

Securing Electronic Medical Records Using Blockchain and Cryptography: A Decentralized Approach

Deepa Rani Dr. ^{1,✉}, Manikant Prasad ², Rajeev Kumar ³

1. Computer Science and Engineering, National Institute of Technology, Hamirpur, IND

2. Computer Science and Engineering, Indian Institute of Technology, Patna, IND

3. Computer Science and Engineering, National Institute of Technology, Hamirpur, IND

Received: June 18, 2026 | Review began: July 14, 2026 | Review ended: September 16, 2026 | Published: October 26, 2026

© Copyright 2026

This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

Electronic storage of health records enhances accessibility and facilitates more effective treatment. However, it also introduces cybersecurity risks, data breaches, and vulnerabilities associated with centralized storage. Existing blockchain-based Electronic Medical Record (EMR) solutions struggle with scalability, high storage costs, and inefficient access control. This paper proposes a decentralized EMR management framework that integrates IPFS, blockchain, and AES-256 encryption to enhance security and efficiency. In this approach, AES-256 encrypts medical records before storing them on IPFS. In contrast, only the Content Identifier (CID) is recorded on the blockchain, reducing storage costs and ensuring tamper-proof access control. Smart contracts regulate access, allowing only authorized users to retrieve and decrypt records securely. Results show that the proposed approach offers improved security with minimal overhead and resilience against certain attacks.

Categories: Blockchain and Cryptocurrency, Cryptography, Data Security

Keywords: electronic health records, blockchain, ipfs, security, aes 256

Introduction

In recent years, the healthcare industry has transitioned from a hospital-centric to a patient-centric model [1], enabling patients to receive medical services at their convenience. The advent of remote health monitoring and IoT-driven diagnostics has revolutionized healthcare, allowing for real-time data collection and seamless access to medical records [2]. The development of electronic medical records (EMRs) is a crucial advancement in telemedicine. EMRs represent a digital transformation of traditional paper-based medical records, enabling electronic storage, retrieval, statistical analysis, and seamless data exchange [3]. EMRs incorporate multimedia elements such as audio, video, and medical imaging, facilitating comprehensive patient history management. Their implementation significantly enhances network-based access, diagnosis efficiency, and real-time medical consultations, thereby promoting advancements in telemedicine and global healthcare interoperability [4]. Authorized healthcare professionals can securely access patient records from anywhere, improving work efficiency, treatment accuracy, and overall healthcare quality. However, the widespread adoption of cloud-based storage and IoT-driven medical systems introduces critical security and privacy challenges [5]. The interconnectivity of healthcare devices increases exposure to cyber threats, unauthorized access, and data breaches, raising serious concerns about patient confidentiality, data integrity, and regulatory compliance in the Internet of Medical Things (IoMT) ecosystem. Traditional centralized cloud storage models suffer from single points of

How to cite this article:

Rani D, Prasad M, Kumar R (October 26, 2026) Securing Electronic Medical Records Using Blockchain and Cryptography: A Decentralized Approach. Cureus J Comput Sci 3 : es44389-026-00337-y. DOI <https://doi.org/10.7759/s44389-026-00337-y>

failure, making medical records vulnerable to tampering, data loss, and unauthorized modifications. As a result, there is an urgent need for a secure, decentralized, and tamper-resistant storage architecture that ensures privacy, integrity, and controlled access to sensitive medical data.

To address these challenges, this paper proposes a blockchain-integrated framework for secure medical data storage and access control [6]. The framework leverages Advanced Encryption Standard (AES-256) for data confidentiality, an InterPlanetary File System (IPFS) for decentralized storage, and a Proof of Authority (PoA)-based blockchain for block creation and user authorization. Additionally, smart contracts enforce access permissions, ensuring that only authorized entities can retrieve and decrypt patient records. The proposed system aims to enhance security, privacy, and interoperability by integrating these technologies while mitigating cyber threats and data integrity risks in modern IoT-driven healthcare environments.

In addition to using existing technologies, the main contributions of this work are: (i) securely managing encryption keys off-chain so that the actual symmetric keys are not stored on the blockchain, (ii) clearly separating the role of blockchain for data integrity and auditing from encryption for data confidentiality, while integrating both into the same access-control process, and (iii) using a PoA permissioned blockchain designed for a trusted group of hospitals rather than a fully public blockchain.

The remaining paper is as follows: the Materials and Methods section contains various related work reviews, the workflow of the proposed approach and methodology; the Results and Discussion section consists of implementation details, system setup, along with result analysis on various parameters; and finally, the Conclusion section contains the conclusion and future scope.

Materials And Methods

Related work

With the rising prevalence of diseases, the demand for healthcare services is significantly increased, leading to the continuous generation of new medical records with every patient visit. Effective management of these records is crucial for accurate diagnosis. However, online storage introduces critical security challenges, making sensitive medical data vulnerable to cyberattacks, unauthorized access, and data breaches. These challenges have drawn significant research attention, prompting scholars and scientists to explore robust security mechanisms for safeguarding EMRs. This section reviews existing studies on secure data storage, privacy-preserving mechanisms, and efficient sharing of medical records, highlighting blockchain-based, cryptographic, and decentralized storage solutions proposed to enhance the security and integrity of healthcare data. Adhikari et al. [7] review the integration of Blockchain and IoT for securing healthcare data, highlighting its tamper-proof structure and enhanced privacy. Limitations include IoT device computational constraints and the complexity of Blockchain consensus mechanisms, which challenge widespread adoption and interoperability. Kaur et al. [8] proposed a dual-channel blockchain framework that combines Rivest-Shamir-Adlema (RSA) and AES encryption techniques for securing healthcare data, addressing privacy concerns, and scalability issues. For managing the access of an access control list, and IPFS is used as an off-chain data storage. However, delay increases with the excess involvement of nodes and the RSA encryption technique, leading to complicated data retrieval management. Tahesdoost [9] explores blockchain's potential in securing healthcare data through decentralization, permissioned blockchain, zero-knowledge proofs, privacy-preserving smart contracts, and immutability. Limitations include interoperability, scalability, and regulatory compliance challenges, which may hinder widespread adoption in the healthcare sector. Rani et al. [10] proposed a hierarchical architecture using fog computing for the security of medical data. It uses the Attribute-based encryption scheme with minimum processing. Vinothkumar et al. [11] proposed the hybrid attribute-based encryption (HABE) that combines the cipher-based and elliptical curve cryptography (ECC) in the paper, ensuring the confidentiality and integrity of healthcare data. Access controls, such as trust-based data level access control (TDLA), restrict user access, enhancing security against unauthorized data manipulation or compromise. Abirami et al. [12] proposed a data encryption technique using ECC-AES, which ensures confidentiality by protecting healthcare data from unauthorized access. Access control systems further enhance security by regulating who can view or modify data, safeguarding patient

How to cite this article:

information, and ensuring compliance with HIPAA and GDPR. Rani et al. [6] proposed a blockchain-based data storage and securely sharing. AES encryption along with IPFS for distributed data storage is used. Access control is achieved through a smart contract that helps in the authentication of authorized users. Nortey et al. [13] used Hyperledger Fabric, a permissioned blockchain, which ensured secure and patient-centric EHR management by granting patients administrative control over their records. Hospitals maintain the network, but third-party access requires explicit patient approval via smart contracts and dedicated channels. The ledger stores either hashed EHR data or references to external servers, ensuring efficient storage and retrieval. Ajayi et al. [14] proposed a blockchain-based secure storage and exchange of EHRs among healthcare providers. It ensures scalability, integrity verification, and protection against insider and outsider threats during data storage and transmission. The architecture standardizes EHR formats, enabling easier malicious activity detection while ensuring data consistency across systems. Additionally, it permanently stores verified EHRs in a distributed manner, allowing secure access and sharing upon request. Pandit et al. [15] evaluates various blockchain-based EHR management solutions, and propose a patient-centric EHR system using Ethereum blockchain. The model incorporates Attribute-Based Access Control (ABAC) to enforce secure and decentralized identity management for EHR sharing. Additionally, the study compares the performance and gas consumption of the proposed system with existing solutions, demonstrating its efficiency and scalability.

From the reviewed literature, it is evident that cryptographic techniques, decentralized storage solutions, IPFS, and access control methods can be used in addressing security concerns. Based on these findings, this paper proposes a secure and decentralized framework along with a cryptographic technique to overcome the limitations identified in existing works by providing a scalable, efficient, and privacy-preserving solution for medical data management. The next section details the proposed methodology, system architecture, and implementation approach.

Compared with recent related systems, the proposed framework addresses some limitations in existing approaches. Kaur et al. [8] use RSA for key protection and an off-chain access-control list, which can increase processing time. Rani et al. [6] use AES and IPFS but do not clearly explain how encryption keys are managed. Nortey et al. [13] use Hyperledger Fabric for identity and access control but do not include a separate encryption mechanism for the stored data. Pandit et al. [15] mainly study Ethereum gas costs and ABAC and do not provide an off-chain encryption mechanism.

In contrast, the proposed framework encrypts data using AES-256, stores the encrypted data in IPFS, keeps only the CID on the blockchain, and clearly defines how encryption keys are securely handled using an envelope-based approach. This combination addresses the key-management and off-chain data-protection gap identified in the related works.

Proposed work

Ensuring the security, privacy, and integrity of medical data becomes a significant challenge with the widespread adoption of EMRs and IoMT devices [16]. Traditional centralized storage systems are vulnerable to single points of failure, unauthorized access, and cyber threats, making them inadequate for safeguarding sensitive healthcare data. To address these challenges, recent advancements in blockchain technology, cryptographic encryption, and decentralized storage have introduced innovative solutions for secure medical data management. This section presents the proposed blockchain-based methodology for secure storage (store data) and controlled sharing of medical records (retrieve data) with authorized healthcare professionals as shown in Figure 1.

How to cite this article:

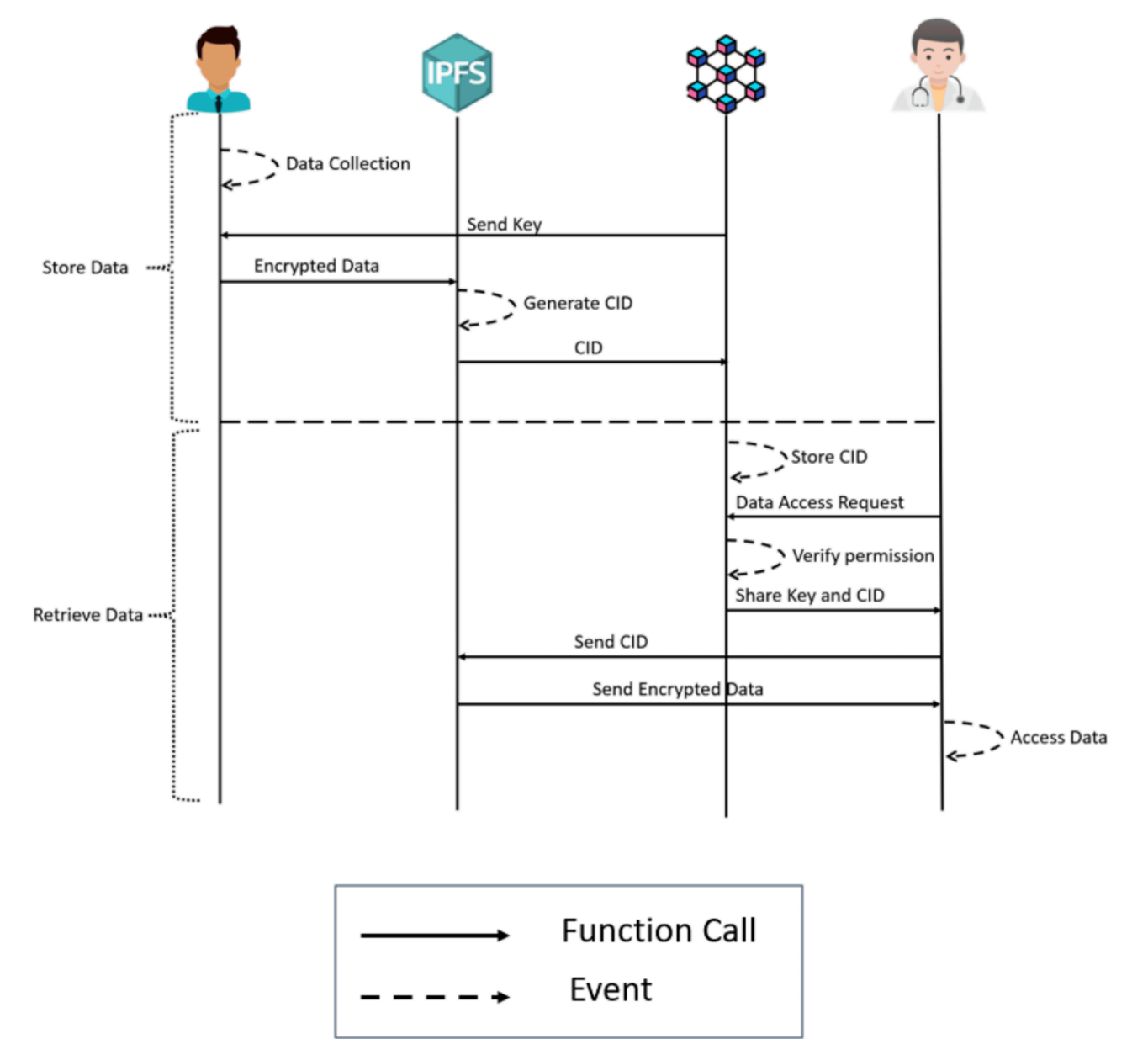


FIGURE 1: Workflow of proposed approach

The framework consists of four key components:

(1) AES Encryption: AES is a symmetric key encryption algorithm widely used for securing sensitive data due to its high efficiency, strong security, and fast encryption speed [17]. It operates on fixed-size data blocks (128-bit) and supports key lengths of 128, 192, or 256 bits, with AES-256 being the most secure. Since it uses the same key for encryption and decryption, key management is crucial for ensuring security. AES is widely adopted in healthcare, banking, and cloud security for encrypting sensitive data, making it an ideal choice for protecting EMRs and securing decentralized storage systems.

How to cite this article:

(2) **Blockchain:** Blockchain is a decentralized, immutable, and distributed ledger technology that enables secure, transparent, and tamper-proof data management [18]. It operates by storing data in blocks, which are cryptographically linked to form a chain, ensuring that once a record is added, it cannot be altered or deleted. This ensures trust, security, and transparency across various applications, including finance, healthcare, and supply chain management. The proposed approach uses a private blockchain network that maintains the Content Identifier (CID), encryption key management, and access permissions, enabling secure authentication and transparent access control.

(3) **Smart Contracts:** A smart contract is a self-executing program stored on a blockchain, designed to automatically enforce and execute agreements without the need for intermediaries [19]. Smart contracts play a crucial role in access control, allowing only authorized users to retrieve encrypted medical records while maintaining an audit trail of access requests.

(4) **IPFS:** It is a decentralized, peer-to-peer storage protocol designed to enable secure, scalable, and efficient data storage and retrieval. Unlike traditional cloud-based systems that rely on centralized servers, IPFS uses content-addressing, where each file is assigned a unique cryptographic hash known as a CID [20]. IPFS is widely adopted for secure document storage, blockchain integration, and decentralized applications (DApps), making it an ideal solution for privacy-sensitive data management.

Threat model

The adversary is assumed capable of: (i) passively eavesdropping on network traffic between clients, IPFS nodes, and blockchain peers; (ii) attempting unauthorized retrieval or tampering with data stored on IPFS; (iii) submitting unauthorized transactions to the smart contract to request access; and (iv) compromising a single IPFS or blockchain node, but not a majority of PoA validators, which are assumed to be operated by authenticated hospital-consortium entities. The trusted computing base includes the PoA validator set and the correctness of the deployed smart-contract logic; key-generation endpoints (patient/provider devices) are assumed honest but are the weakest link if compromised, which motivates the key-management.

Methodology

EMRs contain prescriptions, lab reports, diagnostic results, and other essential health information. It is important to save medical records for better treatment, and they can be needed anytime in hospitals. Saving records in paper-based systems faces challenges related to accessibility and management, whereas digital storage offers efficient retrieval and enhanced protection [21]. Historically, medical records have been hospital-centric, restricting patient control over their own data. However, the shift toward a patient-centric model is gaining traction, emphasizing user ownership, decentralized storage, and secure data access. To achieve tamper-proof, privacy-preserving, and controlled access to EMRs, this paper proposes a blockchain-based framework that integrates encryption, decentralized storage, and access control mechanisms, ensuring that patients retain full authority over their medical data. The proposed system addresses two major challenges: decentralized storage and access control for data sharing. It consists of the following key steps:

Data Collection and Encryption

- Medical records are collected from hospitals, laboratories, healthcare devices, and doctors, converting them into electronic format for digital storage.
- To protect sensitive medical data, AES-256 is employed. EMRs are encrypted before storage, ensuring that only authorized users can access them.
- The encryption key is managed securely via the blockchain, where the symmetric key is stored and retrieved when required.
- The AES-256 key is generated on the client side when each record is encrypted. The key is never stored directly on the blockchain. Instead, it is encrypted using the recipient's public key, and only its reference is linked to the corresponding CID through the smart contract. Revocation is handled by removing the user from `authorizedUsers[]`, preventing future

How to cite this article:

key access, although previously decrypted copies cannot be revoked. Key recovery currently uses administrative re-encryption with a new key pair.

- Once encrypted, the data is sent to IPFS for decentralized storage.

Decentralized Storage Using IPFS

- The encrypted medical records are stored in IPFS, which generates a unique CID-a fixed-size cryptographic hash representing the encrypted data.
- Instead of storing the entire record on the blockchain, only the CID is recorded, significantly reducing storage costs while maintaining data integrity and retrievability.

Data Access and Retrieval

- When access to medical records is requested, the blockchain verifies user authorization using smart contracts, allowing only authorized users (e.g., doctors, hospitals, and researchers) to access records.
- If access is approved, the blockchain retrieves the CID and the recipient-wrapped AES key reference, granting access only to authorized users.
- The user then fetches the encrypted data from IPFS using the CID, unwraps the AES key with their own private key, and decrypts the record.

Security requirements

The proposed approach ensures compliance with essential security requirements necessary for modern healthcare data management before implementation. The following security aspects are achieved through this framework:

- Confidentiality: EMRs are encrypted using AES-256 before storage, and only the CID is recorded on the blockchain. This provides dual-layer security, ensuring protection through both encryption and blockchain immutability.
- Resilience Against Single-Point Failure: Traditional centralized storage solutions are prone to single points of failure, whereas the decentralized architecture guarantees continuous data availability and fault tolerance.
- Availability: The distributed nature of IPFS ensures that medical records remain accessible to authorized users at all times, eliminating dependency on a central server.
- Access Control: Smart contracts enforce role-based access control, allowing the granting and revocation of access permissions securely and transparently.
- Scalability: The workload is distributed across multiple nodes, allowing for seamless integration of additional authenticated users, ensuring a scalable and efficient network.

Access control

Access control using smart contract Solidity language is used to run the smart contract. Algorithm (1) is used to manage the access control, where the user (owner) is authorized to grant and revoke access permission.

Algorithm 1 Smart Contract for managing access control

```
1: Algorithm Manage Access (user, grant)
2: {
3:   if (msg.sender == patient) then
4:     authorizedUsers[user] ← grant
5:   Emit AccessUpdated(user, grant)
6:   if (grant == true) then
7:     Return "Access Granted"
8:   else
9:     Return "Access Revoked"
10:  end if
11: else if authorizedUser[msg.sender] == true then
12:  Return "Medical Record Accessed: [Encrypted Data]"
```

How to cite this article:

```
13: else
14: Return "Access Denied"
15: end if
16: }
17: end Algorithm
```

Blockchain integrity versus cryptographic confidentiality

Blockchain immutability provides integrity and auditability: once a CID and an access-control decision are committed, they cannot be silently altered, and every grant/revoke action is traceable via emitted events. It does not, by itself, provide confidentiality - anyone with ledger read access could see that a transaction occurred, who initiated it, and (if not designed carefully) potentially the CID pointing to the encrypted payload. Confidentiality of the medical record content is provided entirely by AES-256 encryption performed before data leaves the client, independent of the ledger.

Practical considerations: emergency access, insider threats, and ledger privacy

The current smart contract follows the normal authorization process, which may not be suitable for emergency situations. A future extension could provide time-limited emergency access, approved by a hospital administrator and automatically revoked after a short period. If an authorized account is compromised, the attacker may access records within its permissions until access is revoked. Prompt revocation and audit-log monitoring can reduce this risk but cannot completely prevent it. Since blockchain records cannot be deleted, the framework stores only the CID and access-related metadata, not patient or clinical information, to avoid exposing sensitive health data. However, transaction details such as access times and patterns may still reveal sensitive information, which is left for future work.

Results And Discussion

Experimental setup

This section includes the experimental setup for the proposed work implementation and the result analysis. The implemented system configuration is Ryzen 7, 5000 series, 16GB RAM, and 64-bit Windows operating system for execution. Private Ethereum is used for implementation. The software stack consists of Go Ethereum (Geth) for private blockchain setup, Truffle for smart contract deployment, Node.js with Web3.js for interaction, and IPFS for decentralized storage. PoA consensus algorithm is used to join only the authorized users to the network. The setup involves several steps:

- (1) Installing Ethereum (Geth) and initializing a private blockchain using a custom genesis.json file.
- (2) Deploying smart contracts using Solidity and Truffle to manage access control.
- (3) Setting up IPFS for encrypted data storage and obtaining a CID.
- (4) Connecting a frontend using Web3.js to interact with the Ethereum blockchain.
- (5) Running and testing the system by starting the Ethereum node, IPFS daemon, and executing smart contract functions for secure medical data access and sharing.

Cryptographic technique

The proposed approach employs the AES-256 encryption algorithm due to its strong security and efficient encryption process. A performance is compared for various file sizes to evaluate the encryption time of different cryptographic techniques, including AES-256, DES, 3DES, and RSA. As shown in Figure 2, AES-256 demonstrated the fastest encryption time, whereas RSA exhibited the highest computational overhead. Table 1 presents the processing time of AES-256, DES, 3DES, and RSA for file sizes ranging from 1 MB to 100 MB. As expected, the processing time increases with the increase in file size for all encryption algorithms. AES-256 consistently demonstrates the lowest computational overhead, reducing the processing time by approximately 15-20% compared with DES and 3DES and by 18-35% compared with RSA, depending on the file size.

How to cite this article:

File Size (MB)	AES-256 (ms)	DES (ms)	3DES (ms)	RSA (ms)
1	950	1,100	1,400	1,900
20	2,200	2,700	3,800	3,900
40	5,250	6,250	6,350	7,400
60	6,450	7,600	8,250	8,800
80	8,000	9,450	9,000	9,750
100	9,300	10,250	10,450	11,450

TABLE 1: Encryption time vs file size

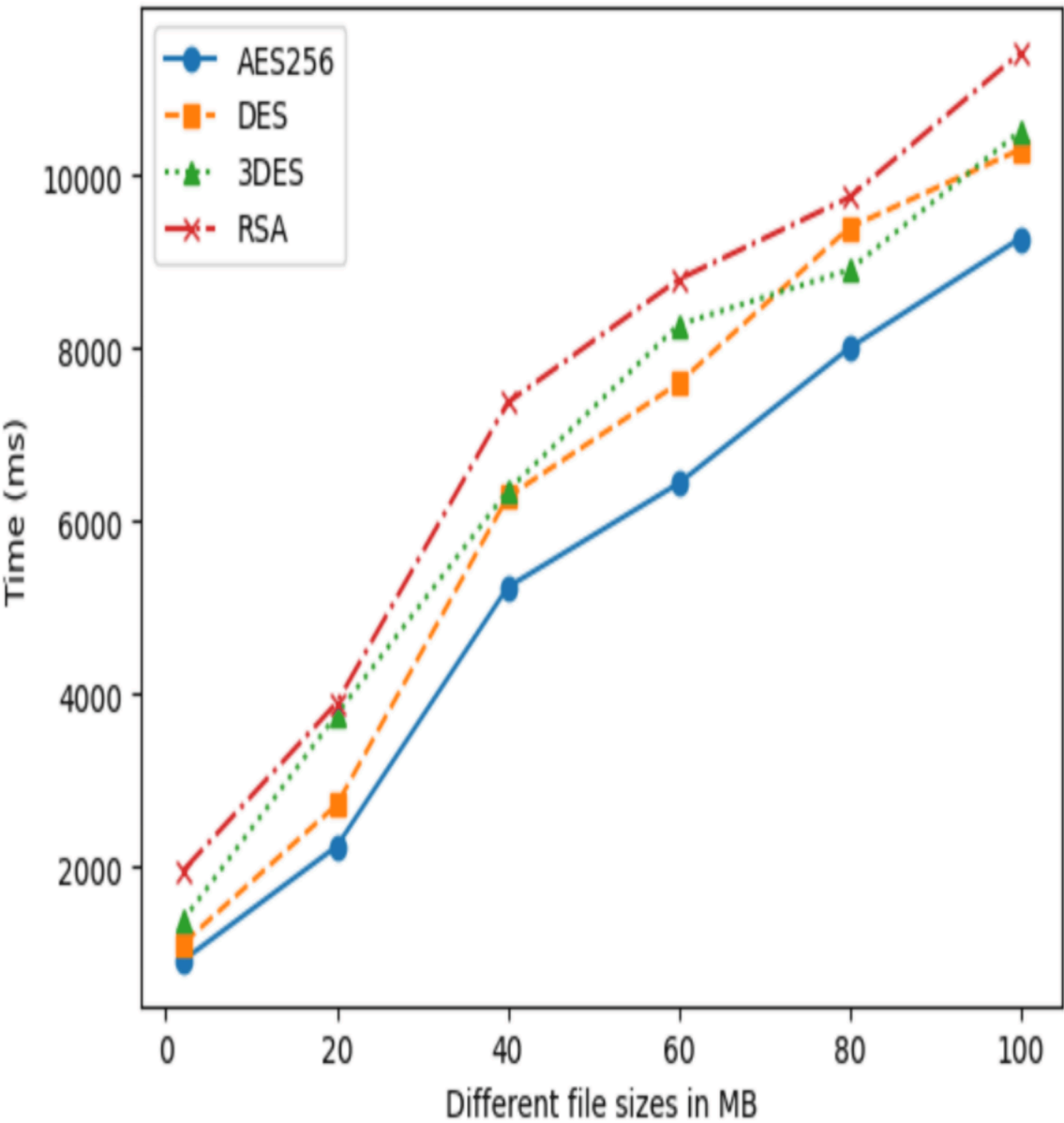


FIGURE 2: Different encryption algorithm with different data sizes

Resilience against attacks

The proposed approach shows resilience against some of the major attacks that usually occur in recent times on the data that is kept online. Table 2 shows how each attack type is addressed by the security mechanisms used in the framework, based on the known properties of AES-256, PoA consensus, and content-addressed storage. However, these results are based on the system design and not on actual penetration testing or attack simulations. Therefore, the table is not presented as a measured security result. In future work, we plan to perform reproducible attack simulations, such as MITM and brute-force attacks, on the testbed to provide measured security results.

Distributed Denial-of-Service (DDoS) Attack

How to cite this article:

The proposed blockchain-IPFS architecture eliminates the single point of failure found in centralized systems. Distributed storage (IPFS) and decentralized transaction processing improve availability and enhance resilience against DDoS attacks.

Man-in-the-Middle (MITM) Attack

AES-256 encryption secures all transmitted data, while IPFS content hashes ensure data integrity. Even if communication is intercepted, attackers cannot decrypt or modify the data without the encryption key.

Brute Force Attack

The framework employs AES-256 encryption, whose 256-bit key space makes brute-force attacks computationally infeasible, thereby ensuring strong confidentiality of healthcare data.

Storage Attack

Only the hash of encrypted data is stored on the blockchain, while the encrypted data resides in IPFS. Even if an attacker compromises the storage, recovering the original plaintext without the decryption key is practically impossible.

Impersonation Attack

User authentication is based on public-private key cryptography, while the PoA consensus permits only authenticated validators to verify transactions, preventing unauthorized identity impersonation.

Attack	Countermeasures
Distributed Denial of Services (DDoS)	Blockchain, IPFS
Man-in-the-Middle attack (MITM)	AES 256 encryption, Hash of IPFS
Brute Force Attack (BFA)	AES 256
Storage Attack (SA)	Hash of encrypted data
Impersonation Attack	PoA algorithm

TABLE 2: Attacks and their countermeasures

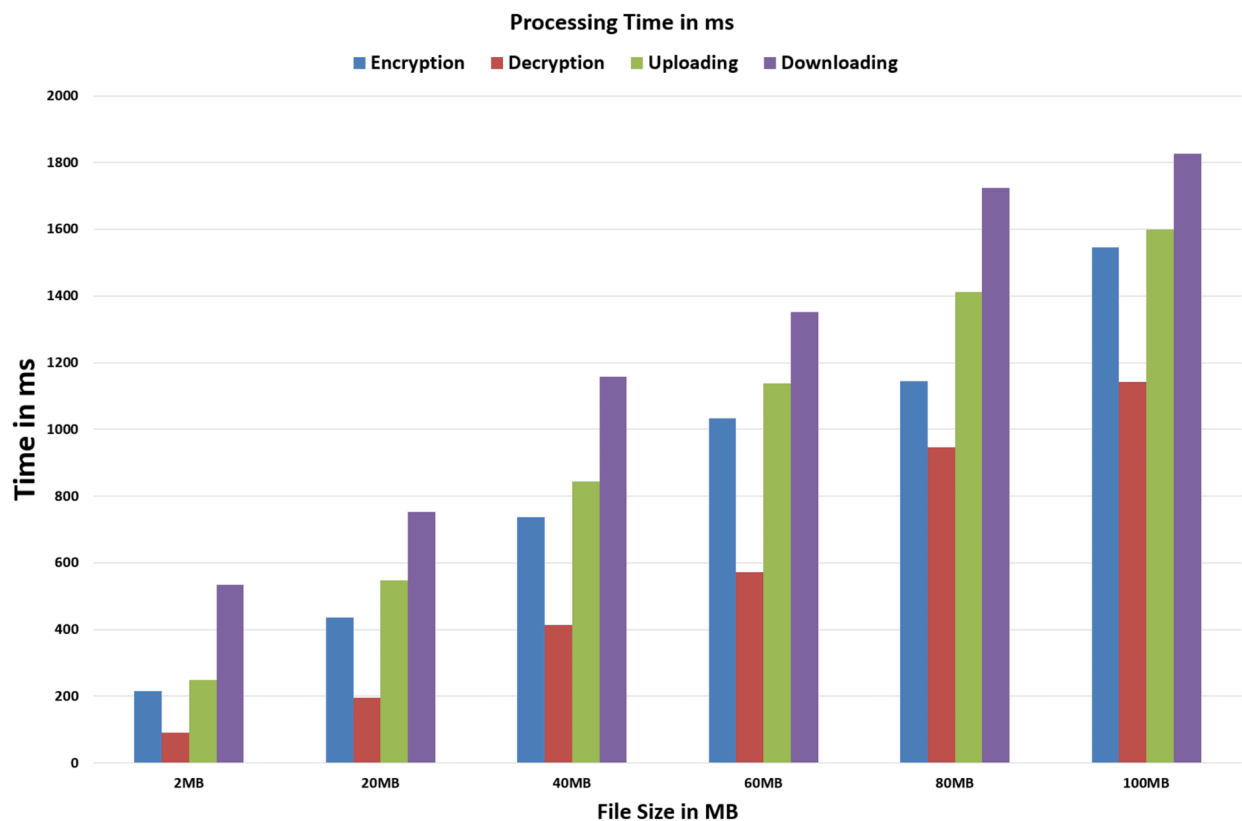


FIGURE 3: Processing cost in ms

Overhead Computation

According to the proposed approach, there are several stages where computation overhead is involved, such as data creation, data encryption and decryption, key management, sending data to IPFS, and retrieving data from IPFS. The PoA consensus algorithm is chosen as it ensures faster block creation and efficient user authorization. For encryption, AES-256 is utilized due to its robust security and high efficiency, ensuring that data remains protected without excessive processing delays. Although storing and retrieving data from IPFS introduces computational costs, it significantly reduces blockchain storage expenses and enhances scalability. For executing different functions, it requires some processing time. The processing time in ms with respect to different file sizes is represented in Figure 3. The file size ranges from 2 MB to 100 MB. The encryption and decryption time increases when the file size increases, so uploading and downloading time also increases.

Table 3 presents the processing time for encryption, decryption, uploading, and downloading operations for file sizes ranging from 2 MB to 100 MB. The encryption process requires more time (from 210 ms to 1540 ms) because it involves cryptographic key generation, secure key distribution, and AES-256 encryption before data storage. In contrast, decryption incurs lower overhead (from 90 ms to 1140 ms) as it only performs the decryption operation using the previously established key. The uploading operation (from 245 ms to 1595 ms) includes both data encryption and subsequent storage of the encrypted data/IPFS hash on the blockchain, resulting in additional communication overhead. Similarly, downloading records the highest processing time (from 530 ms to 1820 ms) because it involves retrieving the encrypted data and cryptographic key from the blockchain/IPFS network, followed by decryption before delivering the original data to the authorized doctor.

How to cite this article:

File Size (MB)	Encryption (ms)	Decryption (ms)	Uploading (ms)	Downloading (ms)
2	210	90	245	530
20	430	190	545	750
40	735	410	840	1155
60	1030	570	1130	1350
80	1140	945	1410	1715
100	1540	1140	1595	1820

TABLE 3: Processing time of different operations

Scalability, Latency, and Comparative Discussion

The reported experiments only measure the encryption and decryption time of AES-256, DES, 3DES, and RSA for different file sizes. They do not yet measure complete transaction latency on the PoA blockchain, IPFS retrieval time under multiple users, blockchain storage growth, or system throughput with multiple hospitals. Therefore, the scalability and efficiency claims are now presented as expected benefits of the architecture, based on the known properties of PoA consensus and IPFS, rather than as experimentally measured results. The framework stores only a small CID (about 34 bytes) on the blockchain, regardless of the file size, while the actual encrypted data is stored off chain in IPFS. Interoperability with existing hospital systems such as HL7/FHIR-based EMR platforms and access-control performance under high concurrent requests were also not evaluated. These aspects, along with a comparison of latency and throughput with the related dual-channel and ABAC-based systems, are planned as future work to provide quantitative evidence of the system's efficiency and scalability.

Conclusions

The proposed framework uses IPFS for decentralized off-chain storage of AES-256 encrypted medical records, while the corresponding CID is stored on the blockchain for secure and efficient data retrieval. Smart contracts manage user authorization, and the encryption mechanism helps protect the confidentiality of medical data. The preliminary encryption and decryption results support the practical use of the proposed approach; however, its scalability, efficiency, and security should be considered architecture-level benefits rather than fully validated outcomes, as end-to-end performance and formal security evaluation were not conducted.

As future work, the framework will be extended to support distributed database storage for handling larger volumes of medical data and will be evaluated in a real hospital environment involving patients, doctors, and hospitals. Further work will also include end-to-end latency and throughput evaluation, emergency-access mechanisms, and threshold-based key recovery to provide a more comprehensive validation of the framework.

How to cite this article:

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Deepa Rani Dr., Manikant Prasad, Rajeev Kumar

Acquisition, analysis, or interpretation of data: Deepa Rani Dr.

Drafting of the manuscript: Deepa Rani Dr., Manikant Prasad

Critical review of the manuscript for important intellectual content: Manikant Prasad, Rajeev Kumar

Supervision: Rajeev Kumar

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue.

Animal subjects: All authors have confirmed that this study did not involve animal subjects or tissue.

Conflicts of interest: In compliance with [the ICMJE uniform disclosure form](#), all authors declare the following:

- **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work.
- **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work.
- **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Data Availability Statements

Data sharing is not applicable. This work is theoretical; no datasets were generated or analyzed.

References

1. Rani D, Kumar R, Chauhan N: [Study influencing factors of maternal health and the role of Internet of Things \(IoT\) to improve maternal care](#). SN Computer Science. 2024, 5:778. [10.1007/s42979-024-03129-0](#)
2. Fan K, Wang S, Ren Y, Li H, Yang Y: [MedBlock: efficient and secure medical data sharing via blockchain](#). Journal of Medical Systems. 2018, 42:136. [10.1007/s10916-018-0993-7](#)
3. Chenthara S, Ahmed K, Wang H, Whittaker F: [Security and privacy-preserving challenges of e-health solutions in cloud computing](#). IEEE Access. 2019, 7:74361-74382. [10.1109/access.2019.2919982](#)
4. Sivan R, Zukarnain ZA: [Security and privacy in cloud-based e-health system](#). Symmetry. 2021, 13:742. [10.3390/sym13050742](#)
5. Marichamy VS, Natarajan V: [Blockchain based securing medical records in big data analytics](#). Data & Knowledge Engineering. 2023, 144:102122. [10.1016/j.datak.2022.102122](#)
6. Rani D, Kumar R, Chauhan N: [A secure framework for IoT-based healthcare using blockchain and IPFS](#). Security and Privacy. 2024, 7:e348. [10.1002/spy2.348](#)
7. Adhikari N, Ramkumar M: [IoT and blockchain integration: applications, opportunities, and challenges](#). Network. 2023, 3:115-141. [10.3390/network3010006](#)
8. Kaur J, Rani R, Kalra N: [Healthcare data security and privacy protection framework based on dual channel blockchain](#). Transactions on Emerging Telecommunications Technologies. 2025, 36:e70049. [10.1002/ett.70049](#)

How to cite this article:

Rani D, Prasad M, Kumar R (October 26, 2026) Securing Electronic Medical Records Using Blockchain and Cryptography: A Decentralized Approach. Cureus J Comput Sci 3 : es44389-026-00337-y. DOI <https://doi.org/10.7759/s44389-026-00337-y> Page 13 of 14

9. H Taherdoost: [Exploring blockchain solutions in healthcare data management and patient data privacy](#). Blockchain-Enabled Solutions for the Pharmaceutical Industry. Ritika W, Parul A, Parma N, Vishal J, Vinay K (ed): Scrivener Publishing, Beverly, MA; 2025. 1-18. [10.1002/9781394287970.ch1](#)
10. Shruti, Rani S, Srivastava G: [Secure hierarchical fog computing-based architecture for industry 5.0 using an attribute-based encryption scheme](#). Expert Systems with Applications. 2024, 235:121180. [10.1016/j.eswa.2023.121180](#)
11. Vinothkumar S, Amutharaj J: [Secure healthcare data transaction using hybrid attribute-based encryption with access control policy on cloud](#). International Journal of System of Systems Engineering. 2024, 14:159-173. [10.1504/ijssse.2024.137071](#)
12. Abirami N, Anbarasi MS: [An Efficient Multilayer approach for Securing E-Healthcare Data in Cloud using Crypto - Stego Technique](#). Engineering World. 2024, 6:128-135. [10.37394/232025.2024.6.13](#)
13. Nortey RN, Yue L, Agdedanu PR, Adjeisah M: [Privacy module for distributed electronic health records \(EHRs\) using the blockchain](#). 2019 IEEE 4th International Conference on Big Data Analytics (ICBDA), Suzhou, China. 2019, 369-374. [10.1109/ICBDA.2019.8713188](#)
14. Ajayi O, Abouali M, Saadawi T: [Secured inter-healthcare patient health records exchange architecture](#). 2020 IEEE International Conference on Blockchain (Blockchain), Rhodes, Greece. 2020, 456-461. [10.1109/Blockchain50366.2020.00066](#)
15. Pandit NK, Das S, Panda CK: [A patient-centric EHR management system using Ethereum blockchain](#). 2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS), Bangalore, India. 2024, 1-5. [10.1109/ICITEICS61368.2024.10625459](#)
16. Dwivedi R, Mehrotra D, Chandra S: [Potential of Internet of Medical Things \(IoMT\) applications in building a smart healthcare system: A systematic review](#). Journal of Oral Biology and Craniofacial Research. 2022, 12:302-318. [10.1016/j.jobcr.2021.11.010](#)
17. Das S, Namasudra S: [A novel hybrid encryption method to secure healthcare data in IoT-enabled healthcare infrastructure](#). Computers and Electrical Engineering. 2022, 101:107991. [10.1016/j.compeleceng.2022.107991](#)
18. Taherdoost H: [The role of blockchain in medical data sharing](#). Cryptography. 2023, 7:36. [10.3390/cryptography7030036](#)
19. Lee J-S, Chew C-J, Liu J-Y, Chen Y-C, Tsai K-Y: [Medical blockchain: Data sharing and privacy preserving of EHR based on smart contract](#). Journal of Information Security and Applications. 2022, 65:103117. [10.1016/j.jisa.2022.103117](#)
20. Azbeg K, Ouchetto O, Andaloussi SJ: [BlockMedCare: A healthcare system based on IoT, Blockchain and IPFS for data management security](#). Egyptian Informatics Journal. 2022, 23:329-343. [10.1016/j.eij.2022.02.004](#)
21. Nemiwal G, Rani D, Kumar R: [Transformer-based network intrusion detection: A multi-dataset analysis](#). 2025 3rd International Conference on Inventive Computing and Informatics (ICICI), Bangalore, India. 2025, 966-972. [10.1109/ICICI65870.2025.11069717](#)

How to cite this article: