IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Cloud Data Security Using DAES Encryption And Blockchain Key Management

Yash S. Shirude[1], Juned F. Pathan[2], Gaurav B. Tidke[3],

Gautam I. Lodha[4], Suvarna V. Somvanshi[5]

[1234]Students, 5Assistant Professor

Department of Computer Engineering,

Pune Vidyarthi Griha's College of Engineering, Nashik, Maharashtra, India

Abstract: As the reliance on cloud computing continues to grow, the security of sensitive data stored in cloud environments has become increasingly critical. This project presents a robust solution for enhancing cloud data security through the integration of Dynamic AES Encryption, Blockchain-based Key Management, and the Elliptic Curve Cryptography (ECC) algorithm. DAES encryption extends the traditional AES encryption by distributing the encryption process across multiple nodes in a cloud environment. This approach not only improves security by eliminating single points of failure but also enhances resilience against various cyber threats. By encrypting data using DAES before it is uploaded to the cloud, we ensure that only authorized users can access the sensitive information, maintaining confidentiality and integrity. To manage encryption keys effectively, this project utilizes a blockchain-based key management system. Blockchain technology provides a decentralized and tamper-proof ledger for storing and distributing encryption keys. Each key generation, usage, and access request is recorded on the blockchain, creating an immutable audit trail that enhances accountability and transparency. This decentralized approach mitigates the risks associated with traditional centralized key management systems, ensuring that keys are securely distributed only to authorized users.

Index Terms - Blockchain-based Key Management, Dynamic Advanced Encryption Standard, Elliptic Curve Cryptography

I. INTRODUCTION

As organizations increasingly migrate to cloud computing environments, the security of sensitive data stored in the cloud has emerged as a top priority. Traditional security measures often fall short in addressing the unique challenges associated with cloud data storage, such as unauthorized access, data breaches, and centralized key management vulnerabilities. To effectively protect sensitive information and ensure its integrity and confidentiality, there is a growing need for advanced security solutions. This project aims to address these concerns through the integration of Dynamic Advanced Encryption Standard (DAES) encryption, Blockchain-based Key Management, and the Elliptic Curve Cryptography (ECC) algorithm. DAES Encryption enhances traditional AES encryption by distributing the encryption process across multiple nodes within a cloud infrastructure. This Dynamic approach not only improves the robustness of data security by mitigating the risk of single points of failure but also enhances resilience against various cyber threats. By encrypting data before it is uploaded to the cloud, DAES ensures that sensitive information remains confidential and accessible only to authorized users, thereby protecting against unauthorized access and potential data breaches. In parallel with DAES encryption, this project employs a Blockchain-based Key Management system. Blockchain technology offers a decentralized, tamper-proof ledger that facilitates secure storage and distribution of encryption keys. By recording every key generation, usage, and access request on the blockchain, we create an immutable audit trail that enhances accountability and transparency. This decentralized framework addresses the vulnerabilities associated with traditional centralized key management systems, ensuring that encryption keys are Dynamic

securely and only to authorized users, reducing the risk of key compromise. To further bolster security, the project incorporates the Elliptic Curve Cryptography (ECC) algorithm for secure key exchange processes. ECC is known for its high level of security paired with relatively small key sizes, making it particularly suitable for resource-constrained environments like cloud services. By leveraging ECC for secure key exchange, we ensure that encryption keys are transmitted confidentially, safeguarding against interception or unauthorized access during transmission. The integration of DAES encryption, blockchain key management, and ECC creates a comprehensive framework for cloud data security that addresses the critical challenges associated with sensitive data storage and access. This project seeks to provide a secure, efficient, and scalable solution for managing sensitive information in cloud environments, enhancing data confidentiality, integrity, and accessibility. Ultimately, by adopting these advanced technologies, we aim to foster greater user trust in cloud services, paving the way for wider adoption of cloud computing across various sectors, from finance and healthcare to government and education.

II. Literature Review

Cloud computing has emerged as the dominant platform for data storage and delivery. However, with its growth, concerns over data confidentiality, key management, and secure transmission have intensified. Several researchers have proposed robust cryptographic and decentralized frameworks to address these challenges. This review critically analyzes existing works around DAES encryption, blockchain-based key management, Elliptic Curve Cryptography (ECC), and their integration for cloud data security.

1. DAES Encryption Techniques

Chan and Chen [1] proposed a distributed version of the AES algorithm (DAES) for securing cloud data. Their model distributes the encryption process across multiple nodes, reducing vulnerability to single-point failures. Building on this, Patel and Patel [2] validated the practical feasibility of DAES in cloud storage environments. Their findings confirmed that DAES not only enhances data confidentiality but also maintains performance during encryption and decryption tasks.

2. Blockchain-Based Key Management

Key management remains a critical challenge in cloud security. Zheng and Wang [3] addressed this by introducing a blockchain-based key management system. Their system offers a tamper-proof, decentralized ledger that records each key operation—generation, access, and usage—ensuring high transparency and accountability. Similarly, Li et al. [4] emphasized integrating blockchain with cloud storage to provide secure and verifiable key exchanges, enabling real-time auditing and preventing unauthorized access.

3. Elliptic Curve Cryptography (ECC) for Key Exchange

ECC has emerged as a lightweight and secure alternative for public key cryptography. Menezes et al. [5] laid its foundational principles, highlighting its high security per bit ratio compared to RSA. Later, Nithya and Karthikeyan [6] demonstrated ECC's application in cloud systems for efficient and secure communication. Their work emphasized ECC's low computational requirements, making it ideal for cloud platforms that must handle large-scale, fast operations.

4. Integrated Security Frameworks

Recognizing the limitations of single-method solutions, researchers have proposed integrated models. Bashir and Awan [7] presented a hybrid security framework combining DAES encryption, blockchain key management, and ECC for end-to-end secure data storage. The framework proved resilient against unauthorized access while ensuring transparency and decentralization. Cheng et al. [8] developed a similar integrated cloud security architecture and validated it through simulations, reporting significant improvements in confidentiality, auditability, and performance.

5. Recent Trends and Future Directions

Khan and Khan [9] offered a comprehensive review of emerging trends in cloud security. They noted the growing shift toward multi-layered, adaptable, and decentralized security frameworks. Their study suggests future research should focus on:

- Optimizing the performance of integrated security systems.
- Achieving cross-platform compatibility between various cloud providers.
- Developing user-centric interfaces for key and data control.
- Improving scalability while minimizing cost and latency in real-time applications.

The literature collectively shows that combining DAES encryption, blockchain for key management, and ECC for secure exchange forms a comprehensive and future-ready security solution for cloud platforms. However, challenges like implementation complexity, cost, and scalability still need active research attention.

III. SYSTEM OVERVIEW

The proposed system aims to provide an advanced, secure, and scalable solution for protecting sensitive data stored in cloud environments. It leverages Dynamic Advanced Encryption Standard (DAES) for data confidentiality, Blockchain-based Key Management for decentralized and tamper-proof key control, and Elliptic Curve Cryptography (ECC) for secure key exchange. The system architecture is modular and includes six main components, as explained below:

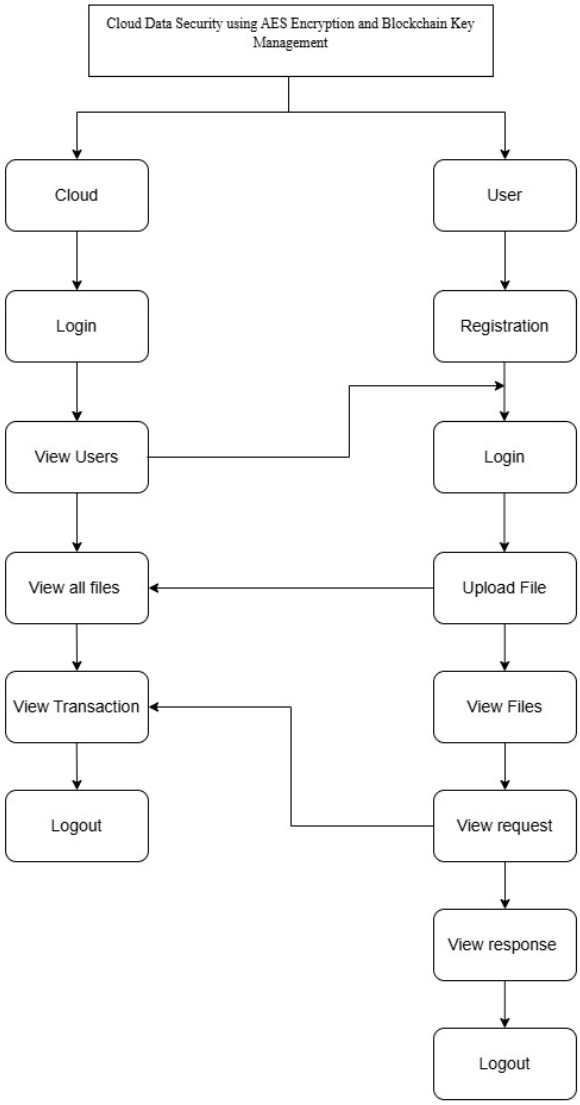

Figure 3.1: Flowchart Diagram

1. User Interface Module

This module provides an interactive platform for users to:

- Register and authenticate securely,
- Upload and download files to/from the cloud,
- View audit logs and manage encryption keys.

It is designed for usability and includes multi-factor authentication for enhanced security.

2. Data Encryption Module (DAES)

This is the core module that handles encryption and decryption operations using the DAES algorithm. The main tasks include:

- Data Splitting: The input file is divided into smaller chunks for distributed processing.
- Encryption: Each chunk is encrypted independently using symmetric keys generated per chunk.
- Reassembly: The encrypted chunks are recombined into one secure file for cloud storage.

DAES offers high resilience by distributing the encryption workload, minimizing risks of single-point attacks.

3. Blockchain Key Management Module

This module implements a private blockchain ledger to:

- Generate and register encryption keys with metadata,
- Store keys in a tamper-proof and decentralized way,
- Control access using smart contracts,
- Audit all actions, including key generation, sharing, and usage.

This design ensures that only authorized users can access the encryption keys.

4. Elliptic Curve Cryptography (ECC) Module

ECC is used for securely exchanging symmetric keys during the encryption/decryption process. Functions include:

- Generating public-private key pairs for users,
- Encrypting symmetric keys using the recipient's ECC public key,
- Ensuring secure and lightweight key transmission across the network.

ECC provides strong encryption with minimal computational cost, making it suitable for scalable cloud systems.

5. User Authentication and Access Control Module

This module ensures that:

- Only verified users can log in or access resources,
- Role-Based Access Control (RBAC) is enforced (e.g., admin vs normal user),
- Sessions are securely maintained using tokens or session cookies.

6. Audit and Monitoring Module

This module provides real-time monitoring and logging of:

- User activities (login, upload/download),
- Key transactions (creation, request, access),
- Suspicious or unauthorized attempts (trigger alerts).

Logs are stored on the blockchain for immutability and accountability.

7. Data Storage Module

- Stores the final encrypted files securely on the cloud (e.g., AWS S3, Azure).
- Ensures redundancy, backups, and scalable storage.
- Integrates with encryption/decryption pipeline for seamless user access.

IV. TECHNOLOGIES USED

Dynamic Advanced Encryption Standard (DAES): DAES is an enhanced version of the AES algorithm that splits data into chunks and encrypts them in a distributed manner. It reduces the risk of single-point failures and enhances the security and performance of cloud-based systems.

Blockchain Technology: A private blockchain is implemented to manage and store encryption keys securely. It ensures tamper-proof logging of key generation, access, and usage. Blockchain provides transparency, decentralization, and auditability to the key management process.

Elliptic Curve Cryptography (ECC): ECC is utilized for secure key exchange between users and the system. It offers strong encryption with smaller key sizes, making it highly efficient and suitable for cloud environments with limited resources.

.NET Framework: The .NET Framework is used as the backbone of the application development. It supports the integration of cryptographic modules, user interface design, and system functionality across both client and server layers.

Visual Studio: Visual Studio serves as the development environment for implementing the system using C#. It offers powerful debugging, UI design tools, and seamless integration with .NET libraries.

C# Programming Language: The core logic of the system, including encryption, decryption, blockchain interaction, and key handling, is written in C#. Its object-oriented structure makes it suitable for developing secure and scalable applications.

Cloud Storage Platforms: Encrypted files are stored on cloud platforms such as AWS S3, Azure, or Google Cloud. These platforms offer scalability, reliability, and secure access for encrypted data.

Multi-Factor Authentication (MFA): MFA is integrated into the system to strengthen user login security. It combines traditional password-based authentication with additional layers such as OTP or biometric verification.

Audit Logging and Monitoring: The system includes audit logging to track all critical activities like file upload/download and key usage. These logs are stored on the blockchain to prevent tampering and enable transparent monitoring.

Testing and Documentation Tools: Various testing tools are used to validate the system's functionality, while documentation and plagiarism checking tools ensure report integrity and originality.

V. MODULES IMPLEMENTED

The proposed system consists of multiple interconnected modules that collaboratively ensure the security, privacy, and integrity of data stored on the cloud. Each module handles a specific responsibility, from user interaction to encryption, storage, and secure key management.

1) User Registration and Login Module

This module is responsible for securely onboarding users and managing authentication.

- Features:
  - New user sign-up with email, username, and password.
  - Secure login with encrypted credentials.
  - Multi-Factor Authentication (MFA) using OTP or mobile verification.
- Purpose: Prevents unauthorized access to the system.
- Working: User data is stored securely after hashing passwords. Login sessions are managed using token-based authentication.

2) DAES Encryption and Decryption Module

This is the heart of the system that ensures confidentiality using enhanced AES.

- Features:
  - Splits files into fixed-size blocks (e.g., 128-bit chunks).
  - Encrypts each block independently using AES with different keys.
  - Parallel processing for speed optimization.
- Purpose: Protects data using high-level symmetric encryption.
- Working: Each encrypted block is tagged with metadata. On download, the blocks are decrypted in the correct sequence.

3) Blockchain-Based Key Management Module

This module securely manages encryption keys and ensures traceability.

- Features:
  - Key generation for each encryption session.
  - Storage of key hash + metadata on private blockchain.
  - Smart contracts for permission control.
- Purpose: Avoids centralized key storage, making the system tamper-proof.
- Working: Every action like key creation, request, or usage is recorded as a block entry, which is immutable and verifiable.

4) ECC-Based Key Exchange Module

This module enables secure transmission of encryption keys over insecure channels.

- Features:
  - ECC public/private key pair generation per user.
  - AES symmetric key encryption using ECC public key.
  - Key decryption using ECC private key.
- Purpose: Prevents interception of encryption keys during transfer.
- Working: ECC ensures only the intended recipient can decrypt the AES key.

5) Cloud Storage Module

Encrypted data is stored in secure cloud servers for availability and reliability.

- Features:
  - Secure APIs to communicate with cloud (e.g., AWS, Azure).
  - Token-based access and permission control.
  - Redundancy and failover handling.
- Purpose: Ensure secure, scalable, and reliable file storage.
- Working: Files are stored with a unique encrypted name. Access requires correct user token and matching key.

6) Audit Log and Monitoring Module

Tracks all activities performed in the system for accountability.

- Features:
  - Logging of all user actions (login, upload, key access).
  - Log hashing and storage on blockchain.
  - Admin viewing and analysis tools.
- Purpose: Helps in detecting misuse or unauthorized access attempts.
- Working: Every log is timestamped and secured on blockchain to prevent tampering.

7) Admin Panel and Access Control Module

Provides advanced management tools for system administrators.

- Features:
  - User role assignment (admin, user, read-only).
  - Viewing system analytics and logs.
  - Manual key override or reset in emergency cases.
- Purpose: Gives oversight to manage system integrity and data access.
- Working: Admin rights are verified through secure login; logs and access records are displayed in dashboard format.

The modular structure of the system ensures that each security layer is isolated, manageable, and efficient. DAES ensures data is unreadable without keys, blockchain makes keys unchangeable and verifiable, and ECC ensures safe key sharing. These modules together build a complete, secure cloud data environment.

VI. RESULTS

Figure 1: Home Page Displaying Secure Cloud Services and Dashboard Options

Main Interface: Admin dashboard showing access options like file upload, view users, view files, and transactions.
Login Module: Secure login screen with options for new registration and user authentication.
User Registration: Interface for adding new users with secure credentials and contact details
Upload Module: File selection interface with DAES-based block division for encryption.
File Management: Interface to view uploaded files, hash values, and file sharing using public key.
Transaction Log: Display of blockchain-backed file upload transactions with hash verification.

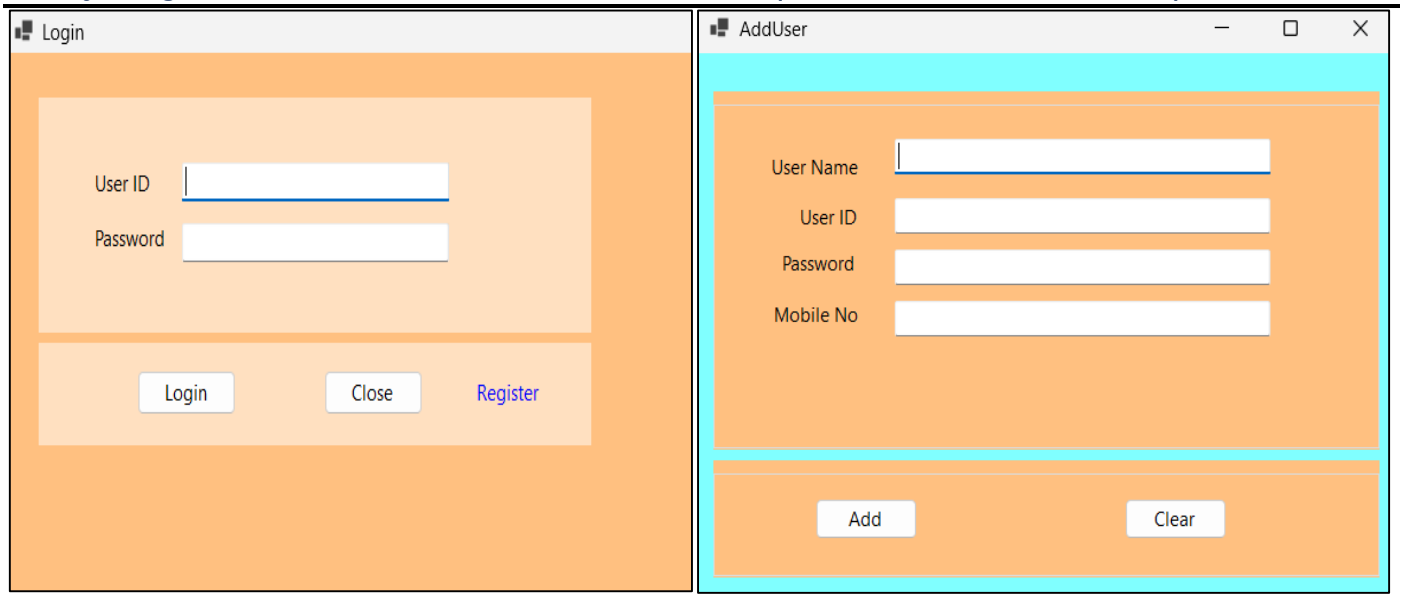

Figure 1: Authentication and Registration Interfaces for Secure Access Control

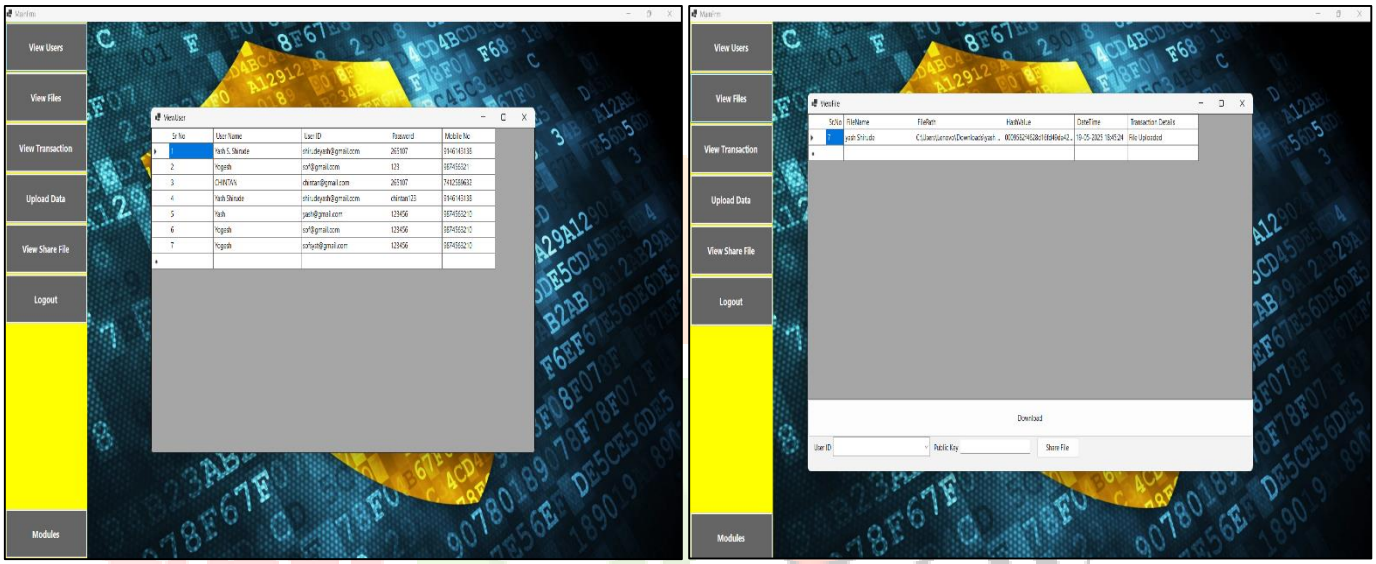

Figure 1: Admin Dashboard Showing User and File Management Interfaces

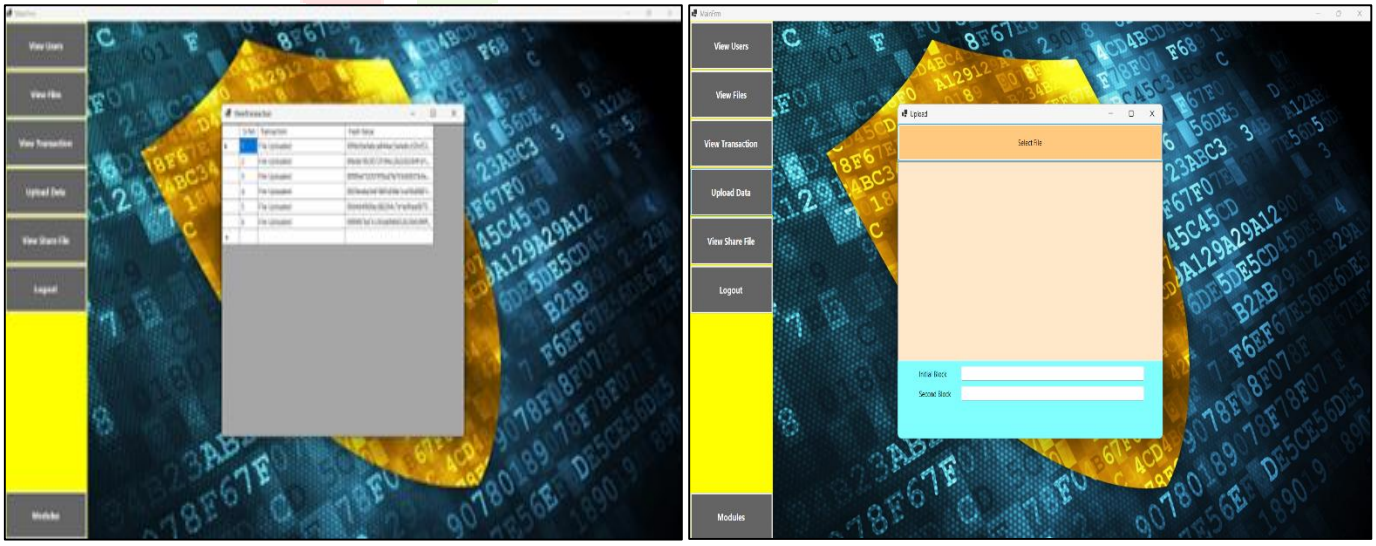

Figure 1: Upload and Transaction Interfaces for File Security and Audit Logging

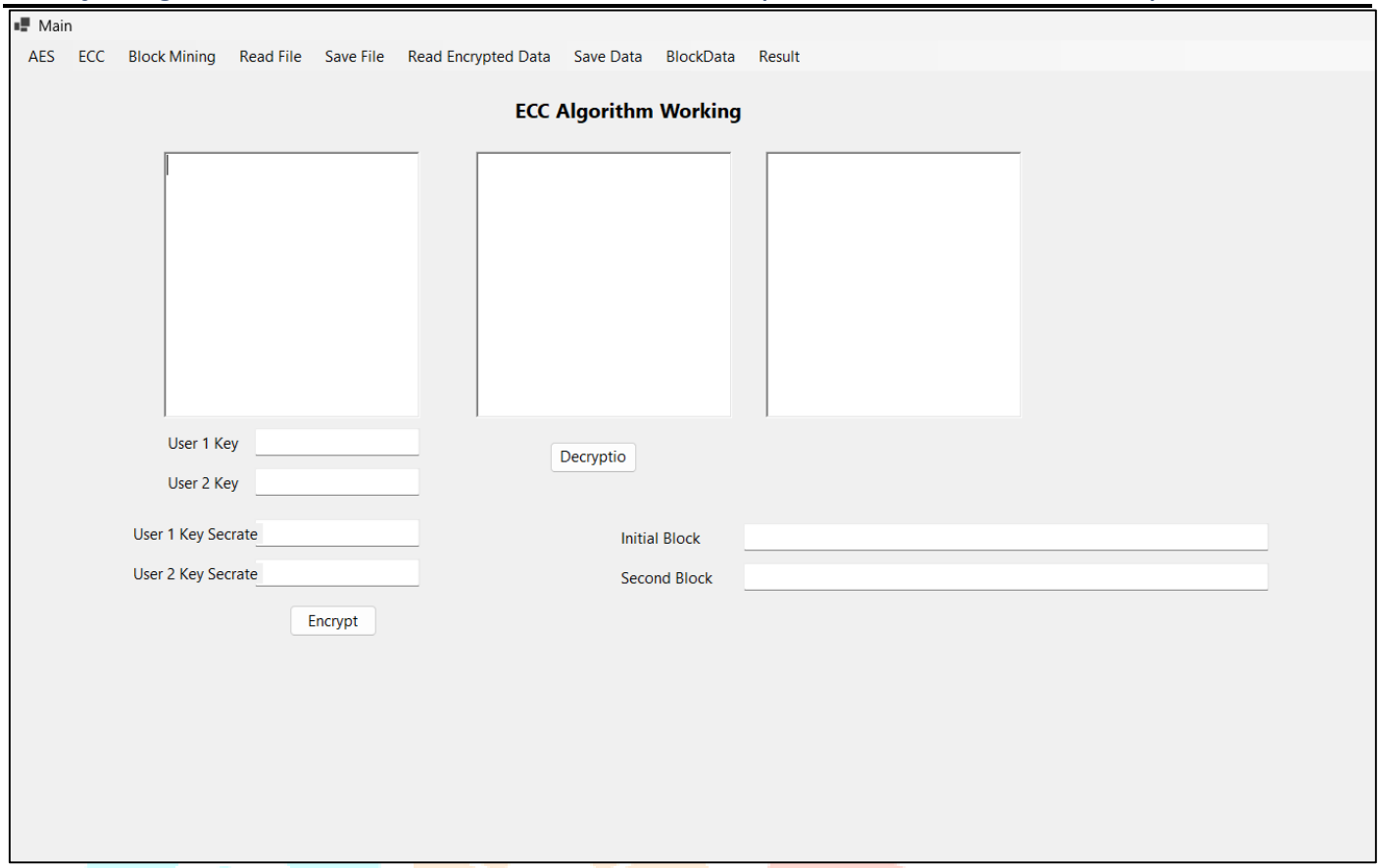

Figure 1: Internal Processing Model for DAES Block Handling and Encryption Workflow

VII. RESULTS

Module	Phase 1(Initial Version)	Phase 2(Final Implementation)
User Interface	Basic login/register	Full GUI with dashboard
Encryption	Standard AES	Distributed AES(DAES)
Key Management	Static Key	Blockchain-based
Key Sharing	Not available	ECC-based secure sharing
File Handling	Local encryption only	Encrypted upload/download
Blockchain Integration	Not used	Integrated for keys & logs
Audit Logs	Not available	Real-time logging
User Roles	Single role	Role-based access
Cloud Support	No	Yes (cloud-ready)
Testing	Basic	Complete with results

VIII. RESULTS

The proposed system provides a secure and efficient solution for protecting cloud data by integrating encryption, blockchain, and key exchange mechanisms. It offers a user-friendly interface, secure file upload/download, and reliable key management. The system is practical, easy to use, and suitable for real-time applications, making it a strong approach for modern cloud data security needs.

REFERENCES

[1] R. Anandkumar, K. Dinesh, A. J. Obaid, P. Malik, R. Sharma, A. Dumka, R. Singh, and S. Khatak, "Securing e-health application of cloud computing using hyperchaotic image encryption framework," Comput. Electr. Eng., vol. 100, May 2022, Art. no. 107860

[2] Z. Bashir, T. Rashid, and S. Zafar, "Hyperchaotic dynamical system based image encryption scheme with time-varying delays," Pacific Sci. Rev. A, Natural Sci. Eng., vol. 18, no. 3, pp. 254–260, Nov. 2016

[3] W. Y. Chang, H. Abu-Amara, and J. F. Sanford, Transforming Enterprise Cloud Services. Berlin, Germany: Springer, 2010.

[4] B.Alouffi,M.Hasnain,A.Alharbi,W.Alosaimi, H.Alyami,andM.Ayaz, "A systematic literature review on cloud computing security: Threats and mitigation strategies," IEEE Access, vol. 9, pp. 57792–57807, 2021.

[5] N. M. Sultana and K. Srinivas, "Survey on centric data protection method for cloud storage application," in Proc. Int. Conf. Comput. Intell. Comput. Appl. (ICCICA), Nov. 2021, pp. 1–8.

[6] F. Thabit, O. Can, S. Alhomdy, G. H. Al-Gaphari, and S. Jagtap, "A novel effective lightweight homomorphic cryptographic algorithm for data security in cloud computing," Int. J. Intell. Netw., vol. 3, pp. 16–30, 2022.

[7] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Integration of blockchain and cloud of things: Architecture, applications and challenges," IEEE Commun. Surveys Tuts., vol. 22, no. 4, pp. 2521–2549, 4th Quart., 2020.

[8] S. N. G.Gourisetti, ¨ U. Cali, K.-K.-R. Choo, E. Escobar, C. Gorog, A. Lee, C. Lima, M. Mylrea, M. Pasetti, F. Rahimi, R. Reddi, and A. S. Sani, "Standardization of the distributed ledger technology cybersecurity stack for power and energy applications," Sustain. Energy, Grids Netw., vol. 28, Dec. 2021, Art. no. 100553.

[9] F. Thabit, A. P. S. Alhomdy, A. H. A. Al-Ahdal, and P. D. S. Jagtap, "A new lightweight cryptographic algorithm for enhancing data security in cloud computing," Global Transitions Proc., vol. 2, no. 1, pp. 91–99, Jun. 2021.