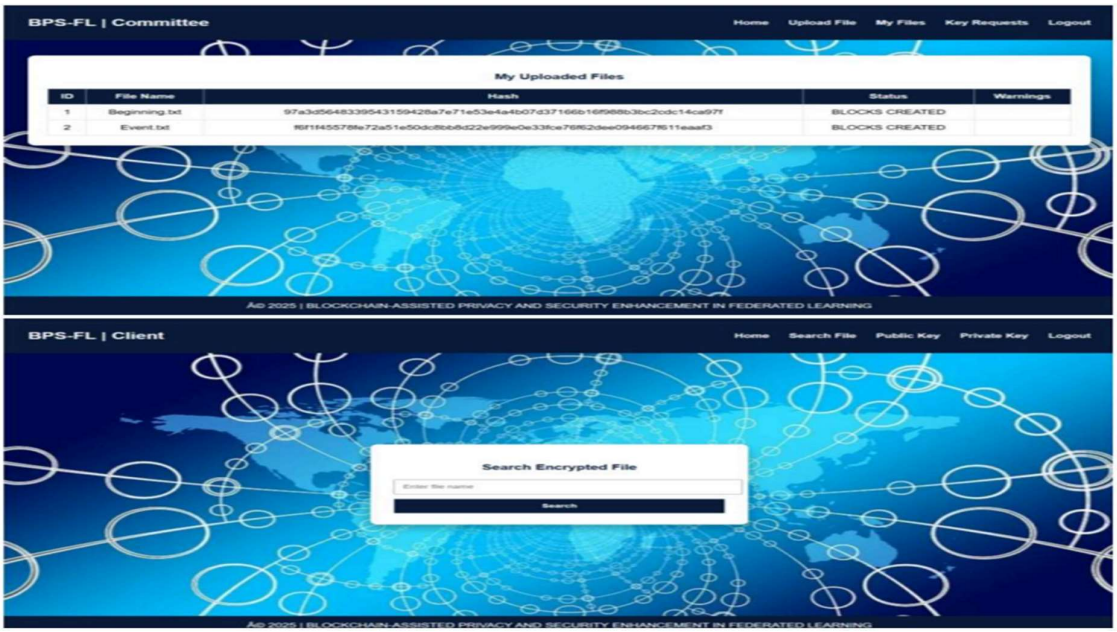
https://doi.org/10.63665/IJMEC.1104s.43
ISSN: 2456-4265
IJMEC 2026

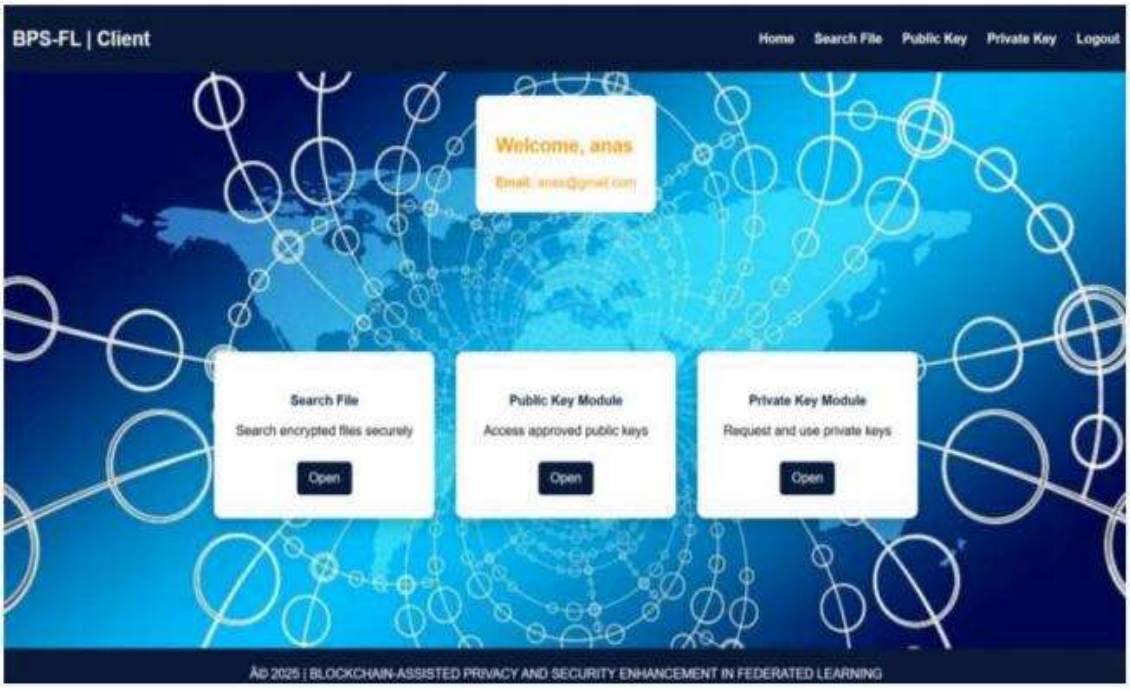
and manipulation of system data, including client records, blockchain metadata, and transaction logs. The proposed Blockchain-based Privacy-preserving and Secure Federated Learning (BPS-FL) framework successfully enhances security, privacy, and reliability in federated learning environments. By integrating blockchain technology, homomorphic encryption, and Byzantine-robust aggregation mechanisms, the system effectively protects sensitive client data while defending against malicious attacks and poisoning attempts. The framework ensures decentralized trust, secure model aggregation, and tamper-proof record management, making it a practical and reliable solution for secure distributed artificial intelligence applications in real-world environments.

Snapshots



https://doi.org/10.63665/IJMEC.1104s.43
ISSN: 2456-4265
IJMEC 2026





Software Testing

Software testing is an essential process used to identify errors, defects, and weaknesses in a software application. The primary objective of testing is to ensure that the developed system functions correctly according to specified requirements and user expectations. Testing verifies the functionality, reliability, performance, and security of the software before deployment.

The process involves executing the software under various conditions to detect bugs and ensure smooth operation without failures. Different testing techniques are applied to validate individual components, integrated modules, and the complete system. Each type of testing focuses on specific aspects of software quality and system functionality.

Development Methodologies

https://doi.org/10.63665/IJMEC.1104s.43
ISSN: 2456-4265
IJMEC 2026

The testing process begins with the preparation of a comprehensive test plan designed to evaluate the overall functionality and unique features of the application across multiple platforms and environments. Proper quality assurance procedures are followed throughout the testing phase to ensure that the software remains reliable, secure, and error-free.

The testing methodology confirms that the application satisfies all requirements specified in the Software Requirement Specification (SRS) document. It also ensures that the system performs efficiently, securely, and accurately under different operating conditions.

Future Enhancement

Although the proposed Blockchain-based Privacy-preserving and Secure Federated Learning (BPS-FL) framework provides strong privacy protection, robustness against poisoning attacks, and reliable performance, several improvements can further enhance its scalability and efficiency.

The current secure distance computation mechanism may introduce additional computational overhead when the number of participating clients increases significantly. Future research can focus on optimizing these computations using lightweight cryptographic techniques and more efficient aggregation methods to reduce processing time and communication latency in large-scale deployments.

In addition, future enhancements may include improving energy efficiency, supporting dynamic client participation, and integrating advanced attack detection mechanisms to strengthen the framework against evolving cybersecurity threats.

Conclusion

In this project, a Blockchain-based Privacy-preserving and Secure Federated Learning (BPS-FL) framework was proposed to address major privacy and security challenges in federated learning environments. The framework protects client data privacy by encrypting local gradients using threshold homomorphic encryption, enabling secure aggregation without exposing sensitive information.

The proposed Byzantine-robust aggregation protocol effectively identifies and minimizes the impact of malicious or poisoned client updates while maintaining high global model accuracy. By integrating blockchain technology, the BPS-FL framework eliminates single-point failures and ensures immutability, transparency, and traceability throughout the learning process.

Experimental analysis demonstrates that the framework maintains high classification accuracy with acceptable computational overhead, even in the presence of malicious participants. Therefore, the proposed BPS-FL system provides a secure, reliable, and practical solution for privacy-preserving federated learning in adversarial and real-world distributed environments.

References

1) L. Peng et al., “Federated Graph Learning with Network Inpainting for Population-Based Disease Prediction,” 2023.
2) R. Zhao et al., “Semi-supervised Federated-learning-based Intrusion Detection Method for IoT,” 2023.
3) X. Gong et al., “Backdoor Attacks and Defenses in Federated Learning,” 2023.
4) X. Ma et al., “Differentially Private Byzantine-robust Federated Learning,” 2022.
5) L. Zhao et al., “Secure and Efficient Aggregation for Byzantine-robust Federated Learning,” 2022.
6) Y. Dong et al., “Oblivious Defender for Private Byzantine-robust Federated Learning,” 2021.
7) X. Liu et al., “Privacy-enhanced Federated Learning against Poisoning Adversaries,” 2021.
8) Y. Li et al., “Privacy-preserving Federated Learning Framework based on Chained Secure Multiparty Computing,” 2021.
9) J. H. Bell et al., “Secure Single-server Aggregation with Polylogarithmic Overhead,” 2020.
10) Y. Li et al., “Toward Secure and Privacy-preserving Distributed Deep Learning in Fog-cloud Computing,” 2020.

https://doi.org/10.63665/IJMEC.1104s.43
ISSN: 2456-4265
IJMEC 2026