IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A PROXY RE-ENCRYPTION APPROACH FOR SECURE DATA SHARING BASED ON BLOCKCHAIN

[1]Mrs.R.Deepa, Assistant Professor

[2]Ms Yamuna P, (II-MCA)

[1,2,3] Master of Computer Applications

[1,2,3] Er.Perumal Manimekalai College of Engineering, Hosur, TamilNadu, India.

Abstract**:** Cloud computing has become an essential platform for scalable data storage and remote accessibility. However, outsourcing sensitive information to cloud environments introduces serious security challenges such as unauthorized access, data tampering, lack of transparency, and dependency on cloud service providers. Traditional encryption mechanisms provide only limited protection because data owners must manually manage encryption, access control, and secure sharing operations. To address these limitations, this paper proposes a Proxy Re-Encryption based secure cloud data sharing framework integrated with blockchain technology. The proposed architecture introduces a dedicated cryptographic agent named CryptoProxy, which performs delegated re-encryption operations on behalf of the data owner without exposing plaintext information to the cloud service provider. The framework combines Delegated Public-Key Re-Encryption, blockchain-based integrity verification, SHA-256 hashing, and immutable auditing to ensure confidentiality, accountability, and tamper resistance. Smart contracts record all file operations and access events within a private blockchain network, enabling transparent and verifiable auditing. The proposed system minimizes trust dependency on the cloud provider, supports secure multi-user data sharing, and enhances end-to-end confidentiality across the complete cloud data lifecycle. Experimental analysis demonstrates that the framework provides efficient file access management, secure re-encryption, integrity preservation, and scalable cloud governance for modern distributed environments.

Keywords— Proxy Re-Encryption, Blockchain, Cloud Security, CryptoProxy, Secure Data Sharing, SHA-256, Cloud Governance.

I. INTRODUCTION

Cloud computing has revolutionized the storage and management of digital information by providing scalable infrastructure, flexible resource allocation, and remote accessibility. Organizations and individuals increasingly depend on cloud storage systems for handling sensitive documents, enterprise records, and collaborative data operations. Despite these advantages, outsourcing confidential information to third-party cloud service providers introduces several security concerns including unauthorized access, insider attacks, data leakage, and integrity violations.

Most traditional cloud security models rely on centralized encryption and access control

mechanisms managed by the cloud service provider. Such approaches create trust dependency on the provider because encrypted data, access logs, and user management are controlled centrally. Furthermore, conventional encryption-based file sharing requires repeated decryption and re-encryption whenever multiple users request access, increasing computational overhead and operational complexity for data owners.

To overcome these limitations, this work proposes a secure cloud data governance architecture based on Delegated Public-Key Proxy Re-Encryption integrated with blockchain technology. The proposed system introduces a software agent named CryptoProxy that performs delegated cryptographic operations on behalf of the data owner. CryptoProxy securely transforms encrypted files into user-specific ciphertexts without exposing plaintext information to the cloud provider.

Blockchain technology is integrated into the framework to provide tamper-evident auditing and integrity verification. SHA-256 hash values of encrypted files and all major access events are recorded on a private blockchain through smart contracts. This mechanism guarantees transparency, accountability, and non-repudiation while preventing unauthorized modification of system logs.

The proposed architecture ensures secure multi-user file sharing, owner-controlled access management, independent integrity verification, and scalable cloud security while minimizing reliance on centralized trust models.

II. RELATED WORK

Several researchers have proposed cloud security frameworks to improve confidentiality, integrity, and secure data sharing in outsourced storage environments.

Aminuddin Mohd Kama proposed a privacy-preserving keyword search model for encrypted cloud storage using secret sharing techniques and secure computation methods. The framework reduced computational overhead while maintaining secure access control in distributed cloud systems. However, the model depended heavily on multiple server coordination and lacked immutable auditing support.

Daniel Morales and Javier Lopez introduced a lightweight secret-sharing mechanism suitable for resource-constrained IoT systems. Their approach reduced communication overhead through Oblivious Sharing Re-Encryption protocols. Although the framework improved scalability, it focused primarily on constrained devices and did not address cloud-level auditing and integrity management.

Longbo Han and his research team proposed a lattice-based blockchain infrastructure for trusted cloud data management. Their work enhanced verification efficiency and quantum-resistant security using lattice-based cryptographic functions. However, the system introduced higher computational complexity and implementation overhead.

Jie Zhang developed a reputation-backed auditable sharing model using blockchain technology. The framework improved decentralized trust and secure auditing for distributed data-sharing systems. Nevertheless, the model relied on complex reputation management mechanisms that increased operational complexity.

Existing cloud security approaches still face major limitations including centralized trust dependency, manual key management, lack of transparent auditing, plaintext exposure during sharing operations, and limited scalability in multi-user environments. The proposed framework addresses these challenges through delegated proxy re-encryption, CryptoProxy automation, and blockchain-integrated integrity verification.

III. PROPOSED SYSTEM

The proposed system provides a secure cloud data sharing architecture using Delegated Public-Key Proxy Re-Encryption, blockchain-based auditing, and automated cryptographic processing through CryptoProxy.

A. Delegated Public-Key Proxy Re-Encryption

Proxy Re-Encryption enables secure delegation of data access without revealing plaintext information to intermediaries. Files are initially encrypted using the data owner's public key before being uploaded to cloud storage. When an authorized user requests access, CryptoProxy decrypts the ciphertext using the owner's private key and re-encrypts it using the requesting user's public key.

This mechanism ensures that:

- The cloud service provider never accesses plaintext data.
- Only authorized users can decrypt shared files.
- Data owners are not required to manually re-encrypt files.
- Multi-user access becomes scalable and efficient.

B. CryptoProxy Agent Architecture

CryptoProxy acts as an automated software agent responsible for delegated cryptographic operations within the proposed framework.

The major responsibilities of CryptoProxy include:

- Secure storage of owner key pairs and user public keys.
- Validation of access permissions.
- Delegated decryption and re-encryption operations.
- Secure delivery of transformed ciphertexts.
- Recording cryptographic events on the blockchain.
- Reducing operational overhead for data owners.

CryptoProxy never exposes plaintext information to the cloud service provider and performs all cryptographic processing within the owner-controlled security boundary.

C. Blockchain-Based Integrity Verification

A private blockchain network is integrated into the framework to provide tamper-evident integrity verification and immutable auditing.

The system performs the following operations:

- Generates SHA-256 hash values for encrypted files.
- Records hashes on blockchain smart contracts.
- Logs uploads, access requests, and file updates.
- Maintains immutable audit trails.
- Detects unauthorized modifications.

This mechanism ensures transparency, accountability, and independent verification of all cloud storage activities.

IV. SYSTEM ARCHITECTURE

The proposed architecture consists of multiple interconnected modules that collectively provide secure cloud governance.

A. Data Owner Module

The Data Owner uploads encrypted files to cloud storage and defines access permissions for authorized users. Files are encrypted using the owner's public key before upload.

B. Data User Module

Authorized users request encrypted files from the cloud system. After approval, users receive re-encrypted ciphertexts that can be decrypted locally using their private keys.

C. Cloud Service Provider

The Cloud Service Provider stores encrypted files and metadata but never accesses plaintext information. The CSP acts only as a storage entity.

D. CryptoProxy Module

CryptoProxy performs delegated decryption and re-encryption operations while enforcing secure access control policies.

E. Blockchain Module

The blockchain module stores SHA-256 hash values and immutable transaction records for auditing and integrity verification.

F. Integrity Verification Module

This module compares newly generated hashes with blockchain records to detect unauthorized modifications or tampering.

V. METHODOLOGY

The methodology of the proposed framework consists of multiple secure processing stages.

A. User Registration and Authentication

Data Owners and Data Users register securely within the system using authentication credentials. Public-private key pairs are generated during registration.

B. File Encryption and Upload

The Data Owner encrypts files locally using the owner's public key before uploading them to the cloud. SHA-256 hash values are generated and recorded on the blockchain.

C. Access Request Validation

Data Users submit file access requests through the cloud dashboard. The system validates permissions based on owner-defined policies.

D. Delegated Re-Encryption

CryptoProxy retrieves encrypted files, decrypts them using the owner's private key, and re-encrypts them using the requesting user's public key.

E. Secure File Processing

Authorized users decrypt the received ciphertext locally using their private keys, perform permitted modifications, and re-upload encrypted files.

F. Blockchain Auditing

All operations including uploads, access requests, re-encryption events, and integrity verification results are

REFERENCES

[1] A. A. A. M. Kama, "Privacy-Preserving Keyword Search With Access Control," IEEE Access, 2025.

[2] D. Morales and J. Lopez, "Dynamic Secret Sharing Mechanism," 2025.

[3] L. Han et al., "Blockchain Infrastructure for Trusted Data Management," 2025.

[4] J. Zhang, "Auditable Model for Cohort Data Sharing," 2025.

[5] I. Bashir, Mastering Blockchain, Packt Publishing, 2018.

[6] M. Grinberg, Flask Web Development, O'Reilly Media, 2018.