Healthchain: A novel framework on privacy preservation of electronic health records using blockchain technology

PLOS ONE

RESEARCH ARTICLE

Healthchain: A novel framework on privacy preservation of electronic health records using blockchain technology

Shekha Chenthara[1]*, Khandakar Ahmed[1], Hua Wang[1], Frank Whittaker[1], Zhenxiang Chen[2]

1 Institute for Sustainable Industries and Liveable Cities, Victoria University, Melbourne, Victoria, Australia, **2** School of Information Science and Engineering, University of Jinan, Jinan, China

* shekha.chenthara@live.vu.edu.au

OPEN ACCESS

Citation: Chenthara S, Ahmed K, Wang H, Whittaker F, Chen Z (2020) Healthchain: A novel framework on privacy preservation of electronic health records using blockchain technology. PLoS ONE 15(12): e0243043. https://doi.org/10.1371/journal.pone.0243043

Editor: Xiaodi Huang, Charles Sturt University, AUSTRALIA

Received: September 1, 2020

Accepted: November 15, 2020

Published: December 9, 2020

Data Availability Statement: All relevant data are within the manuscript and its Supporting Information files.

Funding: The author(s) received no specific funding for this work.

Competing interests: The authors have declared that no competing interests exist.

Abstract

The privacy of Electronic Health Records (EHRs) is facing a major hurdle with outsourcing private health data in the cloud as there exists danger of leaking health information to unauthorized parties. In fact, EHRs are stored on centralized databases that increases the security risk footprint and requires trust in a single authority which cannot effectively protect data from internal attacks. This research focuses on ensuring the patient privacy and data security while sharing the sensitive data across same or different organisations as well as healthcare providers in a distributed environment. This research develops a privacy-preserving framework viz Healthchain based on Blockchain technology that maintains security, privacy, scalability and integrity of the e-health data. The Blockchain is built on Hyperledger fabric, a permissioned distributed ledger solutions by using Hyperledger composer and stores EHRs by utilizing InterPlanetary File System (IPFS) to build this healthchain framework. Moreover, the data stored in the IPFS is encrypted by using a unique cryptographic public key encryption algorithm to create a robust blockchain solution for electronic health data. The objective of the research is to provide a foundation for developing security solutions against cyber-attacks by exploiting the inherent features of the blockchain, and thus contribute to the robustness of healthcare information sharing environments. Through the results, the proposed model shows that the healthcare records are not traceable to unauthorized access as the model stores only the encrypted hash of the records that proves effectiveness in terms of data security, enhanced data privacy, improved data scalability, interoperability and data integrity while sharing and accessing medical records among stakeholders across the healthchain network.

1 Introduction

With the advancement in information and communication technology (ICT), most of the healthcare organizations paved the way for Electronic Health Records (EHRs) from paper based records. EHR, Electronic Health Data (EHD), Electronic Medical Records (EMR) are

digitalized patient records encompassing a huge variety of medical data such as medical histories, demographic information, laboratory test reports and other sensitive patient personal information including social security number and credit card information [1]. The large scale generation and rampant usage of health information in the big data era increases the role of cloud networks not only to house the large amount of data but also to facilitate its access across the Internet [2–5]. Moreover the lion's share of medical data is extremely sensitive and confidential, its storage on third party centralized servers naturally increases the privacy and security vulnerabilities that leads to several attacks includes DDoS attack [6] and Ransomware attacks that have greater ramifications beyond financial or privacy breach [7, 8]. Considering the vulnerable nature of healthcare data in the public domain and unavailing security frameworks, there is an imminent need to protect the data and devise a secure, efficient and effective mechanism to facilitate share and access of data among various stakeholders [2, 9, 10]. Blockchain technology has a large potential to bring significant efficacies to financial transactions, global supply chains, asset ledgers, healthcare and decentralized social networking.

Blockchain is one of the solutions to overcome most of the limitations in the existing distributed environment by introducing a patient centered electronic health system namely Patient Controlled Electronic Health Record System (PCEHR), in which the patient is the universal consent provider of their data to all stakeholders except in emergency situations. Blockchain is a public, decentralized, append-only, immutable digital ledger with a time stamped series of transactions called blocks that are linked to form a chain that are secured by means of Public Key Encryption cryptographic principles [11, 12]. Since the blocks are linked, the data once recorded cannot be altered retroactively without the modification of all subsequent blocks. A cryptographic one way hash function (e.g. SHA-256) is also applied to the blocks to ensure immutability, anonymity and tamper resistant structure for the blocks [13]. Moreover, blockchain uses the consensus protocol mechanism to generate, update and validate transactions for ensuring the security and also employs scripting code to run intelligent smart contracts [14, 15]. In particular, our blockchain network resolves the challenges related with interoperability, scalability, integrity, security and privacy concerns in the health care data systems and delivers a comprehensive clinical care. Our research exploits the inherent properties of blockchain to build a potential framework that fulfills the health care use-cases and supports the shift from institution-driven-interoperability to patient-centric-interoperability. This work employs Hyperledger fabric [16] as the permissioned blockchain solution that provides a framework for securing the interactions within the entities in the healthchain network.

1.1 Motivation

Major drawback in the current system is that since the healthcare records are stored in centralized databases in silos, healthcare data becomes an extremely tempting target for the attackers. Several research studies showed that centralization increases the security risks and requires trust in a single authority. The centralized databases can leave us vulnerable to attacks that escalates in to cyber threats ranging from the recent Ransomware attack [17] to the Equifax attack [18] which hinders the privacy and security of EHRs. Lack of Interoperability in EHR is another main issue faced by healthcare industry today. Health data in the prevalent systems is fragmented and is challenging to share with healthcare providers or stakeholders due to their varying formats and standards. This defines that it is difficult to aggregate and examine patient data that prevents the efficacy of the EHR sharing in emergency situations. Other significant concern faced by health records housed in cloud servers is from internal attacks where the people with authorized credentials within organizations access data such as database administrators or key managers are attackers, which is considerably worse than the external attacks.

Further, when the EHR is deleted from the database of the hospital, the record can be permanently lost which is another issue that requires deliberate attention. It is essential to have a tamper proof system inaccessible to all except to authorized stakeholders. A traditional database system addresses these requirements only in part and thus alternative technologies need to be explored. In addition, conditional access of patient records to various physicians, laboratories and pharmacies rather than full public access is also crucial to preserve privacy of patient records. Moreover, in the existing system, patients are not in complete control of the health records since it is managed by the service providers. Because of the incessant increase of healthcare data, secure storage and scalability of medical records are a major concern. Considering the vulnerable nature of healthcare data, efficient data sharing between the stakeholders in a public domain is a complicated task. Despite the great features the existing healthcare industry provides, it fails to provide an efficient way to store, share and analyze the health data in a globally unified way. The available privacy preserving mechanisms are inadequate to ensure foolproof security for the seemly management of EHRs in the cloud [19].

In this research work, we propose a blockchain framework based on Hyperledger Fabric [16] and designed a system which can be used for efficient data sharing, health records management and systematic access control. Consequently, this research introduces a permissioned patient-centric blockchain namely Healthchain for EHRs that eliminates most of the bottlenecks and evades the likelihood of single point of failure in existing systems by introducing a distributed ledger platform. The interoperability challenges in healthcare is resolved by the healthchain framework in the way it is built. i.e. Healthchain framework stores the patient history by syncing records in different formats by accessing data via REST server API by employing self-governing and constantly executing smart contracts in the framework. And also the patient has complete control over the healthcare records by providing access and identity permissions to the authorized stakeholders by employing appropriate encryption mechanism and access control permission rules. Moreover, the immutability of health records is also achieved by cryptographically storing the data inside i.e. by storing hash values of data in the blockchain and storing encrypted healthcare records in the offchain IPFS database that makes the framework tamper resistant. Healthchain is a decentralised framework in a way that nobody can tamper the records as the data transactions are linked and a consensus of stakeholders need to agree for adding data in the network. Our system contributes to the healthcare by addressing most of the challenges with data privacy, security, interoperability, scalability, trust, immutability and data integrity.

1.2 Contribution

We present the structure and functionality of a permissioned blockchain based architecture called Healthchain by employing Hyperledger Fabric to securely and scalably share healthcare records to preserve patient privacy, deliver efficient permission management among stakeholders for enhancing collaborative clinical decision support and comprehensive patient care. The main contributions of this paper are summarized as follows:

- Initially, this research builds a patient centric interoperability healthchain framework in which patients will have entire control over their medical records that maintains security, privacy, scalability and integrity of the e-health data. The Healthchain framework is built on Hyperledger fabric, a permissioned distributed ledger solutions by utilizing Hyperledger composer and stores EHR in InterPlanetary File System (IPFS) to build this private healthchain network. Because of its decentralized property this framework ensures no single point of failure and also changes to the blockchain will be visible to the participants of the healthchain network that are immutable.

- To maintain the efficiency and scalability of the blockchain, this research stores only the hash of health records on chain and actual huge data is stored after encryption in the off chain storage framework in IPFS, the decentralized storage. Furthermore, the proposed healthchain framework only allows true records to be added on blockchain which is authenticated by the consensus and the access to the health records are given only based on user permission. Moreover, the data stored in the IPFS will be encrypted by using a unique public key encryption cryptographic algorithm to create robust blockchain solutions for electronic health data.
- Our research design focuses on patient-centric approach where the patient has the complete control to provide access permissions to the authorized stakeholders and does not involve any form of mining incentives beyond the efficient use of the system. This framework developed a working prototype in which the blockchain technique is analyzed and also unravels the possibility of blockchain in healthcare solutions.

The remainder of the paper is structured as follows. Section 2 discusses the related work, whereas section 3 discusses the preliminary components, section 4 explains cryptographic process and architecture of proposed framework; section 5 provides the prototype implementation of the framework; section 6 demonstrates the results; section 7 discusses the analysis and discussion of the proposed framework; and section 8 as conclusion.

2 Related works

This section summarizes the related works pertaining to secure storage and efficient access control schemes implemented in e-healthcare using blockchain technology. For permissionless or public blockchains such as Bitcoin [11] and Ethereum [20], anyone can join as a node in the network since public blockchain doesn't have any network barrier. Moreover, transactions in public chains are transparent and open though anonymity is maintained but is less desirable in healthcare industry which manages sensitive health records. In contrast with the public blockchain, permissioned blockchain or private blockchain such as Hyperledger fabric adopts access control mechanism for determining the addition of a new node to the network. However, the previous studies come with the inadequacy of requisite of mining incentives in the form of ether for performing transactions in healthcare arena.

Several tamper-proof mechanisms are proposed using blockchain technology [21] as shown in Table 1. Yue et al [22] proposed the first scheme using blockchain in healthcare industry that mentioned a Healthcare Data Gateway, the possibility of data sharing on a private blockchain that facilitates patients to manage their health data without any violation of privacy or

Table 1. Existing techniques using blockchain technology in healthcare.

Ref.	Addressed Challenges	Challenges to be solved
[23]	Access control, Data Integrity, Interoperability	Data scalability
[26]	Data Sharing, Data Integrity	Data privacy, Data scalability
[29]	Access control, Interoperability, secure data transfer	Data storage
[30]	Data Integrity, Access Control, Interoperability	Collective decision making
[31]	Data Integrity, Data Security	Data Storage and Scalability
[32]	Interoperability, Access Control	Data Storage and Sharing
[33]	Data Integrity, Global data access	Authentication, Interoperability
[34]	Interoperability, Provenance	Data Storage and Security
[35]	Interoperability	Scalability, Data privacy and security
[36]	Data privacy, Data security	Interoperability, Data scalability

https://doi.org/10.1371/journal.pone.0243043.t001

security. However, this scheme is needed to access data without explicit patient agreement and do not allow any family member to allow data access in emergency situations. Also as the e-health data is growing, scalability is a major issue due to data storage on chain which further leads to the centralization of the blockchain. MedRec [23] is the first functioning prototype in healthcare based on permissionless blockchain implementation utilizes the Ethereum smart contract functionality for the intelligent representation of the medical records which are stored in individual nodes in the network. However, mining mechanisms are required to sustain the distributed ledger; also scalability is considered as another concern with the rise of EHR every second. Other blockchain implementation by Ivan et al [24] is the creation of a blockchain based on Electronic Health Records in which healthcare data is encrypted and stored publicly. Some other techniques has proposed against malicious adversaries [25]. Another blockchain approach in healthcare is by the Medchain, a permissioned network of stakeholders to facilitate healthcare data sharing between hospitals, patients and pharmacies [26]. However, the model storing actual data on chain have significant privacy and scalability issues. A decentralised approach proposed in which the encrypted data is stored off chain and the blockchain layer enforces access control mechanisms by Zyskind et al [27]. The data privacy is a crucial issue with this blockchain technique as the patient's metadata is exposed, which exposes all other information. All the approaches discussed here lack security, privacy and scalability concerns that needs to be addressed [28].

Ancile [29] is another permissionless blockchain structure which utilizes Ethereum based smart contracts that stores hash value of the data references on blockchain for secure, interoperable and efficient access control and employs advanced cryptographic techniques such as proxy re-encryption [37] for the secure transfer of the medical records. Nevertheless, Ancile has technical difficulties such as rewriting of the chain structure [38], exposes frequency of node visits during transactions, inability to store huge data on chain and high storage cost. Ancile and Medrec has scalability issues that resolves by our framework contribution of using IPFS by providing secure data storage in the offchain instead storing on the chain itself. FHIR chain proposed by Zhang et al [30] aims at secure sharing of clinical data by employing the Ethereum blockchain in which the onchain stores only encrypted metadata that serves as a pointer to the original healthrecords, whereas the original medical data is stored in the off chain database. Dubovitskaya et al [39] proposed a permissioned blockchain for secure data sharing focused on oncologic care that leverages local database and cloud services to store encrypted patients' data. However, this approach also makes use of an arbiter for uploading the data in the cloud which makes the system less patient-centric. Another approach proposed by Wang and Song [31] is a secure cloud based EHR system using attribute based encryption and blockchain for the secure sharing of medical data. This approach includes the hospital as an arbiter for encrypting patients' data which again contradicts the decentralized advantage of blockchain technology and makes it less patient-centric.

There are some techniques that used blockchain technology for sharing healthcare information including EMR and PHR but still failed to address data storage and efficient sharing of health data [32]. Another secure cloud blockchain EHR system proposed by wang and song based on attribute based cryptosystem integrating identity-based encryption and digital signatures [31]. Another IoT based blockchain platform was presented for tracking patient vital signs using smart blockchain based smart contracts [33]. Andrea et al. proposed a provenance management platform for tracking electronic healthcare records by employing Hyperledger Fabric blockchain smart contracts [34]. A.Roehrs et al. [35] presented a prototype implementation and evaluation of the OmniPHR architecture that maximizes the replication of health data across computing nodes model by integrating distributed health re-cords using blockchain technology and the open EHR interoperability. Another advanced decentralised privacy

preserving technique was proposed for remote patient monitoring based on Internet of Things (IoT) based technology [36]. Several techniques have also been proposed to achieve computational power by employing neural networks [40–42]. Most of the existing approaches fail to guarantee all the essential requirements such as data privacy, security, secure storage, efficient access control, scalability and interoperability for EHRs. Our research work unravels most of the existing challenges in the e-health environment by employing a permissioned blockchain framework by utilizing Practical Byzantine Fault Tolerance(PBFT) [43] as consensus to enable data sharing in a decentralized fashion via IPFS by maintaining effective patient privacy, confidentiality and integrity for health records.

3 Preliminaries

3.1 Components of healthchain framework

A brief explanation of the preliminary components of our proposed Healthchain framework are outlined as follows:

3.1.1 Membership service provider. Membership Service Provider (MSP) [20] abstracts away all the cryptographic mechanisms such as identity validation, signature generation and verification, protocols behind issuing and validating certificates and user authentication in the healthchain. The default interface for MSP used in this model is Fabric-Certificate Authority (CA) API and there is flexibility for the participating organizations to implement an External CA.

3.1.2 Consensus mechanism. One key property and fundamental layer of blockchain is the consensus mechanism for transactions which depends on smart contracts layer to validate and update transactions in the ledger in accordance with the order they occur. Consensus protocol determines the order of transactions and rejection of bad transaction in the ledger. Practical Byzantine Fault Tolerance(PBFT) [16] is the employed consensus in this framework that utilizes crash fault tolerant or Byzantine Fault tolerant and do not require mining to accomplish consensus.

3.1.3 Hyperledger fabric. Hyperledger Fabric [16] is the first permissioned blockchain platform that features a modular architecture established by IBM under Linux foundation for distributed ledger solutions. This research employs Hyperledger Fabric as the permissioned blockchain framework composed of pre-specified parties for sharing the healthcare information in a reliable way without any central authority. The biggest advantages of this research in developing using Hyperledger fabric is that it uses Byzantine Fault Tolerant consensus protocol [44] that does not entail mining or an associated currency to achieve consensus.

3.1.4 Couch DB. CouchD and LevelDB are the two types of peer databases supported by Hyperledger Fabric. LevelDB is the default state database embedded in the peer nodes and stores chaincode data as simple key-value pairs and supports key, key range, and composite key queries. CouchDB [16] is a JSON format datastore instead of a pure key-value store that allows information mapping of the database documents. CouchDB is the on-chain database used in this research that can also improve compliance security and data protection in the healthchain.

3.1.5 Hyperledger composer. Hyperledger Composer [45] is a set of collaborative tools for the designing and modelling of blockchain business networks that makes it easy and quick to build simple smart contracts and blockchain applications for business owners and developers. Composer, in this research creates a business network definition comprised of model file(.cto) that define the assets, script file(.js) with associated smart contracts, ACL(.acl) for access control rules and permissions and Query(.qry) files for defining queries to query the

state database in the healthchain framework. Moreover, it packages the business network definition to a .bna file for deploying the healthchain business network to a distributed ledger.

3.1.6 SmartContracts- chaincode. Smart contracts are self-executing chain codes that encodes the rules of certain network transactions, and are currently written in Go language that is installed and instanced by authorized participants on channel peers. This research work uses smart contracts that encompass the application logic of the system for EHR transactions particularly for data transmission, access management, request handling such as update medical records, allow doctors write, ereferrals to other doctor, update ownerships, eprescription to pharmacist [46]. Smart contracts will be executed during user interaction to identify request, validate request and for granting access permissions, update permissions for medical records.

3.1.7 Interplanetary file system. IPFS [47] is a peer-to-peer distributed file system that shifts the present version of web to a distributed version and it can be used to replace HTTP. For example; if we want to retrieve a data structure or download a file that is available on the web using IPFS it can be retrieved through the peers in the network using a 'cryptographic hash' or unique fingerprint of that file by using content addressing property of IPFS. IPFS stores the encrypted data in multiple nodes if the data is higher than a defined threshold (size>256KB). In the context of this research, IPFS is used as an off-chain database for the storage of infinite healthcare records in which the medical records are encrypted using public key encryption before storage and hash of the health records will be stored in couch database.

4 Proposed framework

4.1 Overview of proposed framework

The proposed Healthchain architecture is shown in Fig 1. This framework includes Angular 4, Composer Rest Server, Hyperledger Composer, Hyperledger Fabric, Chaincode, CouchDB, IPFS and Fabric Client. Angular 4 is the Front end of the DApp (decentralized application) framework that connects with Composer Rest server which exposes and visualizes the state database, couchDB. The DApp admin interacts with user interface via Angular framework and the application processes user requests to the fabric network through a REST API known as the composer Rest Server. The REST API is used to retrieve the current state of the on-chain database which is the couchDB wherein the Angular framework retrieves the data through GET calls to the composer Rest API. Hyperledger composer builds and models the blockchain business network to create smart contracts for decentralized applications. Hyperledger Fabric [16] is the permissioned blockchain platform for distributed ledger solutions that supports the development of smart contracts known as chaincodes which is writable in Go, Java and Node.js to validate medical data entries by network participants. Healthchain framework employs a two-pronged solution platform (1)on-chain solution implemented on the secure network of Hyperledger Fabric utilizes the on-chain database Couch DB (2)off-chain solution to securely store data via IPFS (Interplanetary File System). Similar to Bitcoin [11] designed to maintain financial transactions, healthchain is intended for transactions in the healthcare that is secured via cryptography. In Healthchain, any interactions with the health records will be recorded as a transaction on the network and the transactions will be visible only to the participants related to the transaction.

Overview of the healthchain is shown in Fig 2. It shows a log of transactions as hash values in the blockchain for every event occurred in the healthcare such as a record creation, access, modification or updation. From Fig 2 it is evident that each transaction has a unique hash that guarantees the integrity of the health records and allows append-only revisions. Moreover, it produces a different hash which will not match the prior hash if the record has been tampered. When the identity management is combined with blockchain applications, the ledger becomes

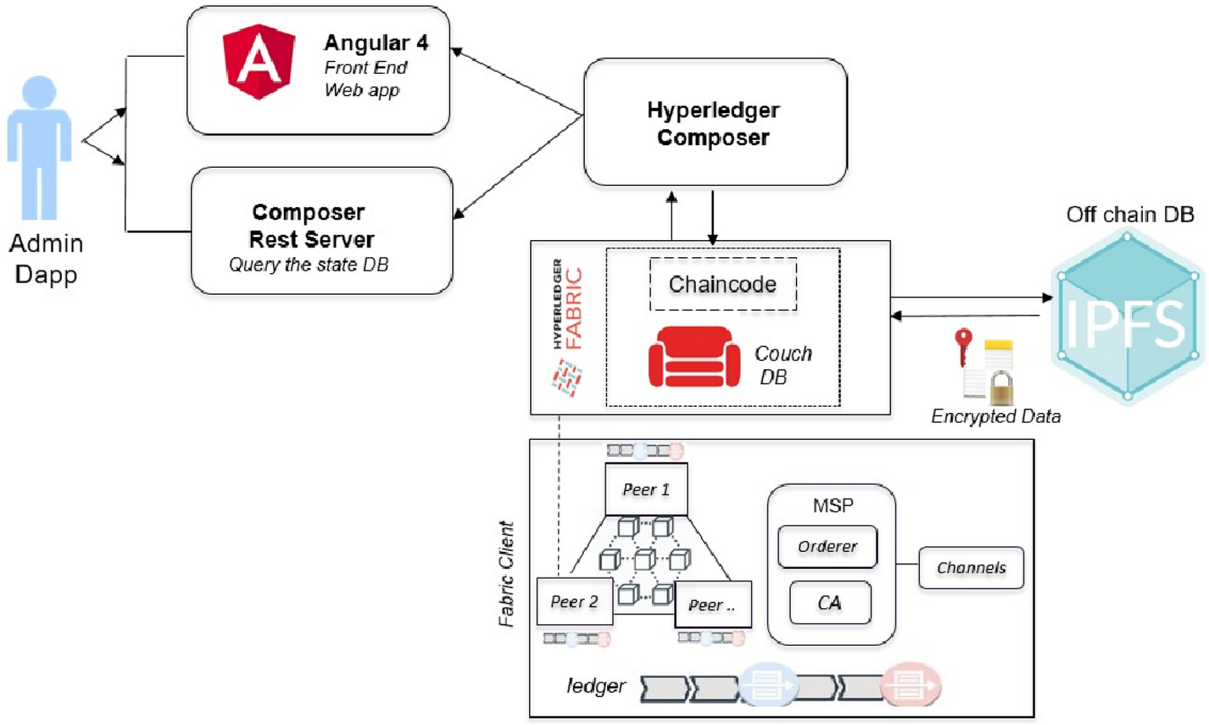


Fig 1. Healthchain architecture.

https://doi.org/10.1371/journal.pone.0243043.g001

the supreme indicator of who did what and when on a blockchain. The working prototype is implemented on a permissioned blockchain called Healthchain on Hyperledger Fabric by employing Hyperledger Composer to create decentralized web applications for a single organization by incorporating three peer nodes as shown in Fig 3. This organization has three peer nodes with one anchor peer node as validating node and an ordering node (Kafka) with a single public channel for registering the network participants. System contains multiple peer nodes configured to use corresponding CouchDB as the world state database and IPFS as the distributed database, a solo ordering node, a Certificate Authority, Membership Service

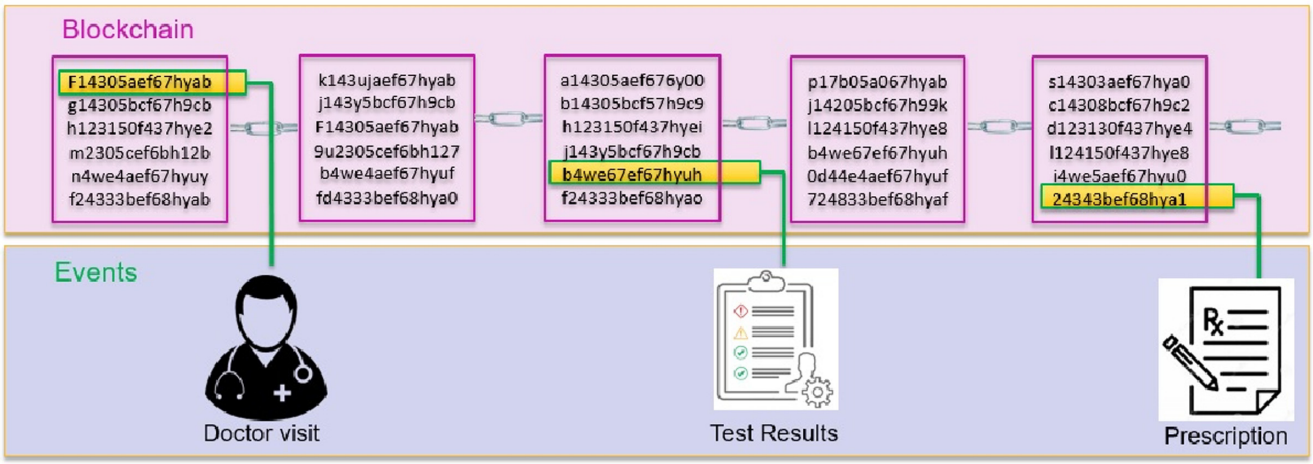


Fig 2. Overview of healthchain.

https://doi.org/10.1371/journal.pone.0243043.g002

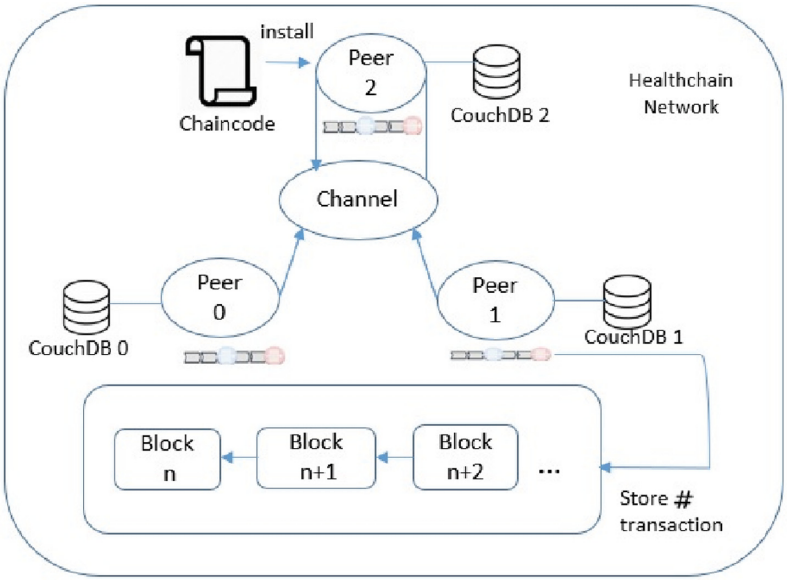


Fig 3. Nodes in healthchain.

https://doi.org/10.1371/journal.pone.0243043.g003

Provider (MSP) and Smart contracts for connecting to the blockchain. This can be extended to multiple peer nodes and multiple organisations in different machines to prove the system scalability. This framework has ledgers and associated smart contracts which has access to the ledgers. The application connects with peer nodes that invokes smart contracts to update the ledger. The Hyperledger Fabric healthchain network is built in a single organisation with three peer nodes using docker containers on the local computer but clearly, in the real world, it would be in separate IP networks or protected cloud environments. The organisation's three peers are labelled as peer0 (P0), peer1(P1) and peer2 (P2) in which each holds their own instance of ledgers and copies of smart contracts. A single channel is designed so that Hyperledger Composer can communicate peers via the channel. In this network, our application A1 generates a transaction T1 to peers peer0, peer1 and peer2 via Channel C. Whenever a transaction executes, the chaincode will be installed to the peers. Application interacts with Peers and invokes chaincodes for querying or modifying the ledger. The transactions are stored within the blocks as hash values in the blockchain enables the history of changes that contributed to the healthchain framework. A block in the ledger pertaining to the health record of a patient i mainly comprises of the workload of that transaction $W_{t(i)}$, hash of the previous transaction $Wp_{\#(i)}$ and hash of the current transaction $W_{\#(i)}$. The total workload of that block can be calculated as $W_{Tot(i)}$:

$$W_{Tot}(i) = W_t(i) + Wp_{\#}(i) + W_{\#}(i) \quad (1)$$

4.2 Cryptographic process in healthchain

Blockchain systems leverage cryptographic techniques to ensure the data integrity and confidentiality. This research employs special public key cryptography for encrypting the data in the off chain storage, IPFS. The wholistic view of the patient-doctor interaction for accessing health records is outlined as shown in Fig 4. The clinician (Doctor) requests permission to access the health record of the patient stored in the IPFS. The patient approves or grants the

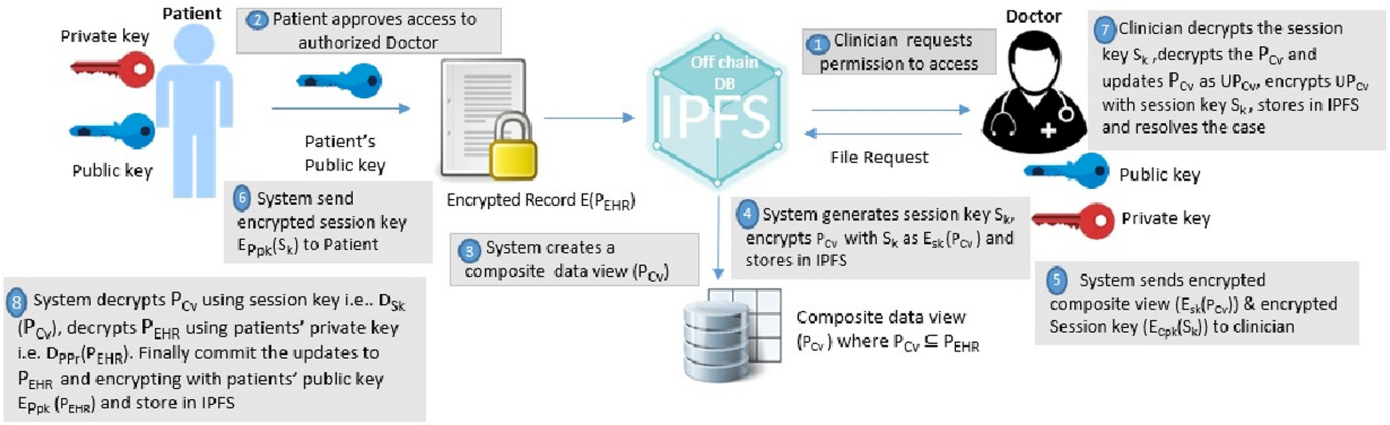


Fig 4. Cryptographic process in healthchain.

https://doi.org/10.1371/journal.pone.0243043.g004

request from permissioned users on the basis of role and rule based access control permissions as shown in Figs 5 and 6. System in this framework refers to client-side application. The system generates a composite view of the record on basis of the request, alternately sharing the whole patient data. The system further generates a session key S_k to access records for a definite

```
<?xml version="1.0"?>
<access-control-rules>
    <role name> Clinician Ci </role_name>
    <permissions desc= "Permissioned Clinician authorized by the Patient">
    <resource desc= "EHRi"> Electronic Health Record </resource>
        <Object>
            <object.ownershipid=Ci.id>
            </Object>
            <action type="read"> ALLOW </action>
            <access mode>
                <access. mode="normal">
                </access mode>
            </permissions>
            <role name> Clinician Ci </role_name>
             <permissions desc= "Permissioned Clinician authorized by the Patient
             and can modify the record for a particular session">
    <resource desc= "EHRi"> Electronic Health Record </resource>
        <Object>
            <object.ownershipid=Ci.id>
            </Object>
            <action type="write"> ALLOW </action>
            <access mode>
                <access. mode="normal">
                </access mode>
            </permissions>
             ......
             <role name> Clinician Ci </role_name>
             <permissions desc= "Permissionless Clinician">
    <resource desc= "EHRi"> Electronic Health Record </resource>
        <Object>
            <object.ownershipid≠Ci.id>
            </Object>
            <action type="read">DENY</action>
            <access mode>
                <access. mode="normal">
                </access mode>
            </permissions>
             ......
             <role name> Pharmacist Phi </role_name>
            <permissions desc= "Permissioned user authorized by the Patient for
            a particular time period">
```

Fig 5. A snippet of the xml document showing access control permission rules.

https://doi.org/10.1371/journal.pone.0243043.g005

```
    <access. mode="normal">
    </access mode>
</permissions>
 ......
 <role name> Pharmacist Phi </role_name>
<permissions desc= "Permissioned user authorized by the Patient for
a particular time period">
<Object>
<object.ownershipid=Phi.id>
</Object>
<action type="read"> ALLOW </action>
<access mode>
    <access. mode="normal OR emergency">
    </access mode>
     </permissions>
     <role name> Referred Clinician Ci </role_name>
<permissions desc= "Permissioned referred clinician authorized by the Patient for
a particular session">
<Object>
<object.ownershipid=Ci.id>
</Object>
<action type="create/update"> ALLOW </action>
<access mode>
    <access. mode="normal OR emergency">
    </access mode>
     </permissions>
      ......
      <role name> Receptionist Ri </role_name>
<permissions desc= "Permissioned user authorized by the Patient for
a particular time period">
<Object>
<object.ownershipid=Ri.id>
</Object>
<action type="view"> ALLOW </action>
<access mode>
    <access. mode="normal OR emergency">
    </access mode>
     </permissions>
      ......
       </access-control-rules>
```

Fig 6. Access control permission rules for healthchain network.

https://doi.org/10.1371/journal.pone.0243043.g006

session and encrypts the composite view with the session key and then stores in IPFS. The system will also send the encrypted session key and encrypted composite view to the clinician. Besides, the system also shares the encrypted session key with the patient. The clinician decrypts the session key, decrypts the composite view and updates the composite view as updated record. Further, the clinician resolves the instance after encrypting the updated record with session key and uploads to the IPFS. The system notifies the record updates to the patient. The system decrypts the updated composite view using the session key, decrypts the encrypted medical record with patient's private key from the IPFS. Finally, the system commit the updates to the original record, encrypts the original record with public key of the patient and upload it to the IPFS. The session key and the composite view for each session expires on session completion. The procedure can be explained with detailed notation in the following algorithms:

4.3 Proposed algorithms

Table 2 depicts the explanation of notations used in the algorithms and Algorithm 1 presents the algorithm to create and update health records by clinician in the healthchain. In our Healthchain framework there are 4 stakeholders in which P denotes Patient, C for Clinician, R for Receptionist and Ph for Pharmacist. We assume there are n participants for each stakeholders in the proposed framework. The Fabric-CA issues public key certificates to all n

Table 2. Explanation of notations.

Notations	Definition
IPFS	InterPlanetary File System
P_{Cv}	Composite data view
S_k	Session Key
C_{Pk}	Public Key of Clinician
C_{Pr}	Private Key of Clinician
P_{EHR}	Patients' Health record
P_{Pk}	Patients' public key
P_{Pr}	Patients' private key
P_i	Patient
C_i	Clinician
R_i	Receptionist
Ph_i	Pharmacist
UP_{Cv}	Updated Composite view
UP_{EHR}	Updated Health Record

https://doi.org/10.1371/journal.pone.0243043.t002

participants such as Patient, Clinician, Receptionist and Pharmacist. There will be a key pair for each participant in which P_{Pk_i} and P_{Pr_i} as the public and private keys of the patient P_i, C_{Pk_i} and C_{Pr_i} as the public and private keys of clinician C_i, R_{Pk_i} and R_{Pr_i} as the public and private keys of the Receptionist R_i and Ph_{Pk_i} and Ph_{Pr_i} as the public and private keys of the Pharmacist Ph_i respectively where i = 1 to n. This scenario gives a detailed explanation of how the Clinician and Patient interacts for accessing health records in the Healthchain framework. The Algorithm 1 is explained as follows. Consider that the patient P_i grants access to his/her medical record P_{EHR_i} to clinician C_i upon request based on access control permissions as shown in Figs 5 and 6. The system then creates a composite view P_{Cv_i} of the patient record P_{EHR_i} that can be accessible to the clinician C_i on request alternately sharing the whole medical record of the patient. Composite view P_{Cv_i} is the attribute set of the stored medical record P_{EHR_i} that the system creates on permissioned user request without sharing the complete patient record. The composite view of a specific health record restricts access to the original data in such a way that a user can see and modify only selected data they need and no more. In other words P_{Cv_i} is a subset of P_{EHR_i} as shown in the Eqs (2) and (3).

Algorithm 1 System(): Create and update composite view of Medical Records

Input: A Clinician C_i with public key C_{Pk_i} and session key S_k to access medical record P_{EHR_i}

Output: Creation and updation of the medical record

1: **for** each user U with access permission to P_{EHR_i}

2: Algorithm checks xml access permission rules to grant or deny access to the user

3: **if** (permission type == "ALLOW" && role Type == 'Clinician')

4: Create composite view P_{Cv_i} of the medical record P_{EHR_i} in IPFS

5: $P_{Cv_i} \rightarrow \int_{i=1}^{n}(D_{P_{Pr_i}}(E_{P_{Pk_i}}(P_{EHR_i})))$

6: $P_{Cv_i} \subseteq P_{EHR_i}$

7: Generate a session key S_k

8: $P_i \leftarrow E_{P_{Pk_i}}(S_k)$ /*Send encrypted session key to patient

9: $C_i \leftarrow E_{C_{Pk_i}}(S_k)$ /*Send encrypted session key to clinician

10: $C_i \leftarrow E_{S_k}(P_{Cv_i})$ /* Send encrypted composite view to clinician

```
11: Algorithm 2() /* Call Algorithm 2() for clinician record access
and update
12: P_EHR_i ← (D_P_Pr_i(E_P_Pk_i(P_EHR_i)))
13: UP_Cv_i ← (D_S_k(E_S_k(UP_Cv_i)))
14: UP_EHR_i ← [(E_P_Pk_i(P_EHR_i)) + (E_P_Pk_i(UP_Cv_i))] /*System commits the update to the
original record
15: return #
16: else
17: access ← deny
18: return access
```

$$P_{Cv_i} \subseteq P_{EHR_i} \tag{2}$$

$$P_{Cv_i} = (D_{P_{Pr_i}}(E_{P_{Pk_i}}(P_{EHR_i}))) \tag{3}$$

The system further generates a session key S_k shared between clinician and the patient for a definite session. The system then sends the encrypted session key S_k to the patient as $E_{P_{pk_i}}(S_k)$ and clinician as $E_{C_{pk_i}}(S_k)$ by encrypting using respective public keys of the patient P_{Pk_i} and clinician C_{Pk_i} for a distinct session as shown in step (8) and step (9) in Algorithm 1. The Composite view P_{Cv_i} will also be encrypted with session key S_k as $E_{S_k}(P_{Cv_i})$ and stores in IPFS. In addition, the system sends encrypted composite view $E_{S_k}(P_{Cv_i})$ to the clinician. Here the Algorithm 1 calls Algorithm 2 for clinician update of health records. Now, Clinician decrypts the session key with his private key and decrypts the composite view with the session key as shown in step (2) and step (3) in Algorithm 2. If there are any updates, clinician updates P_{Cv_i} as UP_{Cv_i}, resolves the case, encrypts with the session key and uploads UP_{Cv_i} to IPFS as $E_{S_k}(UP_{Cv_i})$. System refers to the client-side application in this framework. The patient uses a pass code to encrypt the private key P_{Pr_i} and stores it on the client side. Every time for convenience, the patient can provide this pass code that decrypts the private key instead of sharing or uploading the private key, and the client end application can use this private key to decrypt the medical record. On clinician's record update calls Algorithm(1) in which the system decrypts the encrypted record ie. $E_{P_{Pk_i}}(P_{EHR_i})$ using patient's private key and decrypts the encrypted updated composite view from the IPFS ie. $E_{S_k}(UP_{Cv_i})$ using the session key as shown in steps (12) and step (13) in Algorithm 1. Finally, the patient commits the updates to the original record and encrypts the original record P_{EHR_i} as $E_{P_{Pk_i}}(P_{EHR_i})$ before uploading to IPFS as shown in Eq (4).

$$UP_{EHR_i} = [(E_{P_{Pk_i}}(P_{EHR_i})) + (E_{P_{Pk_i}}(U_{P_{Cv_i}}))] \tag{4}$$

The session key S_k and the composite view P_{Cv_i} for each session expires on session completion. The transactions eventuated on clinician access and record updates that invoke smart contracts thus creates a unique hash value and added to the healthchain. This composed of two main algorithms as summarized in Algorithm 1 and Algorithm 2.

Algorithm 2 System(): Algorithm for clinician creating and updating medical records in Healthchain

```
Input: A Clinician C_i with public key C_Pk_i and session key S_k to create
medical record P_EHR_i
Output: Record Creation and updation
1: for each clinician with access permission on receiving encrypted S_k
and P_Cv_i
```

2: $C_i \leftarrow D_{CPr_i}(S_k)$ /*Decrypt session key with Clinician's private key
3: $C_i \leftarrow D_{S_k}(P_{Cv_i})$ /*Decrypt composite view with clinician's session key
4: $P_{Cv_i} \rightarrow (UP_{Cv_i})$ /* Clinician updates Composite view
5: IPFS $\leftarrow E_{S_k}(UP_{Cv_i})$ /* Encrypts updated composite view with Clinician's session key
6: **System()** /*call System()
7: **End**

4.3.1 Access control permission rules. Figs 5 and 6 shows a snippet of the xml structure of access control permission rules in Healthchain network. The Algorithm 1 checks the access management rules in Figs 5 and 6 for granting or denying access to the health records. Access control policies are defined to safeguard the privacy of patients' healthcare records [48]. Algorithm 2 renders an algorithm for clinician creating and updating health records in the Healthchain network. When an access request is made, the algorithm verifies the access control rules that are written in extensible markup language in Figs 5 and 6 which defines the access rights of the user on resource EHR_i defined by the owner. This access rules will be stored in the blockchain and submitted to the blockchain channel through a transaction called Business Network archive Transaction. In this approach;

- the rules comprises of the condition specifying the ID of the subject to which the access control policy grants the right of access;
- conditions specifying sets of values authorized for the subject, resource, action type and environment attributes for the access to be granted.

In our framework, we designed the rules to modify these conditions properly when they transfer those access rights to other authorised users before submitting to the healthchain. The actors of this scenario is resource owner P, Resource EHR_i and several subjects such as C_i, Ph_i and R_i in the healthchain framework. The clinician C_i or any user can only read, write, modify or update access to health records only according to the access control permissions. From the Fig 5 it is clear that if the subject id matches with the object ownership id and only if the subject is a permissioned stakeholder, permissions such as read, write access are allowed or otherwise access will be denied. The stakeholders such as Pharmacist and Receptionist in this healthchain framework has given read access only to composite view of the health records for a particular session if their subject id matches with the ownership id of the object or resource as shown in Fig 6.

5 Prototype implementation of proposed framework

This section gives a detailed description of how users and records are added in the healthchain framework; steps included to provide access permissions to authorized users; retrieval of records in the healthchain framework.

5.1 Adding users to the healthchain framework

The process for adding users to the healthchain network can be seen in Fig 7. The framework developed is role-based in which Patients, Clinicians(Doctors), Chemists and Receptionists can register themselves and login using login credentials such as email address and password. The nodes will be added by the network admin to the blockchain after validation from the consensus voter nodes. The patients' and the users' will be added to the healthchain with limited validation using their credentials such as username and password with each user having public private key pairs Pk_i, Pr_i. The user password is encrypted using SHA-256 hashing algorithm for improved security. The Composer Rest Server generates a REST API from the deployed

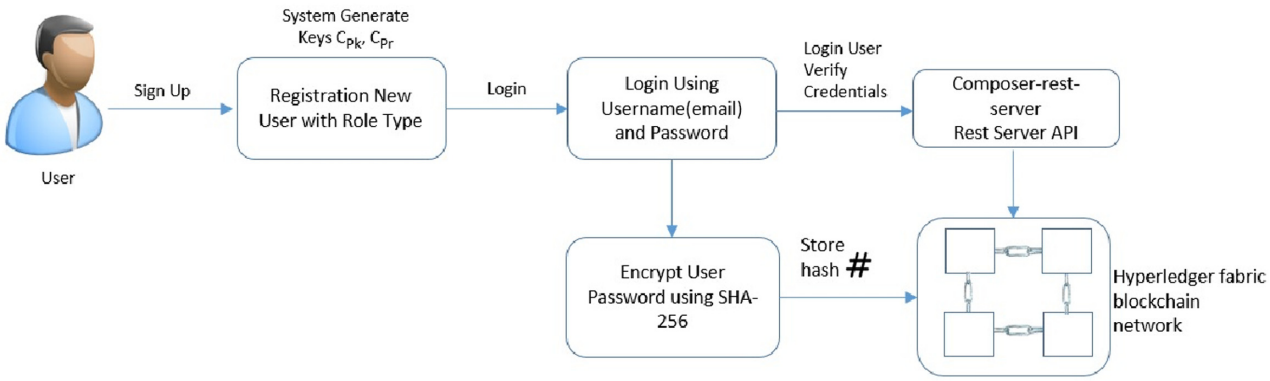


Fig 7. Adding users to healthchain.

https://doi.org/10.1371/journal.pone.0243043.g007

blockchain business network that visualize and queries the values stored in couch database. The rest server also performs create, read, update and delete operations for assets and participants which allows transactions for processing and retrieval.

5.2 Adding records to the healthchain framework

Fig 8 shows the stage by stage process of how the Clinican adds medical record of the patient to Healthchain. This approach begins with assuming that the patient and the clinician have established an authorized relationship for updating health records. The process of adding medical records by clinician to the database is employed via internal encryption mechanism. There are two scenarios of adding patient records to the healthchain. (a)A new patient record will be created by the clinician to the healthchain through uploading the encrypted medical record using the patients' public key to the IPFS. (b)A new patient record will be added or modified by the clinician; the system creates a composite view, P_{Cv_i} of the data that can be accessible to the clinician C_i alternately sharing the whole data. The system further generates a session key S_k shared by patient and the clinician for a distinct session. The system then sends the encrypted session key S_k to the patient as $E_{P_{pk_i}}(S_k)$ and clinician as $E_{C_{pk_i}}(S_k)$ by encrypting using respective public keys of the patient P_{Pk_i} and clinician C_{Pk_i} for a distinct session. The Composite view P_{Cv_i} will also be encrypted with session key S_k as $E_{S_k}(P_{Cv_i})$ and stores in IPFS. In addition, the system sends encrypted Composite view i.e.$E_{S_k}(P_{Cv_i})$ to the clinician. Now, Clinician decrypts the session key with his private key and decrypts the composite view with the session key. If there are any updates, clinician updates P_{Cv_i} as UP_{Cv_i}, resolves the case, encrypts with the session key and uploads UP_{Cv_i} to IPFS as $E_{S_k}(UP_{Cv_i})$. On clinicians' record update, the system decrypts the encrypted record ie. $E_{P_{Pk_i}}(P_{EHR_i})$ using patients' private key and also decrypts the encrypted updated composite view from the IPFS ie. $E_{S_k}(UP_{Cv_i})$ using the session key. Finally, the patient commits the updates to the original record and encrypts the original record P_{EHR_i} as $E_{P_{Pk_i}}(P_{EHR_i})$ before uploading to IPFS. The session key S_k for each session expires and the composite view P_{Cv_i} will be deleted after the session is completed. The transactions eventuated on clinician access and record updates will be hashed by employing smart contracts and added to the healthchain. This procedure can be summarized by two main algorithms as shown in Algorithm 1 and Algorithm 2 by employing Figs 5 and 6 for access management.

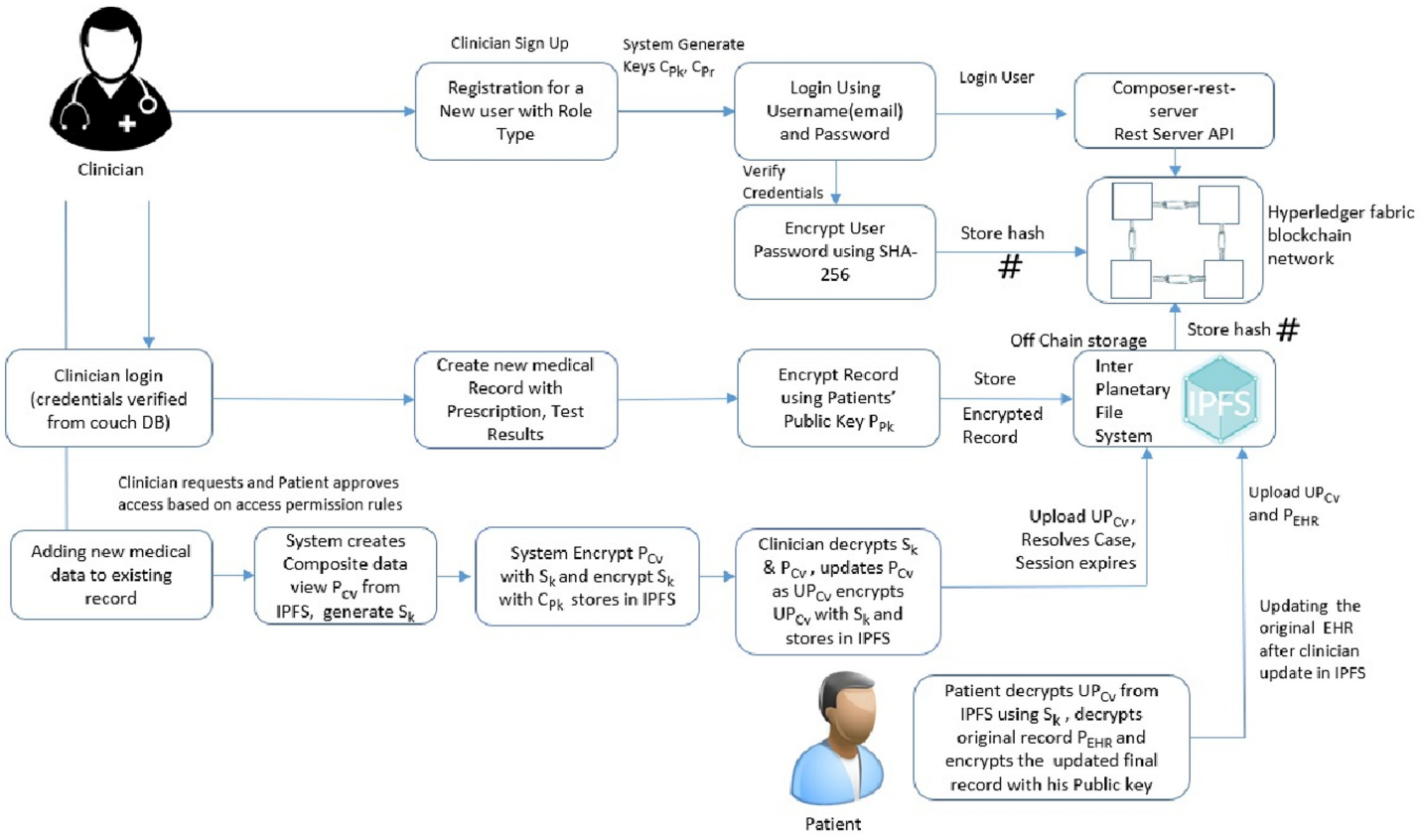


Fig 8. Adding records to healthchain.

https://doi.org/10.1371/journal.pone.0243043.g008

5.3 Providing access permissions to authorized users

The patient has complete control and ownership to grant read, write, deny or revoke access permissions to the provider or other stakeholders such as receptionist, doctor or a pharmacist on the medical record thereby maintaining restrictive access control. Fig 9 describes the block diagram to provide access permission in the framework. The xml rules shown in Figs 5 and 6 presents read, write and deny access permission rules in the proposed healthchain. Moreover, the patient can permit access to health records based on authenticated user approved by the consensus in accordance with role type and permission type. Furthermore, the patient can also revoke the access from a particular clinician on his medical record and in that situation, the permission to the record can be denied from further access. As shown in Fig 9, Healthchain uses permission rules based on Role based and Rule based access control mechanisms for refined and restricted access to medical records. Smart contracts written will be executed during user interaction to identify request, validate request, updating records and granting access permissions for medical records.

5.4 Retrieval of records

Retrieving a medical record can be performed through a series of transactions. The process begins with patient who uploads his data in IPFS via public key encryption. The clinician or a stakeholder who has access to the record for a particular session from IPFS, the system automatically generates a composite data view P_{Cv} which requires encryption with session key S_k.

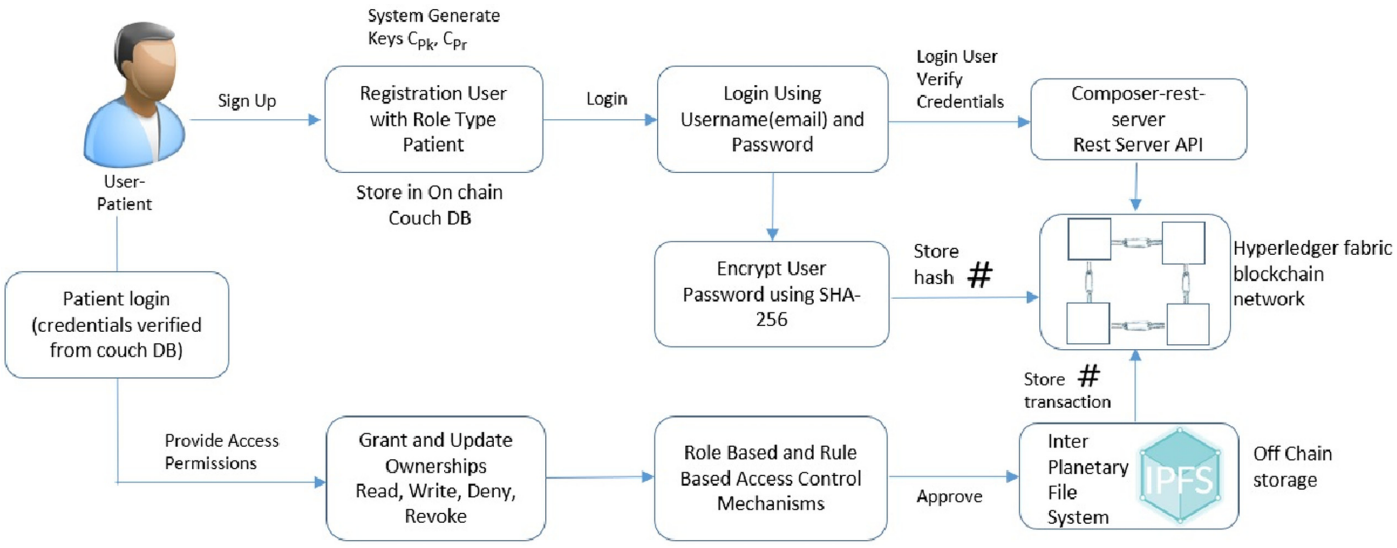


Fig 9. Providing access permission.

https://doi.org/10.1371/journal.pone.0243043.g009

Additionally the session key will be encrypted with the clinician's public key C_{pk} for secure transfer. The clinician updates the medical record on arrival and encrypts with the session key before storing in the IPFS. The system notifies the patient regarding the updates on the medical record that decrypts the updated medical record UP_{Cv_i} with the shared session key. The patient further encrypts the updated record with the patients' public key, commits the updates to the original record and uploads to IPFS. Furthermore, the patient can decrypt his record using his private key from IPFS and upload the encrypted record using patients' public key. All the transactions occured will be hashed by utilizing smart contracts and added to the healthchain. The step by step explanation is as shown in Fig 10.

6 Prototype implementation and results

For the implementation of our proposed Healthchain framework, we initially employed a private Hyperledger fabric blockchain viz healthchain in a Linux environment. Smart contracts are deployed for every transaction in the healthchain, IPFS storage system is utilized and network entities developed to build the healthchain framework. Following are the main components used for the simulation environment and Table 3 presents the machine configurations.

The prototype is a user-centric model to process healthcare records using blockchain network, assuring the data ownership of individuals by preserving data security, privacy, data scalability and data integrity. This prototype is designed with few stakeholders namely Doctor (Clinician), patient, receptionist and pharmacist that builts a private healthchain framework. The framework's flow is detailed as given below:

- Similar to a web application, URL of the framework is visible to users irrespective of the blockchain technology used at the rear end.
- The framework allows the user to signup with vital details like unique id, username, email address and password and the values will be stored in the onchain database, couchDB.
- The user can successfully log in if the username and password matches with the data stored in couch DB by querying the blockchain.

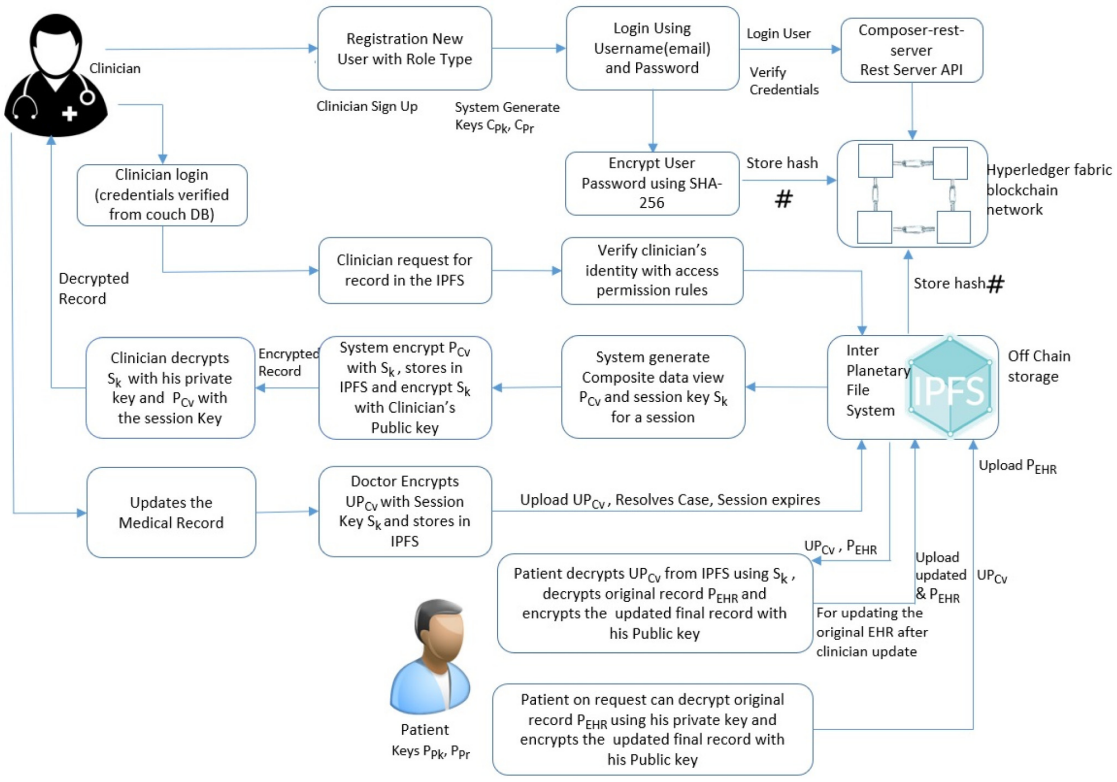


Fig 10. Retrieval of health records.

https://doi.org/10.1371/journal.pone.0243043.g010

- A doctor logged in can upload the medical records to the IPFS by encrypting with the users' public key thereby using public key encryption. The hash value generated by IPFS will be maintained in the couchDB, onchain database of the blockchain and thus preserves data integrity.

Table 3. Development environment for the proposed framework.

Component	Description
Operating Systems	Ubuntu Linux 16.04 64 bit
IDE	Hyperledger Composer
CPU	(Intel(R)Core(TM)i5-8500 CPU @ 2.5GHz 2.7GHz
Memory	8 GB
Node	v8.15.0
CLI Tool	Composer REST Server
Docker-compose	Version 18.09.2
Python	v2.7.12
Blockchain Network	Hyperledger Fabric
Framework Tools	Visual studio code
Programming Language	Angular 4,Node.js,composer modeling language
On-chain Database	CouchDB
Off-Chain Database	IPFS

https://doi.org/10.1371/journal.pone.0243043.t003

- A patient who is logged in, will be able to grant and deny accesses such as read, write, update permissions to the stakeholders on their medical records thus maintained restrictive access control.

The illustration of EHR access in Healthchain is presented in Figs 11–14 and 15. Fig 11(a) shows Rest API that exposes the CouchDB, state database of the blockchain. The data can be queried from the onchain state database via the Rest API as shown in Fig 11(b). Fig 11(c) is the User Sign Up in which the Patients, Doctors, Chemists and Receptionists can register in the healthchain using their roles.

After registration, the user can login with their email address and password by choosing their user type as shown in Fig 11(d). According to the role type Patient, the patient can view his profile, book an appointment for the doctor, view the medical records and add ownership to the doctor on his medical records as presented in Fig 11(e). The patient can book his appointment via Receptionist and the Receptionist can update the participant using patient id by accepting or rejecting the appointment as shown in Fig 12(a). After the approval of appointment by receptionist, the patient can consult the doctor and the doctor can create medical record for the patient. The clinical notes or the diagnosis results can be uploaded to IPFS using public key encryption for a session and IPFS returns the hash of encrypted record which is stored in the couch DB i.e. blockchain as illustrated in Fig 12(b). Being a patient centric blockchain, patients can also provide access permissions such as read, write and in certain situations where in the patient wants to revoke access to a doctor on his medical records, permission to the record can be denied as seen in Fig 12(c). Moreover, the patient can view the medical records added by the doctor as a data provenance [49] shown in Fig 13. Also, healthchain contributes secure e-referrals between doctors and doctor to Pharmacist interactions for drug tracking transaction via smart contracts in this research as shown in Figs 14 and 15. Fig 14 shows the process in which the Doctor creates patient referral records by using unique attributes and the corresponding view of referral records in the Healthchain. Fig 15 shows the representation of creating prescription in Doctor's profile and resultant prescription view in healthchain which offers a promising solution to drug tracking that not only makes prescriptions safer but also guarantees a reliable transaction history of medical records.

7 Analysis of the framework

To validate the functional capability and to evaluate the performance of the prototype, some test cases have been explored. Four case studies are investigated to assess performance of the Healthchain framework systems which are illustrated in terms of efficiency, storage, security and scalability.

- Case I: Efficient storage of Health Records
- Case II: High Degree of Security
- Case III: Enhanced data privacy
- Case IV: Improved data scalability

7.1 Efficient storage of health records- case I

Efficient storage of health records in Interplanetary file system has been tested against a few cases listed in Fig 16. The first test case verifies if a doctor can upload health records or diagnosed test results on IPFS. The implementation results shown in Fig 12(c) shows that the authenticated doctor can have write access for the medical records and upload encrypted

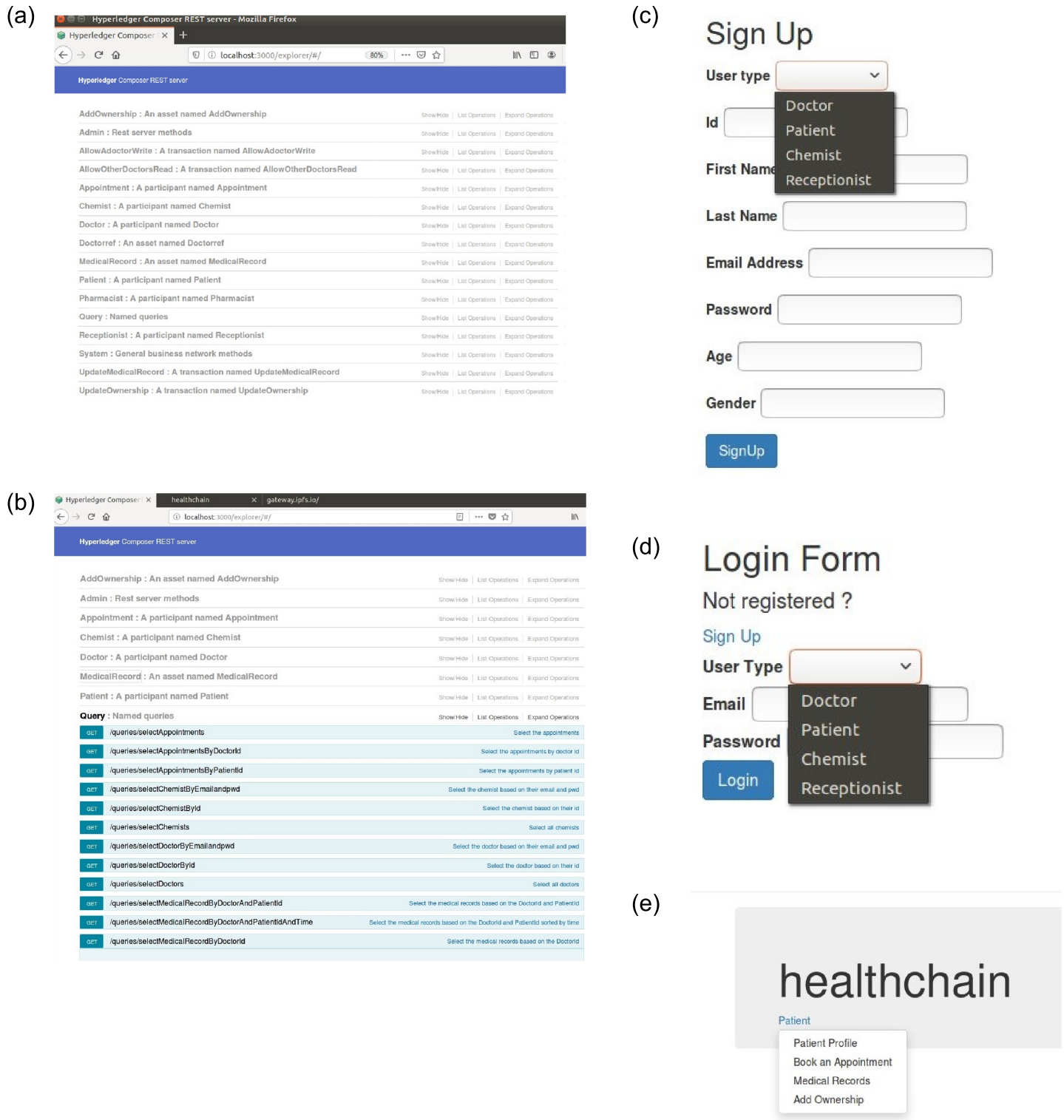


Fig 11. Illustration of EHR access in healthchain.

https://doi.org/10.1371/journal.pone.0243043.g011

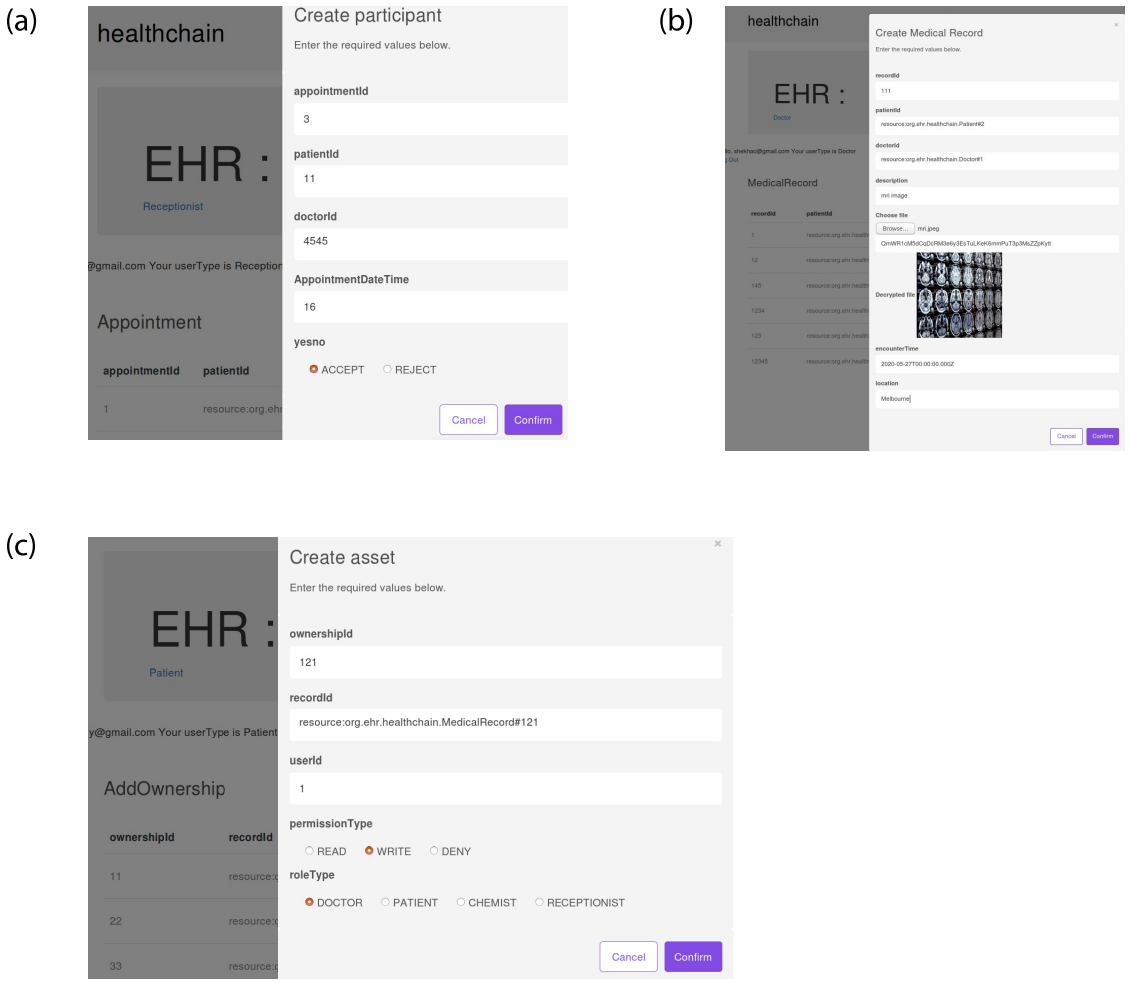


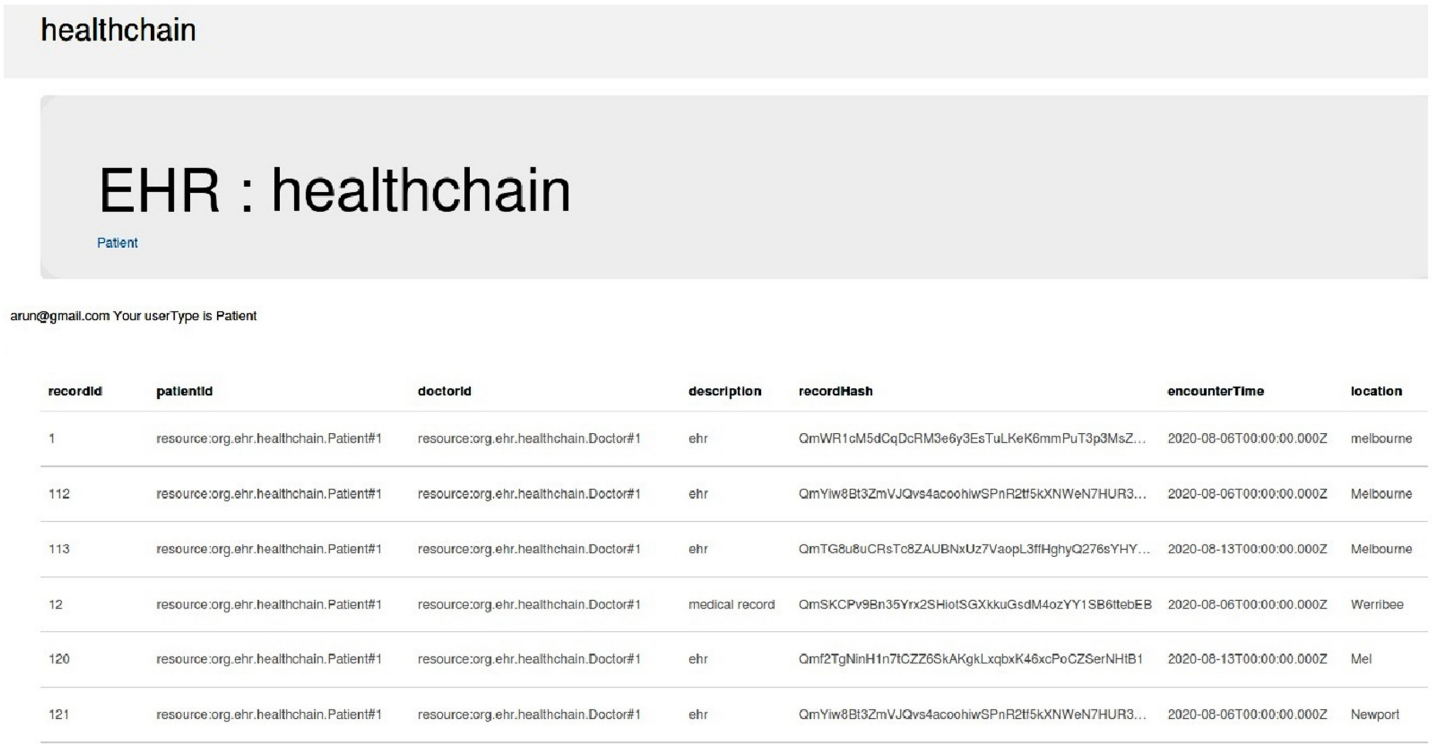
Fig 12. Illustration of EHR access in healthchain.

https://doi.org/10.1371/journal.pone.0243043.g012

records into IPFS. A public key encryption algorithm has been used for encrypting the medical records on to the decentralized storage IPFS. The second case is tested if a doctor has read access permission to the medical records and is successfully verified as the doctor has been authenticated by the patient. Furthermore, it tests that a patient can view the medical records and Fig 13 portrays the provenance history of the medical records. Moreover, the system is tested against whether a record can be uniquely identified or not and has been successful as every medical record is uniquely related with a doctor id and patient id. Additionally the system has been checked against if an encrypted record can be effectively retrieved after decryption and has been successful as shown in Fig 12(b). The outcome is successful as the updated record can be encrypted with doctor's session key for storing in IPFS and updated record can be decrypted by using patients' session key at the patient side.

7.2 High degree of security- case II

Degree of security in healthchain has been verified against a few test cases as shown in Fig 17. The first case is tested and successful as the users' password is encrypted before storing user authentication information in to the couch database. The second test case verifies the degree of security to check whether the medical records are encrypted on the IPFS and returns a unique



recordId	patientId	doctorId	description	recordHash	encounterTime	location
1	resource:org.ehr.healthchain.Patient#1	resource:org.ehr.healthchain.Doctor#1	ehr	QmWR1cM5dCqDcRM3e6y3EsTuLKeK6mmPuT3p3MsZ...	2020-08-06T00:00:00.000Z	melbourne
112	resource:org.ehr.healthchain.Patient#1	resource:org.ehr.healthchain.Doctor#1	ehr	QmYiw8Bt3ZmVJQvs4acoohiwSPnR2tf5kXNWeN7HUR3...	2020-08-06T00:00:00.000Z	Melbourne
113	resource:org.ehr.healthchain.Patient#1	resource:org.ehr.healthchain.Doctor#1	ehr	QmTG8u8uCRsTc8ZAUBNxUz7VaopL3ffHghyQ276sYHY...	2020-08-13T00:00:00.000Z	Melbourne
12	resource:org.ehr.healthchain.Patient#1	resource:org.ehr.healthchain.Doctor#1	medical record	QmSKCPv9Bn35Yrx2SHiotSGXkkuGsdM4ozYY1SB6ttebEB	2020-08-06T00:00:00.000Z	Werribee
120	resource:org.ehr.healthchain.Patient#1	resource:org.ehr.healthchain.Doctor#1	ehr	Qmf2TgNinH1n7tCZZ6SkAKgkLxqbxK46xcPoCZSerNHtB1	2020-08-13T00:00:00.000Z	Mel
121	resource:org.ehr.healthchain.Patient#1	resource:org.ehr.healthchain.Doctor#1	ehr	QmYiw8Bt3ZmVJQvs4acoohiwSPnR2tf5kXNWeN7HUR3...	2020-08-06T00:00:00.000Z	Newport

Fig 13. Illustration of provenance in healthchain.

https://doi.org/10.1371/journal.pone.0243043.g013

hash for the encrypted record as shown in Fig 12(b). The outcome is favorable as the medical records are encrypted using the patients' public key before uploaded into the IPFS. Furthermore, the prototype has also been verified with the usage of public key infrastructure and found successful since public and private keys are used for user identification. The prototype

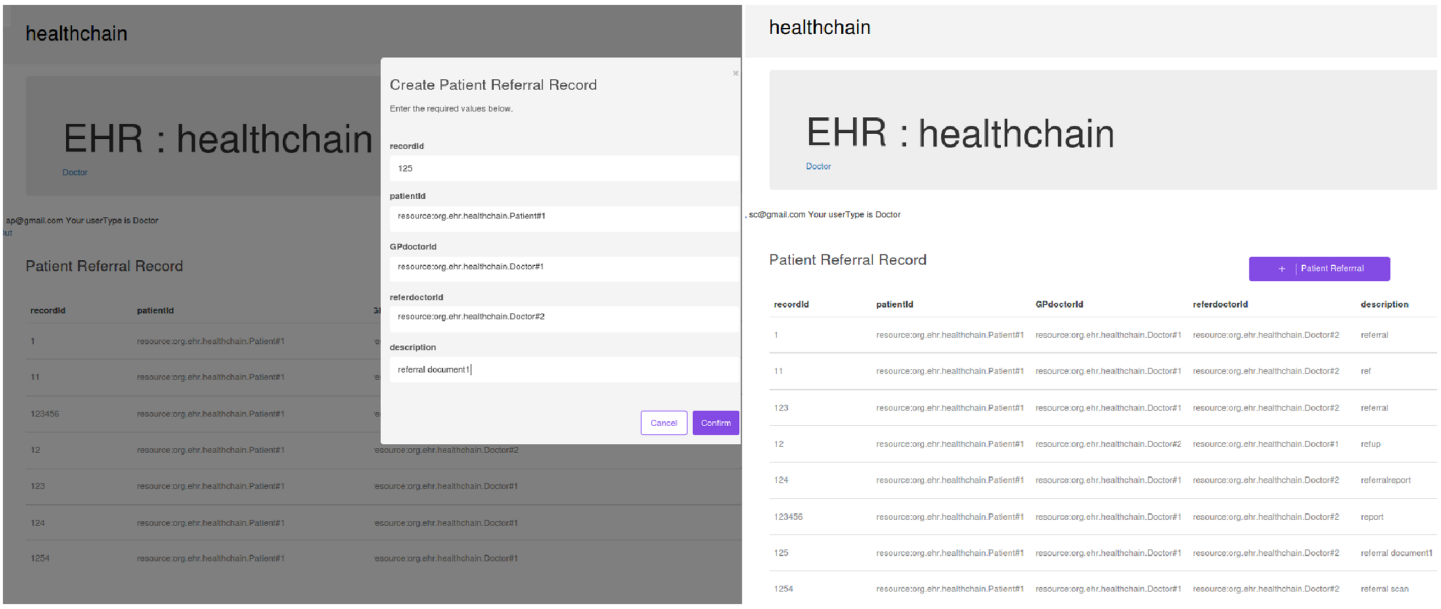


Fig 14. Illustration of referrals in healthchain.

https://doi.org/10.1371/journal.pone.0243043.g014

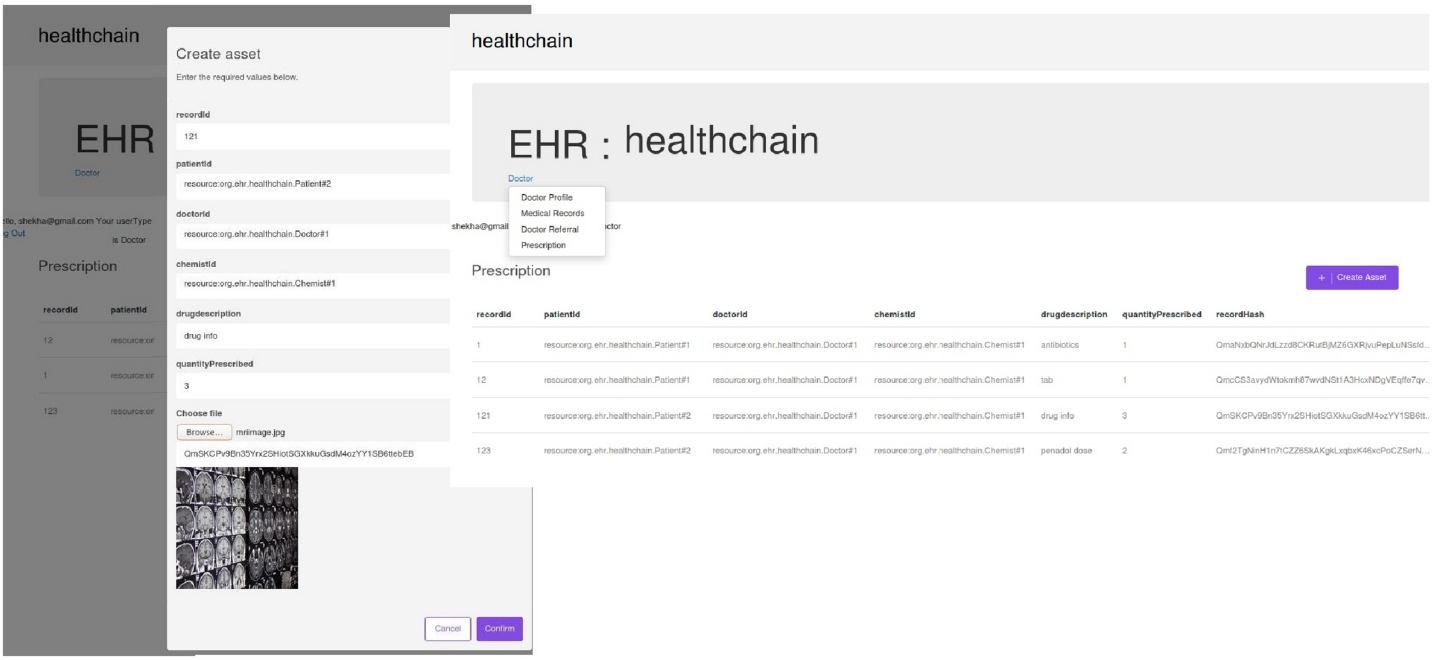


Fig 15. Illustration of creating prescription in healthchain.

https://doi.org/10.1371/journal.pone.0243043.g015

has also been tested to check whether the session has been maintained and found successful as long as the user has not signed out from the application and the session is not expired.

7.3 Enhanced data privacy- case III

HeathChain employs several privacy preserving mechanisms. The Data Privacy in healthchain is determined based on the permission to access the healthcare records. The access control for the medical records are tested against a few test cases as listed in Fig 18. The initial case is verified and successful as the users can view the homepage based on their user type as shown in Fig 11(e). Additionally the system has been tested to check whether a patient can provide grant or revoke access of the health records to the stakeholders and has been successful that preserves

S.No	Test case	Description	Outcome
1	Verify if a doctor can upload medical records on IPFS	The doctor authenticated by the patient can have write access for the record and upload encrypted records into IPFS	Passed
2	Verify if a doctor can view the medical records on permission	The doctor authenticated by the patient can have read access for the record	Passed
3	Verify if a patient can view the medical records	The patient can see the provenance history of their records	Passed
4	Verify if a record is uniquely identified	Each medical record is uniquely associated with a patient id and doctor id	Passed
5	Verify if an encrypted record can be effectively retrieved	The updated medical record can be encrypted with doctor's session key for storing in IPFS and updated record can be decrypted by using patients' session key at the patient side	Passed

Fig 16. Storage of health records.

https://doi.org/10.1371/journal.pone.0243043.g016

S.No	Test case	Description	Outcome
1	Verify if the user's password is encrypted in healthchain	To maintain security, user's password can be encrypted before storing in the state database	Passed
2	Verify if the medical records are encrypted on IPFS	The medical record is encrypted using the patients' public key and uploaded into IPFS and returns a hash value for the encrypted record	Passed
3	Verify if Public Key Infrastructure is used	Two keys public key and private key have been used for user identification	Passed
4	Verify if session is maintained for the user	If the user has not signed out of the application and if session is not expired, the application session will be maintained	Passed
5	Verify if the rest API being used is secured	The state database, couch DB of the healthchain has exposed a rest api that also need to be secured	Can be added in future work

Fig 17. Degree of security.

https://doi.org/10.1371/journal.pone.0243043.g017

the data privacy. Furthermore, the system is also tested to see whether the patient can provide access permissions to the stakeholders. From the simulation results, it can be seen that patients can also provide access permissions such as read, write, and in certain situations where in the patient wants to revoke access of a doctor on his medical records, permission to the record can be denied as shown in Fig 12(c).

7.4 Improved data scalability- case IV

Healthchain is well-founded on various notions to promote scalability. This research further contributes to data scalability by storing the hash value of medical records on chain and encrypted data off chain, in the decentralized storage, IPFS. Scalability of data has been examined against a few test cases as shown in Fig 19. A record of 100 MB was uploaded at a time to IPFS and has been successful which determined the scalability of the system. Considering the machine configuration, the system also verified that the average time taken by multiple users for the uploading and retrieval of the record was less than 60 seconds. A detailed view is portrayed in Figs 27 and 28. Therefore, it can be concluded that the system is able to handle a large dataset at low latency.

S.No	Test case	Description	Outcome
1	Verify if a user can view the homepage based on user type in healthchain	When the url is exposed to the user, homepage can be seen with user type for an existing user	Passed
2	Verify if a patient can grant or revoke access of the medical record	The patient has all the permissions to grant or revoke access from other user types to maintain privacy	Passed
3	Verify if a patient can provide access permissions	The patient has the permission to provide read, write and deny access permission according to the role type	Passed
4	Verify if a doctor can view the medical record using security token	Since the user can revoke access of their medical record from the doctor, providing access again is troublesome and hence some session token has been added into the framework that expires after the session	Passed

Fig 18. Enhanced data privacy: Access control.

https://doi.org/10.1371/journal.pone.0243043.g018

S.No	Test case	Description	Outcome
1	Verify if a huge file can be stored on IPFS	Upload a medical record with size > 100 MB	Passed
2	Verify if a small file can be stored on IPFS	Upload a medical record with size < 10 MB	Passed
3	Verify if the time taken to store and retrieve the medical record is acceptable	The time taken to upload the medical record is few milliseconds	Passed
4	Verify if files or medical records with different extensions can be uploaded in IPFS	Files with different extensions such as video, audio files can be added in a later stage of this research work	Can be added in future work

Fig 19. Improved data scalability.

https://doi.org/10.1371/journal.pone.0243043.g019

7.5 Comparative analysis of proposed framework with existing blockchain techniques

This section performs a comparative analysis of the proposed framework with the existing blockchain techniques in terms of major privacy preserving requirements viz Data Integrity, Data privacy, Data security, Confidentiality and Scalability. The proposed framework is compared against the existing blockchain based implementations such as [26, 31, 32] and [36]. From Table 4, it is evident that proposed system satisfies the shortcomings of existing systems in terms of data security, privacy and scalability.

This section also describes how the proposed framework satisfies the privacy preserving requirements.

Data integrity. Data is immutable and tamper proof as the data is stored as hash values in each block and each block stores the hash value of the previous block in this blockchain framework. The trust on this blockchain framework is based on the consensus, digital signature and the designed cryptographic algorithm despite relying on a third party provider. Since all the blocks are linked, any alteration in the original data will result in a change in its hash value and it is computationally difficult to tamper the ledger, such that the non-tampering of the medical record is also explicitly guaranteed. In addition, the original data is stored in IPFS after performing a special cryptographic encryption technique and IPFS stores the data in multiple nodes if the size of data is higher than a defined threshold.

Data privacy. This framework provides a paramount significance to health record data privacy and Patient Privacy. Besides special encryption mechanisms that ensures data security, access control permission rules has been implemented in the system to safeguard the data privacy of patient health records. The framework ensures fine grained access control by integrating Role, Rule and Attribute based access control permission rules for any data request. Secondly, unauthenticated data requester cannot access data location since the blockchain

Table 4. Comparative analysis.

Scheme	Data Integrity	Data Privacy	Data Security	Confidentiality	Scalability
MedChain [26]	✓	✗	✓	✓	✗
Wang & Song [31]	✓	✗	✓	✓	✗
Blochie [32]	✓	✓	✗	✓	✗
Blockchain for IoT [36]	✓	✓	✓	✓	✗
Proposed Framework (Healthchain)	✓	✓	✓	✓	✓

https://doi.org/10.1371/journal.pone.0243043.t004

stores only hash value of the encrypted medical record. Thirdly, if the data requester attributes do not meet the access policy embedded in the network archive file, it is also impossible to acquire any real medical record data from the blockchain public information.

Data security. Data Security is a crucial feauture as the EHR is cryptographically stored and dealt in the proposed system. This blockchain framework stores only hash of the encrypted data in the on chain and actual huge data is stored after encryption in the offchain storage. Since the framework is a patient centric approach that provides authenticated access permissioned by the patient guarantees the security of the health records. Also the smart contracts functionality combines with blockchain solutions embraces high-level encryption and ensures patient confidentiality in their health care information. In addition, the data stored on IPFS is encrypted using a special cryptographic algorithm to establish robust blockchain data solutions.

Confidentiality. In this framework, every health record of the patient will be stored in the IPFS after encrypting with patients' public key and allows only the permissioned or authenticated users to access the record for a particular session. Since the framework is a patient centric approach in which the patient has complete control, unless for emergency situations to provide access permissions to the stakeholders, the confidential nature of health data is preserved.

Scalability. The proposed scheme preserves most of the privacy requirements and provides cryptographic storage of healthrecords in IPFS thereby resolves the scalability issue in the existing techniques. The scalability of the proposed system has demonstrated and proved that the system is capable of processing large datasets at low latency as shown in Figs 27 and 28.

7.6 Performance analysis and discussion

The evaluation matrix for the framework is shown in Table 5 that specifies the stakeholders, functions and solved problems to facilitate privacy preservation requirements. The framework is user-oriented that handles efficient storage and transfer of medical records ensuring the data ownership of the individuals, patient confidentiality and data integrity. By adopting access control mechanisms, clients can manage their own private information without jeopardizing confidentiality. Meanwhile, each requisition and update from the stakeholders viz receptionist, doctor, patient and pharmacist are reflected in the couchDB, state database of the healthchain. The patients' can handle the access control mechanism by granting or revoking access of the medical records to the stakeholders thereby, maintaining user and data privacy. Data security and patient confidentiality is attained by data storage using public key encryption that secures the user data.

Several experiments have been carried out to analyse and evaluate the performance of the proposed blockchain system. The assets defined here are: (a)Medical Record (b)Referrals (c)Prescription (d)Add Ownership. Transactions are: (a)Create Medical Records (b)Update Medical Records (c)Allow Doctors Write (d)Update Ownership (e)eReferrals to other

Table 5. Evaluation matrix.

Stakeholders	Functions	Solved Problems
Patients	Provide Access Control; User Login; Encrypted Data Storage; Decentralisation, Data Provenance; Data Retrieval	Data Confidentiality; Authentication; Privacy; Data Scalability; Authorization; Non-repudiation; Data Integrity
Doctors	User Login; Data Storage with Encryption; Secure e-Referral; Data Retrieval; Prescription Management	Authentication; Confidentiality; Scalability; Non- repudiation; Integrity; Security
Pharmacists	User Login; Prescription Management	Authentication; Non- repudiation; Integrity
Receptionists	User Login; Appointments Management	Authentication; Confidentiality

https://doi.org/10.1371/journal.pone.0243043.t005

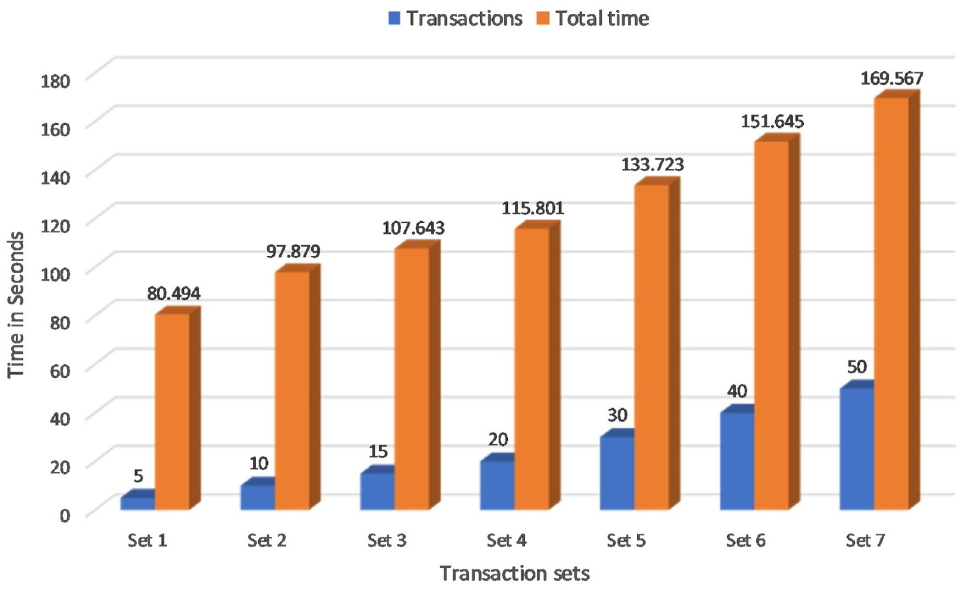


Fig 20. Transaction latency.

https://doi.org/10.1371/journal.pone.0243043.g020

Doctor (f)ePrescription to Pharmacist. The first experiment calculated the Transaction Latency of the proposed blockchain framework as shown in Fig 20. Transaction Latency is the amount of time taken for the transaction to commit and available across the network nodes. if there are *n* number of nodes in the blockchain network, T_{L_n} is the Transaction Latency, T_{C_n} is the confirmation time in the network nodes and T_{S_n} is the transaction submit time in seconds then;

$$T_{L_n} = T_{C_n} - T_{S_n} \tag{5}$$

Seven sets of writing transactions to the network ledger were performed in various transaction sets within a range of 5,10,15,20,30,40 and 50 as shown in Fig 20. Considering the machine configuration in Table 3, it is clear that the initial set of 5 transactions took an average of 80 seconds to commit across the network and the final set of 50 transactions took an average of 160 seconds. The experimental result is further extended to Montecarlo Simulation environment for determining the transaction time for more number of transaction in the range of 50 to 300. It can be seen that average of 450 seconds was required to commit 300 transactions in three peer nodes as shown in Fig 21. Therefore, it is evident that the time taken to execute transactions increases with increase in peers and increase in the number of transactions. This Fig 22 shows a comparative analysis of Transaction Latency of 1 Org 1Peer, 1 Org 2Peer and 1 Org 3Peer. For seven sets of transactions ranging from 5, 10, 15, 20, 30, 40 and 50, It is clear that for 5 transactions, 1 Org 3Peer takes 80 secs to commit across the network in which 1 Org 2Peer took 67 secs to commit and 1 Org 1Peer took an average of 45 seconds to commit across the network. Therefore it shows that, more peers and higher number of organisations exhibit higher latency.

The second experiment calculated the Transaction Throughput of the proposed blockchain framework. The Transaction throughput is the rate at which the blockchain System Under Test (SUT) commits valid transactions in a defined time period at all network nodes. if there are *n* number of nodes in the blockchain network, T_{T_n} is the Transaction Throughput, T_{ct_n} is

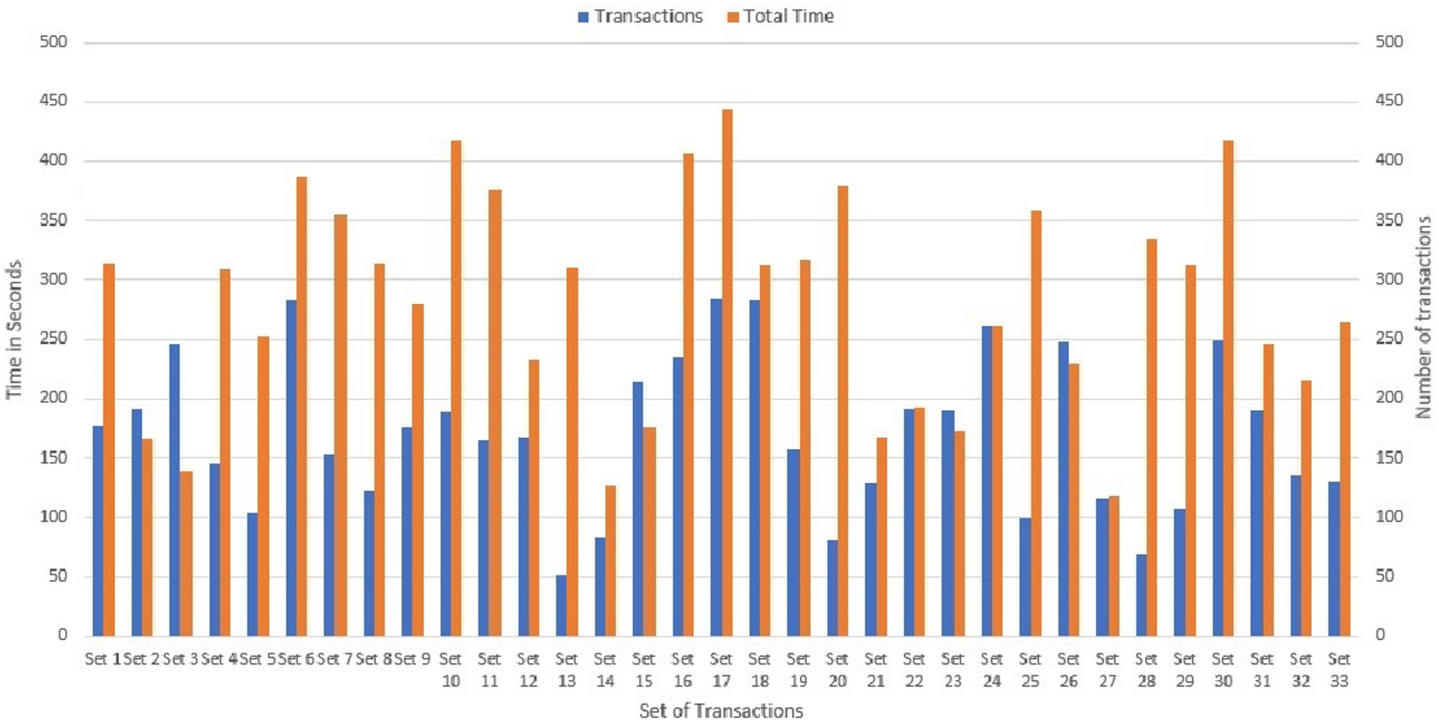


Fig 21. Transaction latency: Montecarlo simulation.

https://doi.org/10.1371/journal.pone.0243043.g021

the total number of committed valid transactions in the network nodes and T_{tot} is the total time in seconds then;

$$T_{T_n} = T_{ct_n}/T_{tot} \tag{6}$$

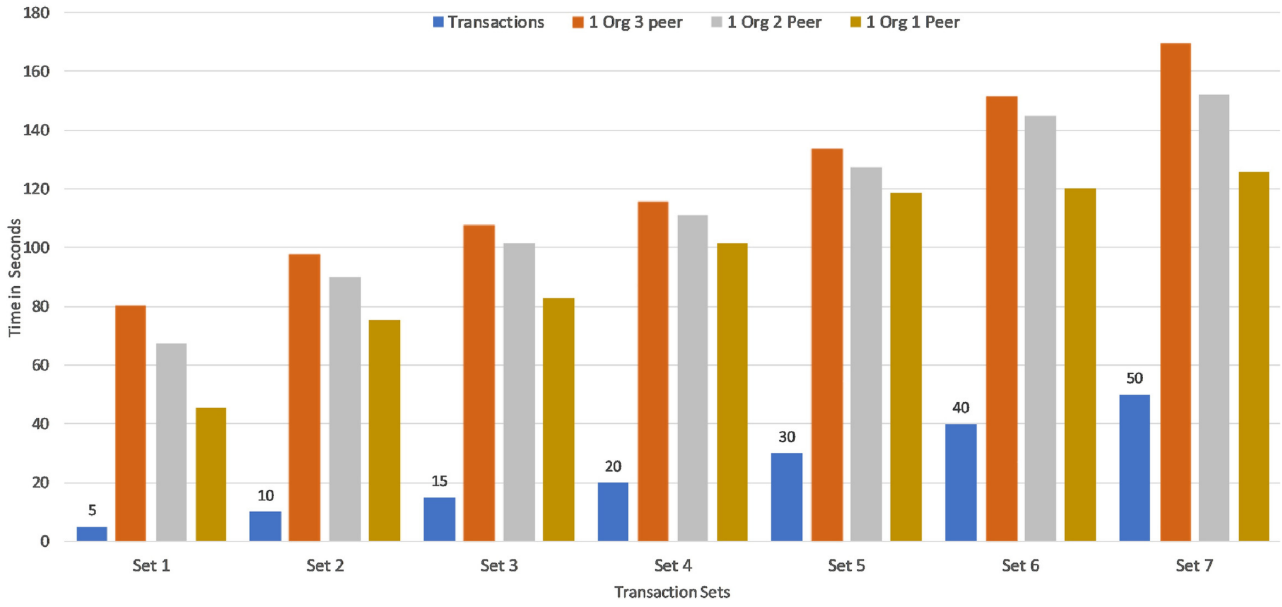


Fig 22. Transaction latency: Comparative analysis.

https://doi.org/10.1371/journal.pone.0243043.g022

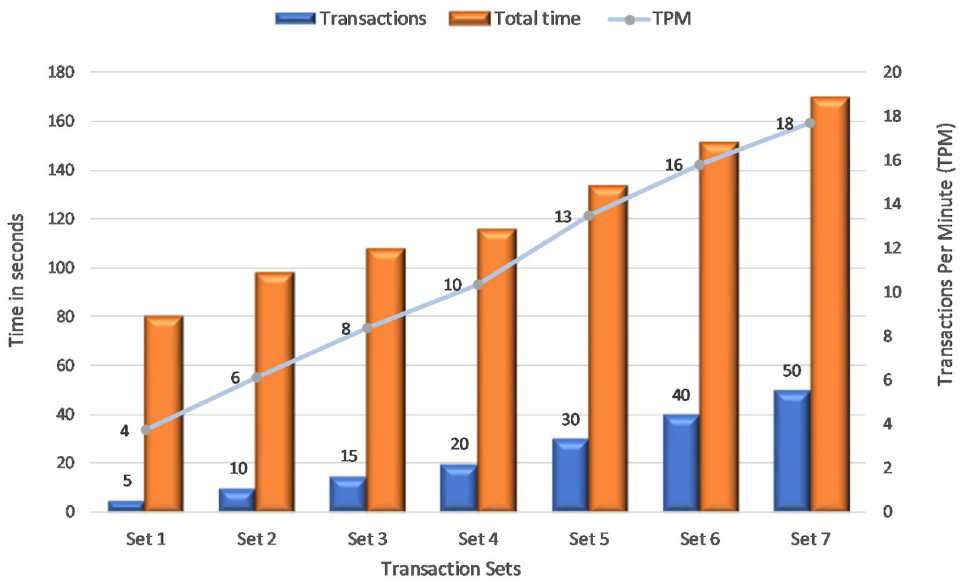


Fig 23. Transaction throughput.

https://doi.org/10.1371/journal.pone.0243043.g023

This Fig 23 portrays Transactions Per Minute (TPM) for various sets of transactions. This experiment runs 7 sets of transactions to determine the TPM in the proposed system. The first set has 5 transactions, took approximately 80 seconds to commit in the network. As a result, the rate of valid transactions across the SUT is 4 TPM in the network. Similarly the last set of 50 transactions took approximately 160 seconds to be available across the network thereby can commit 18 TPM. x-axis indicates the transaction set, y-axis as time in seconds and secondary y-axis for TPM.

Fig 24 demonstrates a comparative analysis of Transaction Throughput that calculates the TPM of the proposed framework for 1 Org 1Peer, 1 Org 2Peer and 1 Org 3Peer. From the Fig 24, It is evident that based on the Transaction Latency in Fig 22, the rate of valid transactions across the SUT is slightly higher for 1 Org 1Peer compared to 1 Org 2Peer and 1 Org 3Peer.

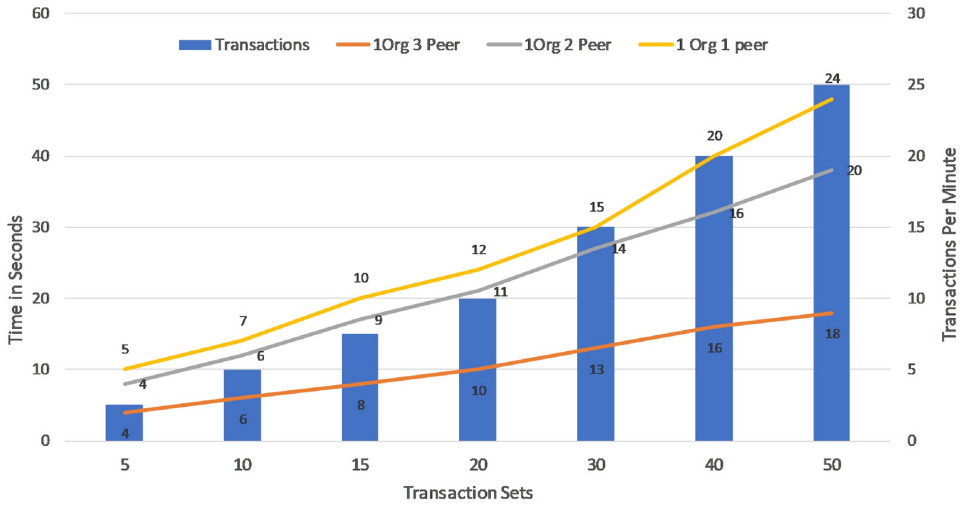


Fig 24. Transaction throughput: Comparative analysis.

https://doi.org/10.1371/journal.pone.0243043.g024

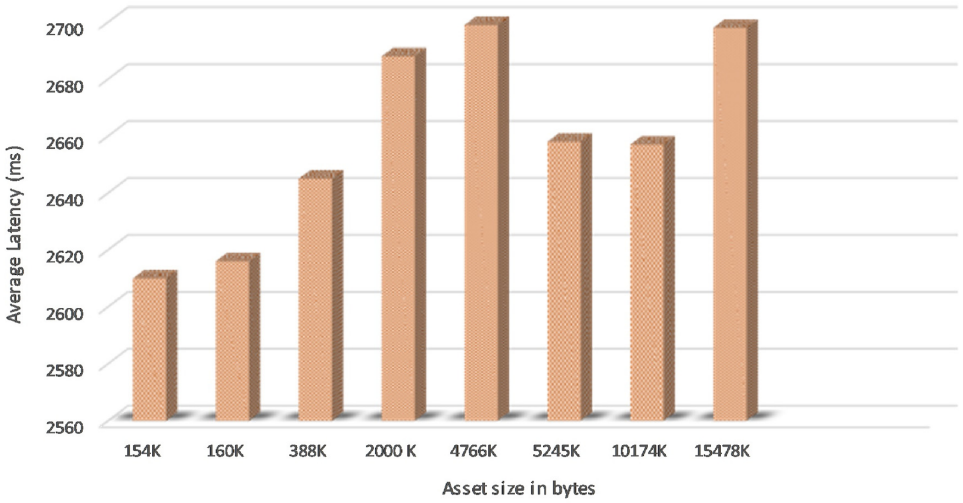


Fig 25. Asset latency.

https://doi.org/10.1371/journal.pone.0243043.g025

The asset latency is the time taken by the SUT to successfully load and write the assets to the couchDB. if there are n number of nodes in the blockchain network, A_{L_n} is the Asset Latency, T_{Res_n} is the Response time and T_{Sub_n} is the asset submit time in milliseconds then;

$$A_{L_n} = T_{Res_n} - T_{Sub_n} \tag{7}$$

Fig 25 shows the Asset Latency of varying assets size in bytes of 5 concurrent users in the proposed system and it is obvious that it took an average latency of 2.7 seconds to commit asset write updates in the couchDB across the network.

It is observed that asset size of 154K bytes took an average of 2.6 seconds and 15478K byte size took an average of 2.7 seconds to commit write updates in the CouchDB. We also extended the experiment to project the number of concurrent users in a range of 5 to 100 and byte size in a range of 154K bytes to 20574K bytes to determine the variation in Asset Latency through Monte carlo simulation and it took an average latency of 3.0 seconds to commit the asset updates in the ledger as shown in Fig 26. Considering the machine configuration in

Fig 26. Asset latency: Montecarlo simulation.

https://doi.org/10.1371/journal.pone.0243043.g026

Table 3, system efficiency is higher as it is obvious that even if the number of users increases from 5 to 100 and assets size increases in the SUT, required a marginally small increase in time to commit the asset updates to the couchDB across the network. Scalability and efficiency have been achieved by uploading a record of 150 MB at a time to the IPFS and the average time taken for 5 concurrent users uploading and retrieval of the record was 60 seconds. Thereby, it can be concluded that the proposed system is capable of processing a large dataset at low latency. Data Provenance can also be attained via preserving user history in the blockchain thereby safeguarding Non-repudiation. Smart contracts combined with blockchain solutions embraces high-level encryption that allows the providers, users, patients and clinicians to ensure patient confidentiality in their health care information and enable it attack-proof. Furthermore, healthchain is designed to enhance scalability of healthcare data by storing hashes on chain and real data in the off chain IPFS. Figs 27 and 28 demonstrates the scalability of IPFS using both the image data and document data with a size comparison upto 100 MB size. The results are obtained from transaction execution of 5 users concurrently upload and download the data in IPFS. Considering the machine requirements, for a 100 MB image file, the system takes an average time of 65 sec to upload the data to IPFS and downloading in an average time of 80 seconds as shown in Fig 27. Also, the system takes an average of 65 seconds uploading time and an average time of 105 seconds for downloading a 100 MB document file as portrayed in Fig 28.

Healthchain is a Patient driven Interoperability framework and employs several security and privacy preserving mechanisms that sustain cyber attacks and internal attacks, however there are still some improvements that could be made to make it as a foolproof solution. Initially, the REST API can be made secure via using HTTPS by encrypting communications between client and server instead of HTTP that is being used nowadays. Secondly, we can employ smart contracts on a large scale that will be executing on more number of nodes for the privacy and safety of patients' information to make it tamper resistant. This work can be extended to multiple nodes for proving the effectiveness of Distributed Ledger Technology in health records as a future work. The implementation of different smart contracts on every

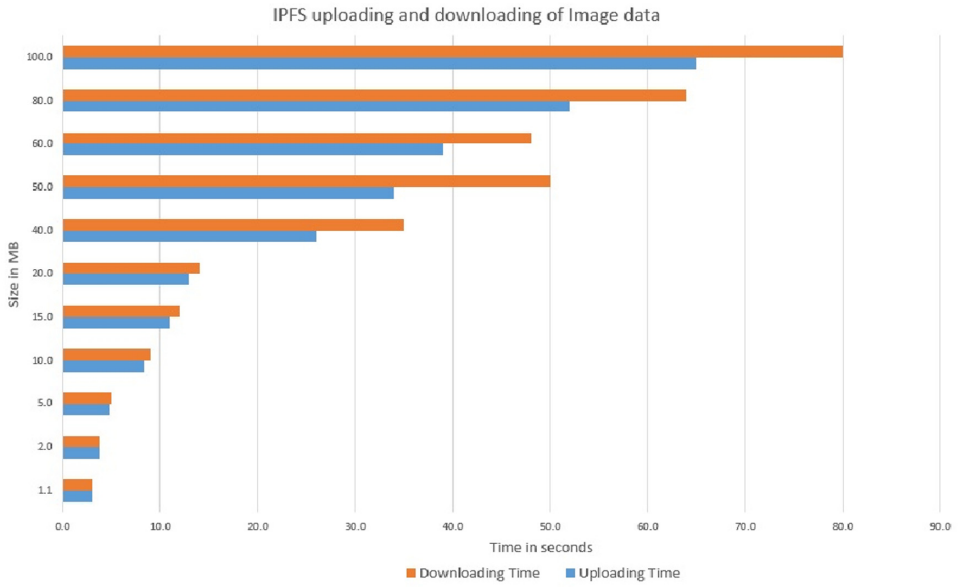


Fig 27. Uploading and downloading time comparison of image data in IPFS.

https://doi.org/10.1371/journal.pone.0243043.g027

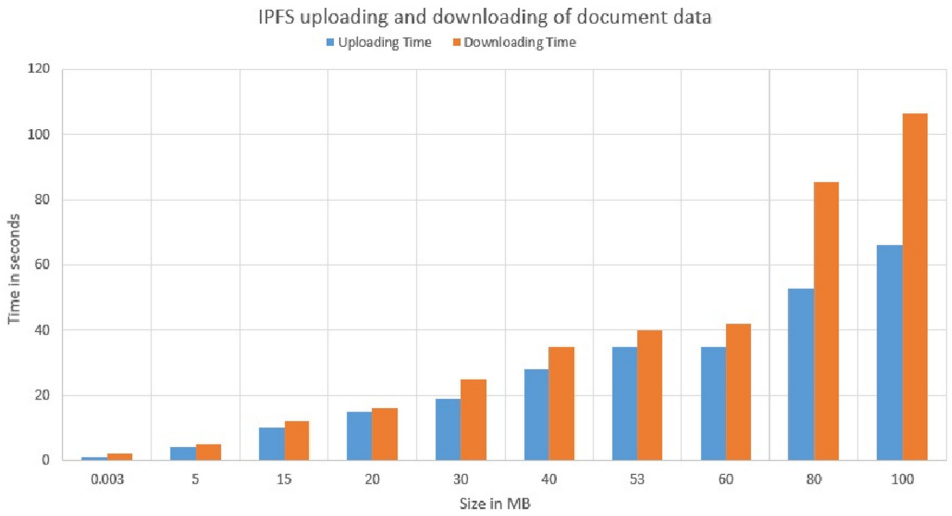


Fig 28. Uploading and downloading time comparison of document data in IPFS.

https://doi.org/10.1371/journal.pone.0243043.g028

node and submitting the node to the system requires several stages of verification which is considered as a future work to prove the efficiency of the proposed system.

8 Conclusion

In this research work, a permissioned blockchain framework has been implemented for secure data storage and access of electronic health records utilizing Hyperledger fabric and Hyperledger composer. Since the blockchain is tamper resistant, the system is tamper proof to handle healthcare records that preserves data privacy, security and integrity. Moreover, no incentive mechanisms for blockchain mining are included that demonstrates the patients' ownership towards their healthcare data. The research proposes an architecture for securing data storage and providing efficient access control between stakeholders viz patients, doctors, pharmacists and other participants via encryption techniques and access control mechanisms. Moreover, a working prototype based on Hyperldger Fabric and Interplanetary File System are made to illustrate the system's viability. The proposed methodology has been implemented and evaluated with some use cases for EHRs and consequently, the framework is successful as a reliable health data network. The result of prototype implementation and analysis proves that the approach is a tamper resistant mechanism as information will be stored as hash values for every healthcare transactions in the blockchain. Moreover, it has enormous potential to ensure privacy, security, integrity, confidentiality and scalability of the e-health information. Performance evaluation of the proposed system is completed using empirical research for various scenarios by configuring asset size, block size, various nodes, asset creation time, transaction sets, for evaluation metrics such as Transaction Latency, Transaction Throughput, Asset Latency and Data Scalability for analysis. Furthermore, this research also explores technology framework and business processes for blockchain applications.

The introduction of this technological innovation which incorporates cryptographic elements offers a more secure and effective framework to store, transfer and access EHR in the cloud environment efficiently. The healthchain prototype based on the blockchain technology is a resilient tamperproof ledger from the test results and it rests heavily on the success. With increase in health data every year, we look forward to refine this prototype with rigorous

simulations in scalability and comparing with other blockchain configurations in a test bed arena that will invite further attention in future research work.

Acknowledgments

The authors would like to thank Prof. Yuan Miao and Taman Shergill for their valuable contributions, comments, suggestions and reviews.

Author Contributions

Conceptualization: Shekha Chenthara, Khandakar Ahmed, Hua Wang.

Data curation: Shekha Chenthara.

Formal analysis: Shekha Chenthara.

Investigation: Shekha Chenthara.

Methodology: Shekha Chenthara.

Project administration: Khandakar Ahmed, Hua Wang, Frank Whittaker, Zhenxiang Chen.

Resources: Shekha Chenthara.

Software: Shekha Chenthara.

Supervision: Khandakar Ahmed, Hua Wang, Frank Whittaker, Zhenxiang Chen.

Validation: Shekha Chenthara.

Visualization: Shekha Chenthara.

Writing – original draft: Shekha Chenthara.

Writing – review & editing: Shekha Chenthara, Khandakar Ahmed, Hua Wang.

References

1. Kruse CS, Mileski M, Vijaykumar AG, Viswanathan SV, Suskandla U, Chidambaram Y. Impact of electronic health records on long-term care facilities: Systematic review. JMIR medical informatics. 2017; 5(3). https://doi.org/10.2196/medinform.7958 PMID: 28963091
2. Chenthara S, Ahmed K, Wang H, Whittaker F. Security and Privacy-preserving Challenges of e-Health Solutions in Cloud Computing. IEEE Access. 2019.
3. Cheng K, Wang L, Shen Y, Wang H, Wang Y, Jiang X, et al. Secure k-nn query on encrypted cloud data with multiple keys. IEEE Transactions on Big Data. 2017.
4. Li P, Guo S, Miyazaki T, Xie M, Hu J, Zhuang W. Privacy-preserving access to big data in the cloud. IEEE Cloud Computing. 2016; 3(5):34–42. https://doi.org/10.1109/MCC.2016.107
5. Masud MAH, Huang X, Islam MR. A Novel Approach for the Security Remedial in a Cloud-based E-learning Network. Journal of Networks. 2014; 9(11):2934.
6. Dong S, Abbas K, Jain R. A survey on distributed denial of service (DDoS) attacks in SDN and cloud computing environments. IEEE Access. 2019; 7:80813–80828. https://doi.org/10.1109/ACCESS.2019.2922196
7. Abbas A, Khan SU. A review on the state-of-the-art privacy-preserving approaches in the e-health clouds. IEEE Journal of Biomedical and Health Informatics. 2014; 18(4):1431–1441. https://doi.org/10.1109/JBHI.2014.2300846 PMID: 25014943
8. Brewer R. Ransomware attacks: detection, prevention and cure. Network Security. 2016; 2016(9):5–9. https://doi.org/10.1016/S1353-4858(16)30086-1
9. Wang H, Zhang Z, Taleb T. Special issue on security and privacy of IoT. World Wide Web. 2018; 21 (1):1–6. https://doi.org/10.1007/s11280-017-0490-9
10. Wang H, Wang Y, Taleb T, Jiang X. Special issue on security and privacy in network computing. World Wide Web. 2020; 23(2):951–957. https://doi.org/10.1007/s11280-019-00704-x

11. Nakamoto S, et al. Bitcoin: A peer-to-peer electronic cash system. 2008.
12. Adams C, Lloyd S. Understanding public-key infrastructure: concepts, standards, and deployment considerations. Sams Publishing; 1999.
13. Sun W, Guo H, He H, Dai Z. Design and optimized implementation of the SHA-2 (256, 384, 512) hash algorithms. In: 2007 7th International Conference on ASIC. IEEE; 2007. p. 858–861.
14. Baliga A. Understanding blockchain consensus models. Persistent. 2017; 2017(4):1–14.
15. Zhang E, Li M, Yiu SM, Du J, Zhu JZ, Jin GG. Fair hierarchical secret sharing scheme based on smart contract. Information Sciences; 546:166–176. https://doi.org/10.1016/j.ins.2020.07.032
16. Androulaki E, Barger A, Bortnikov V, Cachin C, Christidis K, De Caro A, et al. Hyperledger fabric: a distributed operating system for permissioned blockchains. In: Proceedings of the Thirteenth EuroSys Conference. ACM; 2018. p. 30.
17. Mohurle S, Patil M. A brief study of wannacry threat: Ransomware attack 2017. International Journal of Advanced Research in Computer Science. 2017; 8(5).
18. Berghel H. Equifax and the latest round of identity theft roulette. Computer. 2017; 50(12):72–76. https://doi.org/10.1109/MC.2017.4451227
19. Shu J, Jia X, Yang K, Wang H. Privacy-preserving task recommendation services for crowdsourcing. IEEE Transactions on Services Computing. 2018.
20. Dannen C. Introducing Ethereum and Solidity. Springer; 2017.
21. Yin S, Bao J, Zhang Y, Huang X. M2m security technology of cps based on blockchains. Symmetry. 2017; 9(9):193. https://doi.org/10.3390/sym9090193
22. Yue X, Wang H, Jin D, Li M, Jiang W. Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control. Journal of medical systems. 2016; 40(10):218. https://doi.org/10.1007/s10916-016-0574-6 PMID: 27565509
23. Azaria A, Ekblaw A, Vieira T, Lippman A. Medrec: Using blockchain for medical data access and permission management. In: 2016 2nd International Conference on Open and Big Data (OBD). IEEE; 2016. p. 25–30.
24. Ivan D. Moving toward a blockchain-based method for the secure storage of patient records. In: ONC/NIST Use of Blockchain for Healthcare and Research Workshop. Gaithersburg, Maryland, United States: ONC/NIST; 2016.
25. Zhang E, Liu FH, Lai Q, Jin G, Li Y. Efficient Multi-Party Private Set Intersection Against Malicious Adversaries. In: Proceedings of the 2019 ACM SIGSAC Conference on Cloud Computing Security Workshop; 2019. p. 93–104.
26. Shen B, Guo J, Yang Y. MedChain: Efficient Healthcare Data Sharing via Blockchain. Applied Sciences. 2019; 9(6):1207. https://doi.org/10.3390/app9061207
27. Zyskind G, Nathan O, et al. Decentralizing privacy: Using blockchain to protect personal data. In: 2015 IEEE Security and Privacy Workshops. IEEE; 2015. p. 180–184.
28. Li M, Sun X, Wang H, Zhang Y, Zhang J. Privacy-aware access control with trust management in web service. World Wide Web. 2011; 14(4):407–430. https://doi.org/10.1007/s11280-011-0114-8
29. Dagher GG, Mohler J, Milojkovic M, Marella PB. Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology. Sustainable cities and society. 2018; 39:283–297. https://doi.org/10.1016/j.scs.2018.02.014
30. Zhang P, White J, Schmidt DC, Lenz G, Rosenbloom ST. Fhirchain: applying blockchain to securely and scalably share clinical data. Computational and structural biotechnology journal. 2018; 16:267–278. https://doi.org/10.1016/j.csbj.2018.07.004 PMID: 30108685
31. Wang H, Song Y. Secure cloud-based EHR system using attribute-based cryptosystem and blockchain. Journal of medical systems. 2018; 42(8):152. https://doi.org/10.1007/s10916-018-0994-6 PMID: 29974270
32. Jiang S, Cao J, Wu H, Yang Y, Ma M, He J. Blochie: a blockchain-based platform for healthcare information exchange. In: 2018 ieee international conference on smart computing (smartcomp). IEEE; 2018. p. 49–56.
33. Jamil F, Ahmad S, Iqbal N, Kim DH. Towards a Remote Monitoring of Patient Vital Signs Based on IoT-Based Blockchain Integrity Management Platforms in Smart Hospitals. Sensors. 2020; 20(8):2195. https://doi.org/10.3390/s20082195 PMID: 32294989
34. Margheri A, Masi M, Miladi A, Sassone V, Rosenzweig J. Decentralised Provenance for Healthcare Data. International Journal of Medical Informatics. 2020; p. 104197. https://doi.org/10.1016/j.ijmedinf.2020.104197 PMID: 32540775

35. Roehrs A, da Costa CA, da Rosa Righi R, da Silva VF, Goldim JR, Schmidt DC. Analyzing the performance of a blockchain-based personal health record implementation. Journal of biomedical informatics. 2019; 92:103140. https://doi.org/10.1016/j.jbi.2019.103140 PMID: 30844481

36. Dwivedi AD, Srivastava G, Dhar S, Singh R. A decentralized privacy-preserving healthcare blockchain for IoT. Sensors. 2019; 19(2):326. https://doi.org/10.3390/s19020326 PMID: 30650612

37. Ateniese G, Fu K, Green M, Hohenberger S. Improved proxy re-encryption schemes with applications to secure distributed storage. ACM Transactions on Information and System Security (TISSEC). 2006; 9(1):1–30. https://doi.org/10.1145/1127345.1127346

38. Wood G, et al. Ethereum: A secure decentralised generalised transaction ledger. Ethereum project yellow paper. 2014; 151(2014):1–32.

39. Dubovitskaya A, Xu Z, Ryu S, Schumacher M, Wang F. Secure and trustable electronic medical records sharing using blockchain. In: AMIA Annual Symposium Proceedings. vol. 2017. American Medical Informatics Association; 2017. p. 650.

40. Song T, Pan L, Păun G. Asynchronous spiking neural P systems with local synchronization. Information Sciences. 2013; 219:197–207. https://doi.org/10.1016/j.ins.2012.07.023

41. Song T, Rodríguez-Patón A, Zheng P, Zeng X. Spiking neural P systems with colored spikes. IEEE Transactions on Cognitive and Developmental Systems. 2017; 10(4):1106–1115. https://doi.org/10.1109/TCDS.2017.2785332

42. Song T, Pan L, Wu T, Zheng P, Wong MD, Rodríguez-Patón A. Spiking neural P systems with learning functions. IEEE transactions on nanobioscience. 2019; 18(2):176–190. https://doi.org/10.1109/TNB.2019.2896981 PMID: 30716044

43. Sukhwani H, Martínez JM, Chang X, Trivedi KS, Rindos A. Performance modeling of pbft consensus process for permissioned blockchain network (hyperledger fabric). In: 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS). IEEE; 2017. p. 253–255.

44. Mingxiao D, Xiaofeng M, Zhe Z, Xiangwei W, Qijun C. A review on consensus algorithm of blockchain. In: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE; 2017. p. 2567–2572.

45. Dhillon V, Metcalf D, Hooper M. The hyperledger project. In: Blockchain enabled applications. Springer; 2017. p. 139–149.

46. Chenthara S, Ahmed K, Wang H, Whittaker F. A Novel Blockchain Based Smart Contract System for eReferral in Healthcare: HealthChain. In: International Conference on Health Information Science. Springer; 2020. p. 91–102.

47. Benet J. Ipfs-content addressed, versioned, p2p file system. arXiv preprint arXiv:14073561. 2014.

48. Chenthara S, Ahmed K, Whittaker F. Privacy-Preserving Data Sharing using Multi-layer Access Control Model in Electronic Health Environment. EAI Endorsed Transactions on Scalable Information Systems. 2019; 6(22).

49. Cheney J, Chong S, Foster N, Seltzer M, Vansummeren S. Provenance: a future history. In: Proceedings of the 24th ACM SIGPLAN conference companion on Object oriented programming systems languages and applications; 2009. p. 957–964.