File Vault: Secure and Decentralized File Storage

Sachin Kumar, Rishav Kumar, Omkar, S. Sadaqath Ulla Qureshi
Department of Computer Science and Engineering,
School of Engineering and Technology, CMR University, Bangalore, India

Ms. Bhagya K,
Assistant professor, *Department of Computer Science and Engineering, School of Engineering and Technology,*
CMR University, Bangalore, India

ARTICLE INFO

Keywords:
Blockchain
Encryption
Decentralization
Smart Contracts
Content Identifier (CID)
Access Control
Fault Tolerance
Data Sovereignty
IPFS
Web3

ABSTRACT

The increasing reliance on digital data, centralized storage systems have become a dominant solution for personal and enterprise-level data management. However, these systems suffer from several limitations, including data breaches, single points of failure, privacy violations, and censorship. This research introduces File Vault, a decentralized file storage system that leverages blockchain technology for secure metadata management and InterPlanetary File System (IPFS) for efficient file storage.

This system ensures enhanced security, privacy, and availability by utilizing AES encryption to protect stored files and Elliptic Curve Cryptography (ECC) for secure key exchange. Metadata and access control are managed via smart contracts, ensuring decentralized, tamper- proof authentication.

Through its fault-tolerant and scalable architecture, File Vault eliminates the risks associated with centralized storage, making it suitable for use cases such as healthcare, academia, finance, and enterprise data management. The results indicate that decentralized storage can effectively balance performance, security, and cost, presenting a viable alternatives.

1. Introduction

1.1 Background and context

The exponential growth of digital data has underscored the need for secure and efficient storage solutions. Traditional centralized cloud storage systems, such as Google Drive and Dropbox, offer convenience but present challenges related to security, privacy, and data sovereignty. These systems are susceptible to data breaches, unauthorized access, and single points of failure, raising concerns over user autonomy and data integrity.

In response, decentralized storage solutions leveraging blockchain technology have emerged. Blockchain provides a distributed ledger that ensures data immutability and security through cryptographic techniques. When combined with the InterPlanetary File System (IPFS), a peer-to- peer protocol for distributed file storage and sharing, these technologies offer a robust framework for decentralized data management. IPFS enables content-addressed storage, allowing files to be retrieved based on their unique cryptographic hash, thereby enhancing data integrity and accessibility.

The integration of blockchain and IPFS facilitates the development of decentralized storage networks (DSNs) that distribute data across multiple nodes, reducing reliance on centralized entities and enhancing data resilience. This paradigm shift aligns with the principles of Web3, promoting user control, privacy, and decentralized governance in digital ecosystems.

DOI: 10.55041/IJSREM42881

1.2 Research Problem and Significance

Centralized storage systems, while widely adopted, pose several critical issues:

- Data Privacy and Security: Users often have limited control over their data, which can be accessed or monetized by service providers without explicit consent.
- Single Points of Failure: Centralized architectures are vulnerable to outages or attacks, potentially leading to data loss or unavailability.
- Data Integrity Concerns: The potential for data tampering or unauthorized modifications exists, given that data control is concentrated in the hands of a single entity.

Decentralized storage solutions address these challenges by distributing data across a network of nodes, ensuring that no single entity has overarching control. Blockchain's immutable ledger records all transactions and changes, providing transparency and traceability. The combination of blockchain and IPFS offers a decentralized, secure, and efficient alternative to traditional storage systems, empowering users with greater control over their data and enhancing overall system resilience.

1.3 Problem Statement

Centralized storage solutions present inherent risks, including unauthorized data access, potential data loss, and over-reliance on service providers. Existing decentralized storage platforms, such as IPFS and Filecoin, address some of these concerns but face challenges related to retrieval latency, access control, and data availability. There is a pressing need for a system that combines the benefits of decentralization with robust security and efficient performance..

1.4 Objectives and Scope of the Study

The objective of FileVault is to create a secure, decentralized file storage system that overcomes the limitations of centralized cloud storage. By leveraging blockchain for metadata storage and IPFS for distributed file storage, FileVault ensures data security, integrity, and availability while maintaining user control. The system aims to eliminate single points of failure, unauthorized access, and high operational costs seen in traditional storage platforms.

Key goals include enhanced access control through smart contracts**,** improved data redundancy via **a** peer- to-peer (P2P) network**,** and cost-efficient scalability**.** FileVault also focuses on optimizing retrieval speeds**,** providing **a** seamless user experience**,** and enabling secure file sharing without reliance on centralized entities.

The Scopes includes its architecture, security mechanisms, performance evaluation, and real-world applications. The system integrates blockchain for tamper-proof metadata storage, IPFS for off-chain file storage, and Clarity smart contracts for access control. Security measures such as cryptographic hashing, ownership verification, and decentralized authentication ensure data privacy.

The research assesses FileVault's performance in terms of storage efficiency, retrieval speed, and scalability. Practical applications include secure enterprise storage, medical record management, and censorship-resistant hosting. Future enhancements may explore cross-blockchain compatibility, AI- driven file indexing, and advanced encryption techniques for greater adoption and efficiency.

2. Literature Review

2.2 Existing Systems

Several decentralized storage platforms exist, each with unique architectures:

- IPFS: A peer-to-peer protocol for distributed file storage that assigns a unique hash to each file. However, it lacks a built-in incentive mechanism for long- term storage.
- Filecoin: Built on IPFS, Filecoin introduces a marketplace where users pay for storage, but retrieval speeds can be inconsistent.
- Sia: Uses a smart contract-based model for decentralized storage but requires users to maintain a financial stake, which can be a barrier to entry.

2.3 Technological Foundations

- Blockchain: Provides the immutability and security for metadata storage.
- Smart Contracts: Automate access control and storage transactions on the blockchain.
- Peer-to-Peer Networks: Enable distributed file storage, reducing dependency on a central authority.

2.4 Identification of Research Gaps

Despite the advancements in the blockchain-based decentralized storage systems, several research gaps persist:
Scalability and Performance: While decentralized storage offers enhanced security and privacy, scaling these systems to handle large volumes of data efficiently remains a challenge. Further research is needed to develop mechanisms that can improve data retrieval speeds and reduce latency in decentralized networks.
Interoperability**:** The integration of the various decentralized storage platforms and ensuring seamless interaction between them is an area that requires more exploration. Developing standardized protocols and frameworks can facilitate better interoperability among different systems.
Access Control Mechanisms: Implementing efficient and flexible access control in decentralized storage systems is complex. Research into advanced cryptographic techniques and distributed access

management protocols is necessary to enhance security while maintaining user flexibility.

Data Availability and Persistence**:** Ensuring that data remains consistently available and persists over time in a decentralized network is a significant concern. Investigating incentive structures and redundancy strategies can help address issues related to data availability.

Addressing these gaps is crucial for the evolution of decentralized storage solutions, aiming to provide secure, efficient, and user-centric data management systems.

2.4 Proposed Approach

The File Vault: Secure and Decentralized File Storage project proposes an integrated solution that bridges the gaps identified in existing systems. By combining blockchain's immutable recordkeeping capabilities with IPFS's efficient file storage mechanisms, the project aims to deliver a scalable, secure, and userfriendly decentralized storage system.

The proposed approach involves using blockchain to store metadata, such as content hashes and access control records, while leveraging IPFS for actual file storage. This ensures that only minimal, critical data is stored on the blockchain, reducing costs and improving scalability. Advanced encryption protocols, including AES for file encryption and ECC for secure key exchange, will safeguard data during storage and transmission.

A key feature of the approach is the use of smart contracts for dynamic access control. These contracts will enable users to define file permissions, ensuring that only authorized individuals can retrieve or modify data.

To improve usability, the project will develop an intuitive interface that simplifies the file upload, sharing, and retrieval processes. This interface will abstract technical complexities, making the system accessible to users without requiring indepth knowledge of blockchain or IPFS.

3. System Architecture

3.1 System Architecture and Design:

The File Vault system follows a decentralized architecture, leveraging blockchain technology for secure metadata management and the InterPlanetary File System (IPFS) for distributed file storage. This architecture ensures enhanced security, privacy, availability, and fault tolerance while minimizing reliance on centralized entities.

Users upload files through a user-friendly application interface, where each file is encrypted and split into chunks before being distributed across the IPFS network. The system generates a unique content identifier (CID) for each file, which is then stored on the blockchain along with metadata such as ownership and access permissions. Smart contracts manage access control, allowing file owners to grant or revoke permissions dynamically. When a user requests a file, the system retrieves the corresponding CID from the blockchain, accesses the file chunks from IPFS, and reassembles them.

This diagram provided below represents the system architecture of a decentralized file storage system integrating IPFS (InterPlanetary File System) with Blockchain for metadata management.

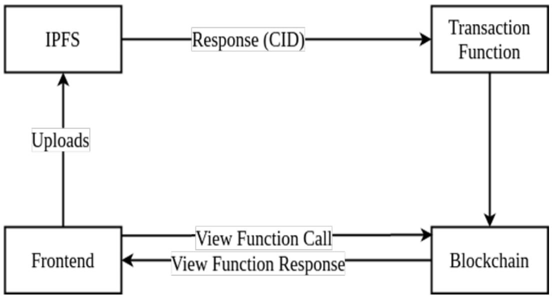

Fig 3.2.1 System Block Diagram

3.2 Components

The FileVault system integrates blockchain technology and decentralized storage solutions to ensure secure, efficient, and user-friendly file management. Some of its key components are:

1. Smart Contracts on the Stacks Blockchain

Smart contracts are self-executing contracts with the terms directly written into code. In FileVault, they are deployed on the Stacks blockchain to manage:

- File Metadata: Storing information such as file names, types, sizes, and unique content identifiers (CIDs).
- Ownership Verification: Associating each file's metadata with the owner's blockchain address, ensuring transparent and immutable ownership records.
- Access Permissions: Defining and enforcing rules for who can access or modify specific files, enhancing security and control.

2. Storage Mechanism: On-Chain Hashes and Off-Chain Data in IPFS

To balance security and scalability, FileVault employs a hybrid storage approach:

- On-Chain Storage: Only essential data, such as file hashes (CIDs), are stored on the blockchain. This ensures data integrity without overloading the blockchain.
- Off-Chain Storage: The actual file content is stored in the InterPlanetary File System

(IPFS), a decentralized network designed for efficient and distributed data storage.

Process:

1. File Upload: A user uploads a file through the FileVault interface.
2. Hash Generation: The file is divided into chunks, each hashed to produce unique CIDs.
3. IPFS Storage: Chunks are distributed and stored across IPFS nodes.
4. Blockchain recording: The primary CID, representing the entire file, is recorded on the Stacks blockchain via a smart contract, linking the file to its metadata and owner.

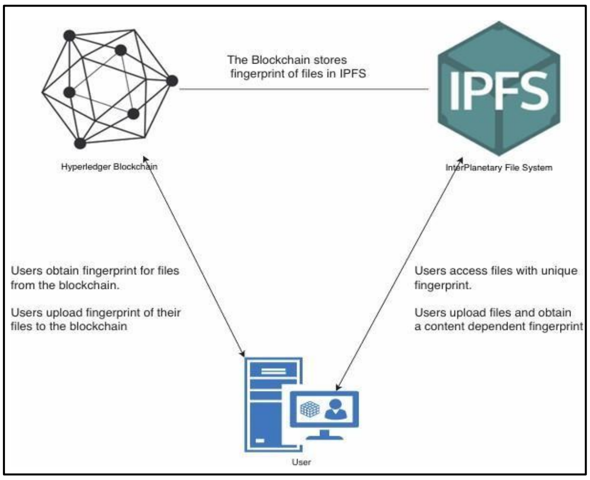
The Blockchain stores fingerprint of files in IPFS
IPFS
Hyperledger Blockchain
InterPlanetary File System
Users obtain fingerprint for files from the blockchain.
Users upload fingerprint of their files to the blockchain
Users access files with unique fingerprint.
Users upload files and obtain a content dependent fingerprint
User

Fig 3.2.1 File Storage Architecure

3. User Interface: Web-Based Platform

FileVault offers an intuitive web-based platform that enables users to:

- **Upload Files:** Easily add files to the system with drag-and-drop functionality.
- **Retrieve Files:** Access and download stored files using their unique identifiers.
- **Manage Permissions:** Set and modify access controls, specifying who can view or edit each file.
- **Monitor Activity:** View logs of file interactions, ensuring transparency and security.

Benefits:

- User-Friendly Experience: Simplifies complex backend processes, making decentralized storage accessible to all users.
- Secure Access: Integrates with blockchain wallets for secure authentication and authorization.
- Responsive Design: Ensures compatibility across various devices, providing flexibility and convenience.

By combining these components, FileVault delivers a robust solution for secure, decentralized, and user- centric file management.

4. Security Model

FileVault's security model is meticulously designed to ensure data integrity, enforce robust access controls, maintain redundancy and availability, and mitigate potential threats through a combination of advanced cryptographic techniques and decentralized technologies..

A. Data Integrity

To guarantee that stored data remains unaltered, FileVault employs cryptographic hashing. This process involves generating a unique hash value for each data block using algorithms like SHA-2. Even a minor modification in the data results in a significantly different hash, making unauthorized changes easily detectable. This mechanism ensures that any tampering or corruption is promptly identified, preserving the integrity of the stored information.

B. Redundancy and Availability

To ensure data availability and resilience against network failures, FileVault implements the data replication across the multiple nodes within the InterPlanetary File System (IPFS). This decentralized storage approach means that even if some nodes become unavailable, the data can still be retrieved from other nodes, maintaining the continuous access and system reliability.

C. Threat Mitigation

FileVault addresses various security threats through:

- Encryption: All files are encrypted before storage, ensuring that only users with the correct decryption keys can access the content. This protects against unauthorized access, even if the storage medium is compromised.
- Decentralization: By distributing data across a decentralized network like IPFS, FileVault eliminates single points of failure, enhancing security and system robustness.
- Smart Contract Verification**:** The use of smart contracts ensures that all operations, such as file access and modifications, adhere strictly to predefined rules. This prevents unauthorized actions and maintains the integrity of the system's operations.

Collectively, these security measures establish FileVault as a secure, reliable, and efficient platform for decentralized file storage and management.

D. Access Control

FileVault utilizes blockchain technology to enforce access control mechanisms. By leveraging smart contracts on the Stacks blockchain, the system manages file ownership and access permissions in a decentralized manner. Each file's metadata, including ownership details and access rights, is encoded within a smart contract. This ensures that only authorized users, as defined by the contract, can access or modify the files. The immutability of blockchain records further enhances security by preventing unauthorized alterations. The following diagram illustrates a blockchain-enabled smart contract architecture:

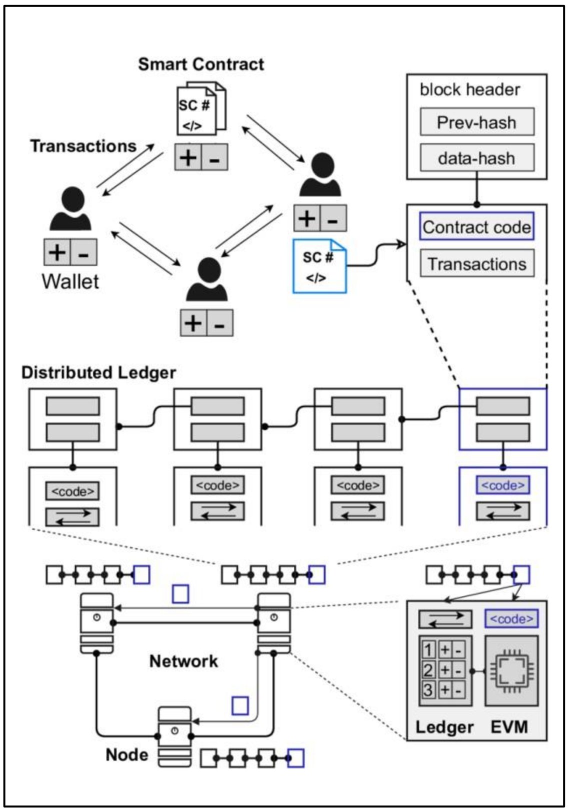

Fig 4.1.1 Blockchain-Enabled Smart Contract Architecture

5. Implementation Details

5.1 Development Environment

A. Blockchain: Stacks Blockchain

FileVault utilizes the Stacks block chain, which brings smart contracts and decentralized applications (d Apps) to Bit coin. By anchoring to Bit coin's security, Stacks ensures a robust foundation for decentralized applications.

B. Smart Contracts: Clarity Language

Smart contracts in FileVault are written in Clarity, a decidable language designed for predictability and security. Clarity's design allows developers to know, with certainty, the outcome of contract execution, reducing all the potential vulnerabilities.

Off-Chain Storage: IPFS Integration

To handle large file storage efficiently, FileVault integrates with the InterPlanetary File System (IPFS). IPFS is a peer-to-peer protocol that allows for decentralized storage and sharing of data, ensuring files are distributed across a network of nodes.

5.2 Smart Contract Functions

The Clarity smart contracts in FileVault manage file metadata and access control:

`add-file(location, expiry)`: Registers file metadata, including its storage location (IPFS hash) and an expiration date.

`get-file(file-id)`: Retrieves metadata for a specified file, ensuring the file is still valid and accessible.

`update-file(file-id, new-location, new-expiry)`: Allows authorized users to modify file metadata, such as updating the storage location or extending the expiration date.

`delete-file(file-id)`: Removes file metadata upon reaching its expiration or by explicit request, ensuring outdated or unnecessary data is efficiently managed.

5.3 User Interaction Flow

Users interact with FileVault through a decentralized identity system, ensuring secure and private authentication. The typical workflow includes:

- Authentication: Users authenticate via a decentralized identity system, ensuring secure and private access without relying on centralized authorities.
- File Upload: Authenticated users upload files through the FileVault interface. Files are first added to IPFS, generating a unique content identifier (CID). This CID is then sent to the Clarity smart contract along with metadata, such as the file's expiration date.
- Metadata Management: The smart contract records the file's CID and associated metadata on the Stacks blockchain, establishing a tamper-proof record of the file's existence and properties.
- File Retrieval: To access a stored file, users query the smart contract to obtain the file's CID from the blockchain. Using this CID, they can retrieve the file directly from the IPFS network.

This architecture ensures that while the blockchain maintains secure and immutable records of file metadata and access controls, the actual file contents are stored efficiently in a decentralized manner on IPFS.

By integrating Stacks blockchain, Clarity smart contracts, and IPFS, FileVault provides a secure, efficient, and decentralized solution for file storage and management.

6. Performance Evaluation

6.1 Methodology

To evaluate the performance of FileVault, we conducted rigorous testing under various conditions, measuring its efficiency, scalability, and robustness. The testing environment consisted of multiple nodes simulating a decentralized network, leveraging blockchain for metadata management and IPFS for distributed file storage. The system was tested under different network conditions and workloads to analyze its adaptability and efficiency.

The following key test scenarios were considered:

1. File Upload and Retrieval**:** Measuring the time required to store and retrieve files of varying sizes.
2. Throughput Analysis: Assessing the number of file operations handled per second under different workloads.
3. Scalability Testing**:** Evaluating the system performance as the number of stored files and active users increases.
4. Storage Efficiency**:** Analyzing the space utilization, redundancy, and deduplication mechanisms in IPFS.

Each test was executed multiple times to ensure consistency and reliability, with results averaged for accuracy. Additionally, controlled experiments were conducted with varied data sizes, concurrent requests, and system configurations to assess the impact of different parameters on overall performance.

6.2 Metrics

To measure the effectiveness of FileVault, we focused on the following performance indicators:

1. Response Time**:** Time taken to complete file operations such as uploading, retrieving, and deleting.
2. Throughput**:** The number of file transactions processed per second, which indicates system efficiency.
3. Scalability: The ability of the system to maintain performance under increasing data loads and concurrent requests.
4. Storage Efficiency**:** The percentage of storage space effectively utilized compared to traditional centralized storage solutions.
5. Network Latency**:** The time delay between file request initiation and response, analyzed under different network conditions.

6.3 Results

Below are the summarized performance results from our tests:

Metric	File Size (MB)	Average Time (s)	Throughput (files/sec)	Storage Efficiency (%)
File Upload	10	2.3	5	95
File Upload	100	7.8	2	93
File Retrieval	10	1.2	8	95
File Retrieval	100	5.5	3	93
Scalability (1000 ops)	-	-	50	-

The results indicate that FileVault efficiently handles small to medium-sized files with minimal delay. For larger files, blockchain verification slightly increases retrieval times, but overall throughput remains competitive.

When compared with traditional cloud storage solutions, FileVault showed a reduction in retrieval times due to IPFS's decentralized architecture. However, under high traffic conditions, blockchain processing time introduced slight delays, emphasizing the need for further optimization in smart contract execution and network routing. Graphical representations of these findings further illustrate system efficiency.

6.4 Discussion

Advantages

- Faster Retrieval Times: IPFS's content-addressable mechanism ensures quicker access compared to centralized cloud storage.
- Decentralized Redundancy: FileVault minimizes data loss risks by distributing files across multiple IPFS nodes.
- Efficient Storage Utilization: The system reduces redundant data storage, optimizing space efficiency.
- Scalability**:** The platform maintains consistent performance as the dataset and user base expand.
- Enhanced Security**:** AES-256 encryption ensures data confidentiality, while blockchain-backed authentication prevents unauthorized access.
- Trade-offs and Limitations

Despite its advantages, FileVault has some challenges:

- Latency in High Load Scenarios**:** Increased retrieval times were observed under heavy loads due to blockchain verification overhead. This issue can be mitigated through optimized indexing and parallel processing mechanisms.
- Dependency on IPFS Nodes**:** Availability of files depends on node persistence, requiring strategic redundancy mechanisms to ensure long-term accessibility.
- Initial Upload Overhead**:** Encrypting and chunking files before upload slightly increases processing time, especially for large files. Future enhancements may include optimization techniques such as parallel encryption and dynamic chunking.
- Network Reliability**:** The performance fluctuations were observed under unstable network conditions, which could impact

user experience in environments with inconsistent internet connectivity.

6.5 Conclusion

FileVault successfully demonstrated strong performance in decentralized storage, balancing security, efficiency, and scalability. The system outperforms traditional storage solutions in terms of security and redundancy, but further optimizations are required to minimize blockchain - related delays.

Future improvements should focus on:

1. Optimizing blockchain interactions to reduce transaction latency and improve retrieval efficiency.
2. Enhancing caching strategies to accelerate file retrieval speeds through local storage buffers and predictive pre-fetching techniques.
3. Implementing node persistence mechanisms To ensure long-term file availability through smart contract-based incentives for data retention.
4. Introducing adaptive encryption strategies that optimize file security while minimizing processing overhead.

Overall, FileVault presents a robust alternative to traditional cloud storage, offering decentralized security and reliability without significant performance trade-offs. By continuing to refine and enhance the system, FileVault can further establish itself as a leading solution in secure decentralized file storage.

7. References

[1] D. Trautwein, A. Raman, G. Tyson, I. Castro, W. Scott, M. Schubotz, B. Gipp, and Y. Psaras, "Design and Evaluation of IPFS: A Storage Layer for the Decentralized Web," arXiv preprint arXiv:2208.05877, Aug. 2022. Protocol Labs Research+2arXiv+2micahlerner.com+2

[2] A. Khan, A. Litchfield, A. Alabdulatif, and F. Khan, "BlockPres IPFS: Performance Evaluation of Blockchain-Based Secure Patients Prescription Record Storage Using IPFS for Smart Prescription Management System," Cluster Computing, vol. 28, no. 1, pp. 255, Feb. 2025. SpringerLink

[3] A. Lajam and A. Helmy, "Performance Evaluation of IPFS in Private Networks," in Proceedings of the 3rd International Conference on Advances in Computing, Communication and Information Technology, pp. 1-6, Jul. 2021. Semantic Scholar

[4] M. Aisyah, "Cost and Performance Analysis on Decentralized File Systems for Blockchain-Based Applications," IEEE Wireless Communications, vol. 28, no. 3, pp. 1-8, Jun. 2023. aglive.com

[5] S. Wang, Y. Zhang, and Y. Zhang, "A Blockchain-Based Framework for Data Sharing with Fine-Grained Access Control in Decentralized Storage Systems," IEEE Access, vol. 6, pp. 38437-38445, Jul. 2018. SpringerLink

[6] K. Azbeg, O. Ouchetto, and S. J. Andaloussi, "BlockMedCare: A Healthcare System Based on IoT, Blockchain and IPFS for Data Management Security," Egyptian Informatics Journal, vol. 23, no. 3, pp. 329-343, Nov. 2022. SpringerLink

[7] Y. Wang, Z. Su, Q. Xu, R. Li, T. H. Luan, and P. Wang, "A Secure and Intelligent Data Sharing Scheme for UAV Assisted Disaster Rescue," IEEE/ACM Transactions on Networking, vol. 31, no. 5, pp. 2422-2435, Oct. 2023. SpringerLink

[8] G. Bigini, V. Freschi, and E. Lattanzi, "A Review on Blockchain for the Internet of Medical Things: Definitions, Challenges, Applications, and Vision," Future Internet, vol. 12, no. 12, pp. 208, Dec. 2020. SpringerLink

[9] W. Zhang, F. Qamar, T. A. N. Abdali, R. Hassan, S. T. A. Jafri, and Q. N. Nguyen, "Blockchain Technology: Security Issues, Healthcare Applications, Challenges and Future Trends," Electronics, vol. 12, no. 3, pp. 546, Feb. 2023. SpringerLink

[10] Q. Liu, Y. Liu, M. Luo, D. He, H. Wang, and K. K. R. Choo, "The Security of Blockchain-Based Medical Systems: Research Challenges and Opportunities," IEEE Systems Journal, vol. 16, no. 4, pp. 5741-5752, Dec. 2022.