IJIRAE:: International Journal of Innovative Research in Advanced Engineering ISSN: 2349-2163
Volume 12, Issue 11, November 2025 https://www.ijirae.com/archives
https://doi.org/10.26562/ijirae.2025.v1211.23

De-Centralized Block chain Based E-vault System: Leveraging Block chain, Decentralized Database (IPFS)

Lekha P
Department of Computer and Communication Engineering
Sri Sairam Institute of Technology, Chennai, India
lekha.cce@sairamit.edu.in
https://;orcid.org/0009-0004-5581-2978
Rishi Kowshic R , Murugappan P, Vasikara Vanan SS
Department of Computer and Communication Engineering
Sri Sairam Institute of Technology, Chennai, India
sit22co050@sairamtap.edu.in
sit22co054@sairamtap.edu.in
sit22co053@sairamtap.edu.in

Publication History
Manuscript Reference No: IJIRAE/RS/Vol.12/Issue11/NVAE10102
Research Article| Open Access | Double-Blind Peer-Reviewed | Article ID: IJIRAE/RS/Vol.12/Issue11/NVAE10102
Received:22,October 2025,Revised:28,October 2025, Accepted:31, October 2025, Published Online: 21, November 2025.
https://www.ijirae.com/volumes/Vol12/iss-11/23.NVAE10102.pdf
Citation:Lekha,Rishi,Murugappan,Vasikara(2025), De-Centralized Block chain Based E-vault System: Leveraging Block chain, Decentralized Database (IPFS) ,IJIRAE: International Journal of Innovative Research in Advanced Engineering, Volume 12, Issue 11 of 2025 pages 579-585
doi:>https://doi.org/10.26562/ijirae.2025.v1211.23
BibTeX Key: Lekha@2025De-Centralized
IJIRAE papers should be cited as IJIRAE (International Journal of Innovative Research in Advanced Engineering, AM Publications, India 2025, ISSN 2349-2163, https://doi.org/10.26562/ijirae.2025.v1211.23 The journal's official abbreviation is IJIRAE. Orcid: https://orcid.org/0009-0004-9398-7488

Abstract: In an era where digital transformation dominates and data privacy is paramount, existing cloud-based storage systems often fall short in providing trustless environments and tamper-proof data integrity. This paper justifies the need for a decentralized e-Vault storage application that addresses these limitations by leveraging block chain technology and the Inter Planetary File System (IPFS). The proposed solution integrates Pinata IPFS for decentralized file hosting, ensuring data immutability and accessibility. Ethereum-based Smart Contracts are employed to maintain a transparent, tamper-resistant ledger of records, while AES-GCM encryption safeguards data confidentiality. To enhance usability, modern web frame works are used for the user interface, while Meta Mask enables decentralized wallet management. Additionally, Fire base supports user authentication and metadata management, and Ganache facilitates local block chain testing. This architecture is deliberately chosen to overcome the vulnerabilities inherent in centralized systems, offering a secure, flexible alternative particularly well-suited for applications demanding high integrity and controlled access such as legal document storage, academic record management, and governmental archives. Each component of the system is strategically integrated to uphold data sovereignty, ensure auditability, and empower users with secure control over their digital assets.
Keywords: Decentralized storage, Pinata IPFS, block chain, AES-GCM encryption, content identifier (CID), immutable records, smart contracts, data integrity.

I. INTRODUCTION

In today's digital era, the amount of data generated, stored, and accessed online has reached unprecedented levels. From personal files and academic records to legal documents and government archives, sensitive data is increasingly being digitized and stored using cloud-based solutions. While cloud storage systems offered by major providers such as Google Drive, Dropbox, and Amazon S3 have revolutionized data accessibility and convenience, they are fundamentally built on centralized architectures. This centralization means that all files and metadata are stored on servers controlled by a single Organization or entity. As such, these systems inherently suffer from critical vulnerabilities including single points of failure, potential data breaches, unauthorized data access, and susceptibility to censorship or manipulation[1]. These issues are further magnified in institutional environments where data integrity, privacy, and auditability are not just important but mandatory. Government agencies, legal institutions, healthcare providers, and academic organizations often deal with data that must be protected by laws and regulations such as the General Data Protection Regulation (GDPR), HIPAA, and others. In these domains, data breaches or tampering can lead to severe legal consequences, loss of public trust, and compromise of individual rights.

Yet, traditional cloud services, despite robust service-level agreements and compliance certifications, still require trust in a third-party provider to handle all critical data operations. Users must assume that the provider will not access their data, will maintain security patches, and will remain operational assumptions that history has shown do not always hold true[2].

Problem Statement:

There is a critical need for a secure, transparent, and resilient data storage system that does not rely on centralized authorities. Institutions must be able to store, access, and audit sensitive records without the risk of external tampering ,internal abuse, or service unavailability. Centralized systems inherently lack transparency regarding how data is managed and who has access. Furthermore, even with strong perimeter defences, insider threats and human errors remain unresolved risks. Additionally, centralized architectures are vulnerable to targeted cyber attacks, Distributed Denial-of-Service (DDoS) as saults, and catastrophic failures such as data centre outages or shutdowns due to political or commercial factors. The need is for a system where trust is distributed, data access is tightly controlled and encrypted, and file records are immutable and independently verifiable without depending on a single provider[3].

To address these challenges, decentralized storage architectures have emerged as a transformative alternative. Built on peer-to-peer networks and secured by block chain technologies, these systems offer a way to distribute trust, remove reliance on a central authority, and enhance transparency. Decentralized systems store data across multiple nodes in a network, making them highly fault-tolerant and resistant to censorship. Blockchain, in particular, introduces immutability and verifiability through cryptographic consensus, enabling the secure recording of transactions and access histories. Meanwhile, technologies like the Inter Planetary File System (IPFS) allow files to be stored and retrieved based on their content hashes rather than location, ensuring data integrity and resistance to unauthorized modifications[4].

This paper presents a novel decentralized e-Vault storage system that combines the power of block chain and IPFS to create a tamper-proof, encrypted, and authenticated platform for institutional file management. The system addresses the weaknesses of traditional storage models by integrating AES- GCM encryption for client-side file confidentiality, Ethereum smart contracts for immutable recordkeeping, and Firebase Authentication for identity verification and access control[5].

By using Pinata IPFS as the storage layer, the system ensures that data is stored in a distributed and verifiable manner, while Meta Mask allows users to securely interact with the block chain through their digital wallets. This architecture ensures that only authorized users can upload or retrieve encrypted files, and every transaction is logged on the block chain, creating a complete and auditable history of document interactions[6].

In summary, this work contributes to the growing body of research in decentralized storage by proposing a secure and user-friendly solution specifically designed for institutional contexts. By combining modern cryptographic practices with block chain and decentralized networks, the proposed e-vault system re imagines how critical data can be stored, protected, and accessed in a truly trust less environment. The following sections detail the related works, architecture ,implementation, and evaluation of this system to demonstrate its practicality and advantages over traditional cloud-based solutions[7].

II. RELATED WORK

The concept of decentralized storage has steadily gained prominence over the last decade as the limitations of centralized cloud infrastructure become increasingly apparent. Traditional cloud services are often plagued by challenges such as vendor lock-in, data breaches, system downtimes, and opaque access control policies. In contrast, decentralized systems aim to eliminate these single points of failure by distributing data across a peer-to-peer network, often with the help of cryptographic verification, consensus mechanisms, and block chain integration[9]. Several notable projects have emerged in this space, each aiming to redefine how data is stored and managed. Filecoin, developed by Protocol Labs, is one of the most prominent decentralized storage platforms. It operates on an incentivized model where storage providers are rewarded with tokens for offering disk space and maintaining data availability. Filecoin is built atop IPFS and introduces mechanisms like proof-of-replication and proof-of-space time to verify that data is being stored reliably. Storj, another key player, focuses on encrypting and sharing files before distributing them across multiple storage nodes. Storj prioritizes data redundancy and privacy, with the aim of ensuring high availability and resistance to data loss. These platforms demonstrate the technical feasibility of decentralized storage at scale[10]. Beyond pure storage platforms, Hyperledger Fabric, developed under the Linux Foundation, explores decentralized ledger technologies (DLTs) tailored for enterprise use. Hyperledger supports permissioned blockchains, where only approved parties can access or modify data, making it suitable for corporate environments that require regulatory compliance and data confidentiality. However, such systems often require significant infrastructure and administrative overhead, which can limit their accessibility for smaller institutions or light weight applications [11]. Despite these advancements, real-world adoption of decentralized storage platforms still faces several barriers.Many existing solutions do not fully integrate essential components like client-side encryption, blockchain-based file traceability, or fine-grained user authentication. For instance, while Filecoin provides robust decentralized storage, it does not inherently enforce data encryption or user-specific access control these must be implemented externally. Similarly, platforms like Storj provide encryption, but they often lack native support for immutable blockchain audit trails or decentralized identity management [12]. To illustrate the practical limitations of current systems, consider a government office responsible for storing land registry records. If this office were to adopt a standard cloud storage provider, it would be vulnerable to data breaches or insider threats that could tamper with ownership histories leading to legal disputes or fraudulent claims. If the same office were to use Filecoin, they could decentralize storage, but they would still need a separate solution for encrypting the documents, managing user roles, and creating a verifiable transaction log.

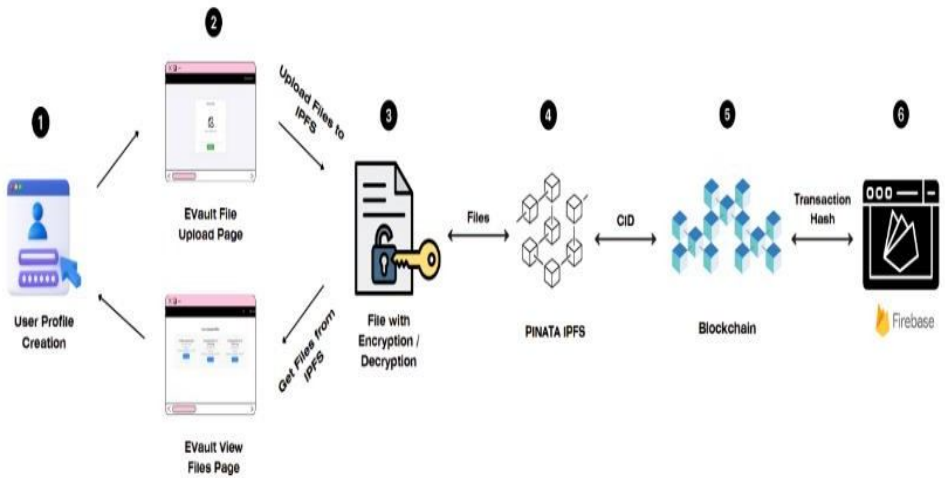
IJIRAE:: International Journal of Innovative Research in Advanced Engineering ISSN: 2349-2163
Volume 12, Issue 11, November 2025 https://www.ijirae.com/archives
https://doi.org/10.26562/ijirae.2025.v1211.23

The lack of an integrated platform that combines encryption, authentication, and blockchain traceability creates operational inefficiencies and potential security gaps [13]. Our proposed decentralized e-Vault system seeks to address these gaps by combining multiple technologies into a single, cohesive platform. Unlike many existing solutions, our system feature send-to-endencryption using AES-GCM, ensuring that documents are securely encrypted before ever leaving the client's device. Access control is enforced through Firebase Authentication, allowing only verified institutional users to upload or retrieve files[14]. The files themselves are uploaded to Pinatal PFS, a gateway for persistent and distributed content storage, while the Ethereum blockchain stores the content identifier (CID) in an immutable and auditable ledger. Smart contracts govern the interactions, adding transparency and tamper resistance to the entire process. In summary, while existing decentralized storage platforms like Filecoin and Storj have made impressive strides, they often operate as component solutions that require additional layers to ensure privacy, access control, and transparency. Our proposed system differentiates itself by providing a fully integrated approach tailored specifically for use cases that involve sensitive institutional data, such as academic records, legal files, or government documentation where both security and traceability are essential[15]

III. SYSTEM ARCHITECTURE

The proposed decentralize de-Vault storage system is designed with a modular architecture that integrates modern web development tools, block chain technology, secure encryption standards, and decentralized file storage. The architecture is carefully orchestrated to ensure scalability, security, and a smooth user experience. The system consists of the following key components:

- Frontend: The user interface is developed using React.js, a popular JavaScript library for building responsive and dynamic web applications. It is supported by Node.js and managed through npm (Node Package Manager), allowing for efficient dependency management and modular component-based development. The frontend facilitates seamless interaction with users, enabling them to upload, retrieve, and manage encrypted files with ease, while also interacting with the block chain backend.
- Block chain Layer: The core of the system's transparency and immutability lies in the Ethereum block chain. Smart contracts are written and deployed using the Remix IDE, a browser-based development environment tailored for Ethereum.
- These smart contracts handle the secure recording of content identifiers (CIDs) of uploaded files, ensuring that every file entry is cryptographically verifiable and tamper- proof. Local testing is conducted using Ganache, a personal block chain environment that allows developers to simulate transactions and verify contract behaviour in a controlled setup before deploying to a live network.
- Wallet Integration: To enable secure and decentralized interactions with the block chain, the system integrates Meta Mask, a browser extension and crypto wallet that allows users to manage their Ethereum identities and sign transactions. Meta Mask provides a trusted gateway for uploading or accessing files, with all actions authenticated and logged through block chain transactions, ensuring transparency and non- repudiation.
- Authentication: Secure access control is enforced through Firebase Authentication, which enables only verified institutional users to interact with the mentioned system. Firebase supports email/password login, OAuth providers, and custom identity providers, making it a flexible choice for managing official user access in academic, legal, or governmental environments . It also provides real-time session management, ensuring that unauthorized access attempts are immediately rejected.
- File Storage: The system utilizes Pinatal PFS, are liable and developer-friendly service built on top of the Inter Planetary File System (IPFS), to store files in a decentralized manner. IPFS breaks down and distributes files across a peer-to-peer network, with each file being assigned a unique content identifier (CID) based on its hash. Pinata ensures persistent file availability and integrates easily with IPFS gateways for seamless file retrieval.



- Encryption: Prior to uploading, all files are encrypted using the AES-GCM (Advanced Encryption Standard in Galois/Counter Mode) algorithm. AES-GCM offers both confidentiality and integrity by providing authenticated encryption, which means that any unauthorized tampering with the data can be detected. The encryption is performed on the client side, ensuring that even the storage and block chain layers never see plain text data. Files are decrypted in-browser only when accessed by authorized users.

This architecture ensures that data is encrypted before leaving the client, stored securely in a decentralized network, and recorded immutably on the block chain all while maintaining a user-friendly interface and secure authentication mechanism. Each component complements the others to offer a highly secure and trustworthy system for storing and managing sensitive institutional records

IV. PROPOSED SOLUTION

To tackle the issues of centralized storage vulnerabilities, lack of data transparency, and improper access control in sensitive data management systems, this project proposes a decentralized e-Vault storage application. The solution is architected using block chain technology, decentralized file systems, encryption protocols, and secure authentication mechanisms. Each technology component has been selected to maximize security, transparency, and data ownership while ensuring scalability and ease of access for verified users.

A. System Overview

The system leverages a hybrid architecture combining decentralized components and centralized verification layers. At the heart of the platform is a blockchain-enabled storage registry that links encrypted files stored on IPFS with the identity of authenticated users. Unlike traditional cloud-based solutions, this application does not store user data or files on centralized servers. Instead, it decentralizes the storage layer, ensures cryptographic security for file content, and immutably records references on the Ethereum blockchain.

B. Architectural Components

1) Front end Layer (React, Node.js,NPM):

The frontend is built using React, supported by Node.js and npm for dependency management and development tooling. It serves as the primary interface for users to interact with the system. The frontend is responsible for:

- Establishing connection with Meta Mask for Ethereum account integration.
- Enabling Firebase-based login and session management.
- Encrypting files locally using AES-GCM before uploading.
- Interacting with IPFS and smart contracts to store and retrieve encrypted file CIDs.

The UI is designed for minimalism, security awareness, and usability, providing clear access points to official users while presenting read-only capabilities to viewers (depending on roles).

2) Block chain Layer (Ethereum,Smart Contracts, Remix IDE, Ganache):

1. Smart contracts written in Solidity are deployed using Remix IDE and executed in a local block chain environment provided by Ganache during development and testing End-to-End Encryption: By encrypting files prior to upload, data privacy is preserved even on public IPFS nodes.
2. Immutability: The Ethereum smart contract ensures that once a CID is registered, it cannot be altered or deleted, preventing tampering.
3. Role-Based Access Control: Fire base is used to manage role- based access (official vs. viewer), ensuring only authorized uploads.
4. Wallet Binding: All file transactions are tied to the user's wallet address, preventing impersonation or spoofing.

These contracts manage:

- The registration of file uploads through storage of IPFS CIDs.
- Associating CIDs with Ethereum wallet addresses.
- Enforcing read-only access to stored CIDs without exposing file content.

No actual files are stored on-chain, which preserves Ethereum's light weight nature. Only file identifiers (CIDs) are recorded to ensure data integrity without incurring high gas costs or storage limitations.

3) Storage Layer (Pinatal PFS):

IPFS (Inter Planetary File System) is used for decentralized file storage, and Pinata provides a dedicated, reliable IPFS node interface for pinning encrypted files. Upon encryption, files are uploaded to Pinata IPFS which returns a CID a cryptographic hash uniquely representing the file's content. Pinning ensures that files remain persistently available on the IPFS network.

4) Encryption Layer (AES-GCM):

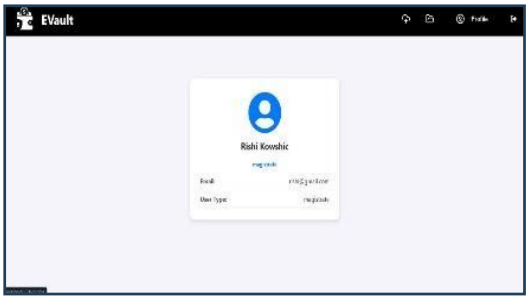
Before any file is uploaded to IPFS, it is encrypted client- side using the AES-GCM (Galois/Counter Mode) encryption algorithm. AES-GCM is chosen for its speed and ability to provide both confidentiality and integrity checks through authentication tags. The encryption process ensures that:

- Even if an attacker accesses the IPFS-stored file, it remains unreadable.
- Decryption can only occur with the proper key, which is kept off-chain and not stored in IPFS or Firebase.

5) Authentication Layer (Firebase):

Firebase Authentication is used to manage user access and verify identities. Users can log in via email/password or other methods. Fire base Authentication plays a critical role in access control:

- General users can view metadata and potentially download public files.
- Official users (administrators or authorized personnel) have elevated privileges allowing them to upload, encrypt, and store documents on the system.
- All file upload and user interaction logs are time stamped and optionally stored in Firebase for auditability.



C. Work flow Description

The overall flow of interactions in the system is as follows:

1. User Authentication: The user initiates a login via Fire base and connects their Ethereum wallet through Meta Mask.
2. File Encryption: An official user selects a file to upload. The file is encrypted using AES-GCM within the browser. A unique encryption key is generated per session.
3. Decentralized Storage: The encrypted file is uploaded to Pinata IPFS. Upon successful upload, a CID (Content Identifier) is returned.
4. Block chain Registration: The CID is submitted to the Ethereum smart contract. This creates an immutable mapping between the user's wallet address and the file identifier.
5. Accessing Files: When a user requests access to a file, the frontend retrieves the CID from the blockchain, fetches the encrypted file from IPFS, and prompts the user for the decryption key (managed off-chain).
6. Audit & Validation: Firebase stores interaction metadata for future evalidation, including upload timestamps, user IDs, and document tags (if applicable).

D. Security Considerations

Security is foundational to the system design. Specific mechanisms implemented include:

1. End-to-End Encryption: By encrypting files prior to upload, data privacy is preserved even on public IPFS nodes.
2. Immutability: The Ethereum smart contract ensures that once a CID is registered, it cannot be altered or deleted, preventing tampering.
3. Role-Based Access Control: Firebase is used to manage role- based access (official vs. viewer), ensuring only authorized uploads.
4. Wallet Binding: All file transactions are tied to the user's wallet address, preventing impersonation or spoofing.

E. Design Benefits

The system offers several key benefits over traditional centralized platforms:

1. Decentralization: Reduces single point of failure, increases resilience.
2. Transparency: Public smart contract ensures verifiability of stored data pointers.
3. Scalability: Uses light weight block chain transactions by storing only CIDs.
4. User-Controlled Security: Encryption and decryption are user- driven, reducing reliance on trusted third parties.
5.

V. IMPLEMENTATION

The decentralized e-Vault storage system was implemented using a combination of frontend, backend, and blockchain technologies:

- Front end: Built using React.js, the user interface allows for file uploads, retrieval, and interaction with MetaMask. Node.js and npm are used for project setup and dependency management.
- Encryption: Files are encrypted on the client-side using AES-GCM before upload, ensuring both confidentiality and integrity. Decryption also occurs locally upon retrieval.
- Smart Contracts: Ethereum smart contracts, developed and tested using Remix IDE and Ganache, are used to store and manage file content identifiers (CIDs) on the block chain securely.

- Meta Mask Integration: Meta Mask enables users to sign and verify block chain transactions securely, ensuring only authenticated actions are executed.
- Authentication: Firebase Authentication is used to verify users and control access. It supports secure login and tracks session data.
- Decentralized Storage: Encrypted files are uploaded to Pinata IPFS, which stores and manages the file across the IPFS network. Each file is assigned a unique CID, which is stored on the Ethereum blockchain.

This implementation ensures that data remains encrypted, verifiable ,and accessible only to authorized users, fulfilling the requirements for secure and decentralized institutional storage.

VI.RESULTS AN DDISCUSSION

The implementation and testing of the decentralized e-Vault storage system have yielded promising results that validate its design objectives interms of data confidentiality, integrity, and availability. During the testing phase, users were able to successfully upload various types of files through the web interface, with each file undergoing client-side encryption using the AES-GCM algorithm. This step ensured that all files were encrypted before transmission and storage, maintaining data confidentiality throughout the process. The encrypted files were uploaded to Pinata IPFS, which provided content-addressed storage by generating a unique content identifier (CID) for each file[1]. This CID, representing the file's hash, was stored on the Ethereum block chain through smart contract interactions facilitated by MetaMask. As a result, each file's existence and history became immutably recorded on the block chain, enabling transparent auditing and verification of every upload or retrieval action.

The encryption and decryption processes performed on the client side were validated across different sessions, demonstrating consistent and correct handling of file security. Files could be decrypted only by the same user who uploaded them, confirming that the key and initialization vector used during AES-GCM encryption were effectively managed. Moreover, access control was seamlessly enforced through Firebase Authentication. Only verified institutional users, as registered in the Firebase system, could interact with the application and perform upload or retrieval operations. This layer of user verification prevented unauthorized access, thereby maintaining system integrity and restricting interactions to trusted parties[2].

The React-based user interface contributed to a smooth and intuitive experience. Users could interact with the platform easily, with clearly defined prompts for uploading, decrypting, and verifying transaction records. Real-time feedback from MetaMask allowed users to confirm block chain transactions directly, adding transparency and trust to the system's workflow. Testing on the Ganache local blockchain environment demonstrated low and predictable gas costs, making the system efficient and practical for real-world use[3].

Even though local testing did not fully replicate public block chain performance, the results confirmed the feasibility and cost-effectiveness of the approach. Over all, the integration of these various technologies encryption, IPFS, blockchain, and authentication resulted in a robust and secure platform, well-suited for handling sensitive institutional data such as legal documents, academic records, or government files[4].

CONCLUSION

This paper presents a novel approach to secure file storage by designing and implementing a decentralized e-Vault system that integrates block chain, Pinata IPFS, client-side encryption, and user authentication technologies into a unified platform. The system addresses the shortcomings of traditional centralized storage solutions, which often suffer from single points of failure, lack of transparency, and increased risk of data breaches. By utilizing Ethereum smart contracts to record immutable file metadata, the solution ensures that each document's upload or retrievalis verifiable and tamper-proof. The use of Pinatal PFS as the file storage backbone adds decentralized resilience and fault tolerance, ensuring that files remain accessible without depending on a central server. Furthermore, the application of AES-GCM encryption ensures end-to-end confidentiality and data integrity, with all encryption and decryption operations confined to the client side eliminating risks associated with plaintext data exposure on the server. A critical differentiator of this system is the integration of Firebase Authentication, which enforces strict access control by permitting only verified institutional users to interact with the storage platform. This feature ensures that sensitive data, such a s academic transcripts, court files, or official contracts, are only accessible to authorized personnel. The system's frontend, built with React.js, provides an accessible and user-friendly interface that bridges the gap between complex decentralized technologies and real-world usability. Users can easily manage their files and monitor block chain interactions without needing deep technical knowledge.

In conclusion, the decentralized e-Vault storage system not only demonstrates a technically sound implementation but also presents available and secure alternative for institutions seeking to enhance data protection and transparency. Future enhancements may include deploying the system to a public Ethereum testnet or main net, benchmarking performance across different network loads, and introducing advanced features such as zero-knowledge proofs for privacy-preserving audits. With growing concerns about digital security and data sovereignty, the proposed solution holds significant potential for real-world adoption across legal, academic, and governmental sectors.

REFERENCES

1. Akbarfam, Asma Jodeiri, M. Heidaripour, H. Maleki, G. Dorai, and G. Agrawal, “ForensiBlock: A Provenance-Driven Blockchain Framework for Data Forensics and Auditability,” arXiv.org, 2023. https://arxiv.org/abs/2308.03927.

2. S.Lamichhane and P.Herbke, "Verifiable Decentralized IPFS Cluster: Unlocking Trust worthy Data Permanency for Off-Chain Storage," arXiv.org, 2024. https://arxiv.org/abs/2408.07023.
3. M.R.Haque, S.I.Munna, S.Ahmed, M.T.Islam, Hassan, and Rahman,"An Integrated Blockchain and IPFS Solution for
4. Secure and Efficient Source Code Repository Hosting using Middleman Approach," arXiv.org, 2024.
5. https://arxiv.org/abs/2409.14530.
6. Aditya Kumar Sharma and Brijesh Kumar Chaurasia, "Blockchain-Based NFT for Evidence System," Algorithms for intelligent systems, pp. 441–451, Oct. 2023, doi: https://doi.org/10.1007/978-981-99-2229-1_37.
7. "Integrating Public Reported Evidence Collection, Public Court Records Archive And Realizing Secure And Decentralized Case Document Management Using IPFS And Hyperledger Fabric Blockchain: An Implementation Study-Peer-reviewed Journal," Peer-reviewed Journal, 2018.
8. M.Ahmed, A.R.Pranta, Mst. F. A. Koly, F. Taher, and M. A. Khan, "Using IPFS and Hyperledger on Private Blockchain to Secure the Criminal Record System," European Journal of Information Technologies and Computer Science, vol. 3, no. pp.1–6,Jan2023,doi:https://doi.org/10.24018/compute.2023.3.1.81.
9. S.Liu and Q.Zheng, "A study of a blockchain-based judicial evidence preservation scheme," Blockchain: Research and Applications, vol. 5, no. 2, p. 100192, Jun. 2024, doi: https://doi.org/10.1016/j.bcra.2024.100192.
10. H.Chen, Y.Lu, and Y.Cheng, "File Insurer: A Scalable and Reliable Protocol for Decentralized File Storage in Block chain," arXiv.org, 2022. https://arxiv.org/abs/2207.11657
11. J.Hao, Y.Sun, and H.Luo,"A safe and efficient storage scheme based on block chain and IPFs for agricultural products
12. tracking," 電腦學刊, vol.29, no. 6, pp. 158–167, Jan. 2018, doi: https://doi.org/10.3966/199115992018122906015.
13. Q. Zhang and Z. Zhao, "Distributed storage scheme for encryption speech data based on block chain and IPFS,"The Journal of Supercomputing, Jul. 2022, doi: https://doi.org/10.1007/s11227-022-04702-1.
14. TahirAlyasetal.,"Multi block chain architecture for judicial case management using smart contracts,"Scientific Reports,vol. 15, no. 1, Mar. 2025, doi: https://doi.org/10.1038/s41598-025-92842-8.
15. M.Antony Saviour and D.Samiappan,"IPFS based file storage access control and authentication model for secure data transfer using block chain technique," Concurrency and Computation: Practice and Experience, Nov. 2022, doi: https://doi.org/10.1002/cpe.7485.
16. S. Athanere and R.Thakur, "Block chain based hierarchical semi-H. Chen, Y. Lu, and Y. Cheng, "FileInsurer: A Scalable and Reliable Protocol for Decentralized File Storage in Blockchain,"arXiv.org,2022.https://arxiv.org/abs/2207.11657
17. H.S.Huang, T.S. Chang, and J.Y.Wu, "A Secure File Sharing System Based on IPFS and Block chain,"arXiv.org, May 02,2022. https://arxiv.org/abs/2205.01728
18. M.Dave and Rajkumar Banoth,"Block chain-based, Decentralized Evidence Archive System using IPFS," 2022 International Conference on Sustainable Computing and Data Communication Systems(ICSCDS),pp.1166–1170,Apr.2022, doi: https://doi.org/10.1109/ICSCDS53736.2022.9760983.
19. V.Kanchana Lakshmi and Dr.Karamsettyshouryadhar, "DESIGNOFABLOCKCHAIN-BASEDSECUREDATA TRANSFER MODEL FOR IPFS FILE STORAGE WITH ACCESS CONTROL," International Journal of Engineering Research and Science &Technology,vol.20,no.4,pp.135– 143, 2024, Accessed: Apr. 09, 2025. [Online]. Available: https://ijerst.org/index.php/ijerst/article/view/456
20. C.Rahalkar and D.Gujar,"Content Addressed P2P File System for the Web with Blockchain-Based Meta-Data Integrity," arXiv.org, 2019. https://arxiv.org/abs/1912.10298

21.]Y.M.GajmalandR.Udayakumar,"Blockchain-Based Access Control and Data Sharing Mechanism in Cloud Decentralized Storage System," Journal of Web Engineering, Aug. 2021, doi: https://doi.org/10.13052/jwe1540-9589.2054
22. J. Swati and P. Nitin, "Securing Decentralized Storage in Block chain: A Hybrid Cryptographic Framework, "Cybernetics And InformationTechnologies,vol.24,no.2,pp.16–31,Jun. 2024, doi: https://doi.org/10.2478/cait-2024-0013.