Journal of Medical Systems (2018) 42: 136
https://doi.org/10.1007/s10916-018-0993-7

SYSTEMS-LEVEL QUALITY IMPROVEMENT

MedBlock: Efficient and Secure Medical Data Sharing Via Blockchain

Kai Fan[1] · Shangyang Wang[1] · Yanhui Ren[1] · Hui Li[1] · Yintang Yang[2]

Received: 27 February 2018 / Accepted: 12 June 2018 / Published online: 21 June 2018

Abstract

With the development of electronic information technology, electronic medical records (EMRs) have been a common way to store the patients' data in hospitals. They are stored in different hospitals' databases, even for the same patient. Therefore, it is difficult to construct a summarized EMR for one patient from multiple hospital databases due to the security and privacy concerns. Meanwhile, current EMRs systems lack a standard data management and sharing policy, making it difficult for pharmaceutical scientists to develop precise medicines based on data obtained under different policies. To solve the above problems, we proposed a blockchain-based information management system, MedBlock, to handle patients' information. In this scheme, the distributed ledger of MedBlock allows the efficient EMRs access and EMRs retrieval. The improved consensus mechanism achieves consensus of EMRs without large energy consumption and network congestion. In addition, MedBlock also exhibits high information security combining the customized access control protocols and symmetric cryptography. MedBlock can play an important role in the sensitive medical information sharing.

Keywords Medical data sharing · Blockchain · Security · Privacy preserving · Openness · Efficiency

Introduction

The era of information has arrived. Due to the development of digitization and cloud storage, more and more data is transferred from paper to the electronic equipment [1]. The digitization storage of information in medical institutions is popular. Electronic medical records are usually stored in a private database, which brings a problem that patients leave data scattered across various hospitals because life events take them away from one hospital and into another. It is noteworthy that these records are generated in hospitals after patients visit them by recording in electronic medical records. Therefore, patients lose easy access to past data even if it belongs to them [2]. When they visit other hospitals, they are not available to provide the doctor their detailed past medical records, because their past records were stored in somewhere else. Interoperability challenges between different hospital systems pose tough hurdles to data sharing. It is difficult for people to obtain the data they want because of lack of unified data management and sharing.

On the one hand, data requestors want to acquire the patients' past medical records in order to determine their treatment plans [3]. On the other hand, the medical records stored in private databases contain much privacy related to hospital and patient. Therefore, querying data and sharing may bring serious risk of confidentiality for data providers. It is not everyone can access to the EMRs. To meet the high demands on data sharing [4], some researchers have proposed some relative schemes about cloud storage and computing technologies to provide suitable solutions to compression storage and processing demands. However, cloud service providers (CSP) face some significant hurdles in persuading hospitals to use centralized cloud services due to the adverse risks posed on exposing the contents on data. Some cryptographic schemes have been proposed to solve these problems about medical data sharing. But they are insufficient, the disadvantages still exist [5, 6]. For the hospital, the sheer volume of data stored in third parties is not reassuring [7]. These semi-trusted third parties may misuse and disclose providers' privacy. Article 17 of the

This article is part of the Topical Collection on *Systems-Level Quality Improvement*

✉ Kai Fan
kfan@mail.xidian.edu.cn

[1] State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China

[2] Key Lab. of Minist. of Educ. for Wide Band-Gap Semicon. Materials and Devices, Xidian University, Xi'an 710071, China

soon-enforceable General Data Protection Regulation in the EU has strengthened the rights of individuals and imposed many restrictions on the storage of personal data by third parties. Personal medical data would come under the protection of privacy laws and many of legal provisions would not allow personal data to be kept perpetually. In the face of the legal disputes caused by data leakage, hospitals lack a reason to provide data to third parties.

For the government, medical records need to be monitored to find illegal medical procedures. In the meantime, researchers also hope to analyze past medical data to make a breakthrough for the discovery of new techniques and therapies for curing diseases [8]. The Institute for Business Value at IBM issued a whitepaper titled, "Healthcare rallies for blockchains: Keeping patients at the center" [9]. The survey predicts that blockchain technology will be used to manage clinical trial records, supervised compliance and EHRs. The Chinese government recently set up a blockchain industrial park in Hangzhou and hopes that more institutions and companies can tap the value of blockchain technology in more field.

Motivated by the above issues, a further research was made on the sharing of medical information. When designing new system to overcome these barriers, the patients' needs are obliged to be placed in the first place. We must ensure that patients using the system can easily query their past medical records even if they are stored in different hospitals' database. Therefore, the blockchain would be very suitable for providing an appropriate solution for this problem through its attractive features such as openness and verifiability. The decentralized nature of the blockchain avoids performance bottlenecks coming from frequent network requests and responses [10].

Considering the disquiet of the hospitals themselves, hospitals have the right to store data in their original way instead of uploading data to semi-trusted third parties. What they need to do is just upload the encrypted summary data and hash value to the blockchain so that users can retrieve and verify the data. When patients query the information on the blockchain, the retrieval mechanism on the ledger can help us quickly retrieve the location of encrypted information, which greatly improve the efficiency of the system [11, 12]. The design employs a way to submit requests by turns and a hybrid consensus mechanism which can effectively avoid network congestion caused by data flood peak, and reach a consensus with few resources to realize low-power green communication. To assure the security and the privacy of medical data, we need to develop an effective data encryption solution. The asymmetric cryptography is adopted to encrypt these data in this paper, which is efficient and low cost. If someone tries to read a record, he must know the corresponding decryption key. On condition that attackers don't have decryption key, what they get is meaningless. In many fields, ring signature algorithm [13, 14], group signature and zero-knowledge proof scheme [15] are used to enhance the anonymity of data. We achieve the same effect in MedBlock based on access control protocols by hiding the signature information and encrypting summaries for unauthorized users.

The rest of the article is organized as follows: in Section II, we review the related work about privacy protection of information and medical information sharing process, and then discuss their limitations. The scheme and system model of this paper will be described in Section III. Next in Section IV, we will analyze the performance of the MedBlock in detail. Finally, Section V concludes the paper and illustrates future extensions.

Related works

In this section, research trends about medical data sharing via cloud service and blockchain technology are outlined.

Zyskind et al. proposed a blockchain usage for access control management and secure data storage [16]. In the paper, encrypted data is stored in trusted third party hosting services and logging log of events on the blockchain. There is no credible third party in the real world, which brings the risk of data disclosure.

Asaph Azariaet al. presented a blockchain-based data sharing system which was used as decentralized record management system to handle EMRs. They provide miners with access to aggregate, and reward the data to bookkeepers [17]. However, the efficiency of data usage is not satisfactory. And it is illegal to gather patient data together and share them as rewards.

Recently, Xia et al. proposed a system to manage and protect medical records effectively. The system is blockchain-based and provides data protection and management for shared medical data in cloud repositories among big data entities. They ensure data security through verifying their identities and cryptographic keys [18]. But the scheme does not take the concerns of the risk of data disclosure. That is, the hospital is reluctant to give the data to the third party, which makes the scheme untenable at the beginning.

Esposito et al. [19] detailed the drawbacks of using cloud storage technology to establish a data sharing system in the medical field. They also raised the possible challenges of using blockchain technology in medical data sharing (such as privacy protection). However, the article does not propose practical schemes to address these challenges.

Li et al. proposed a novel patient-centric framework and a suite of mechanisms for access control of data to PHRs stored in semi-trusted servers. They leverage ABE techniques to encrypt each patient's PHR file [20]. However, ABE has many disadvantages. Once a user modifies his access polices, system needs extra computational expenditures to execute

attribute revocation and encrypt data again. The non-tampered nature of the blockchain also makes CP-ABE-based access control unable to be modified in ledger, so it is unsuitable for this scheme [21, 22]. To reduce the computational cost, Gu et al. [23] proposed a more efficient ABS scheme with the monotone predicates. Unfortunately, their general form cannot solve the problems caused by the modification of access policies. Guo et al. [24] introduced an attribute-based signature scheme with multiple authorities to guarantee the validity of EHRs encapsulated in blockchain. After treatment, all patient information including EHRs, consumption records, insurance records, etc. is encapsulated in one block. Medical data, such as imaging and treatment plans, however, can be large and relational that requires searching. Ferdous et al. [25] presented DRAMS, a blockchain-based decentralized monitoring infrastructure for a distributed access control system. The scheme provides a solution to data security, but it does not solve the problem of efficient sharing of data.

In this paper, we propose a secure system based on blockchain to share electronic medical records among authorized users. The retrieval mechanism on the ledger allows the users easily to get involved and actively find the information they want in an efficient way. We use a simple and effective access control and encryption strategy to ensure the security and privacy of information with smaller delay and energy cost. This mechanism ensures that the patients' identity information is not leaked out which achieves the same effect as the ring signature.

The overview of medblock

This section discusses the MedBlock model. Firstly, we introduce the overall data flow and components of the system. Then, the details of the blockchain are introduced. Finally, we show the business rules, such as the access control protocols, consensus mechanism, and the detailed format of the ledger.

System architecture

The architecture of system is represented in Fig. 1.

Certificate authority

CA is both a system administrator and an authority management agency. It will promptly remove malicious nodes from the system to ensure the health of system. At the same time, it is responsible for the generation, distribution and management of digital certificate. The patients' public-private key () is also generated by the CA. In order to facilitate the state regulation of the information on the block and medical research, the CA may use the patients' private key to decrypt the data on block in certain circumstances.

User layer

The user layer consists of all the users who want to access data from the system, such as patients. As the owner of information, patients are more concerned about the data privacy and the convenience during information querying. When a patient visits a hospital that can intervene in the system, he can get the summaries of the past medical records stored on the chain and find a detailed electronic medical record according to the summaries by his private key. Before leaving, he can use public key to encrypt the medical information generated by this visit and sign the data through his private key.

Processing layer

The processing layer is composed of the servers and databases of the hospital. Community hospitals generally do not have the database to store detailed patient information. Their function is relatively simple, namely uploading the encrypted medical information through system clients and helping the patient to query the summaries on the block. Before uploading the encrypted summaries to the superior hospital, the community hospital also needs to sign the data by its private key. Authorized community hospitals can also serve as consensus nodes(orderers) in the system to increase the fault-tolerant capability of the system.

The various departments in the hospital assume the same tasks as community hospitals in the system. It is worth mentioning that EMRs are stored in the hospitals' database, only the summaries and the hash value of EMRs are encrypted and uploaded.

National hospitals bear the major task in the system. The hospital needs to arrange the encrypted summaries of EMRs uploaded by the sub-area community hospitals and the various departments. After sorting the data, the hospital will pack sorted data into blocks and send a request to consensus nodes to add blocks. After reaching a consensus, the committers will add the blocks to their own ledger. In our framework, hospitals need to undertake the task of sending requests and the tasks of consensus nodes. Hospitals can choose to maintain the ledger or not according to their own respective realities because this is not a task that must be undertaken. However, the consensus tasks and initiating request tasks are borne by them, which means that they should serve as orderers and endorsers. Links between each other are shown in Fig. 2.

Medblock

Components

The MedBlock consists of six modules: client, endorser, orderer, committer, database and ledger.

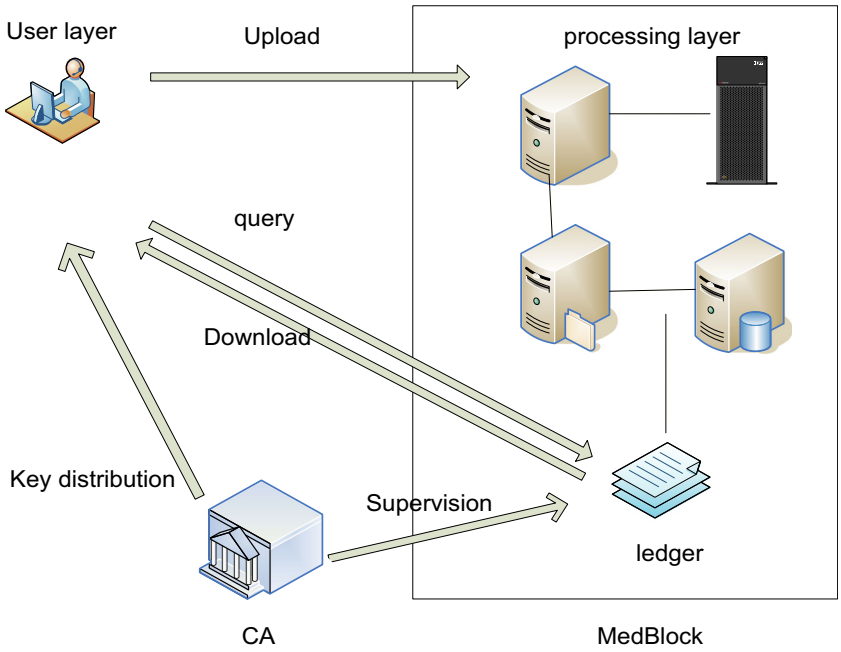

Fig. 1 The architecture of system

Database refers to the hospitals' data storage method for storing EMRs. It can be a database or cloud storage. When an authorized user requests to access EMRs, the Database will provide the relevant data to the user.

We divide the tasks of nodes into four parts that each node only takes a single task. In this way, we can increase the efficiency of the system and configure the number of different nodes as needed, which is very important for the scalability of the system. Clients exist in the various departments of hospitals to upload and download data. Endorsers are responsible for initiating the proposal. Orderers are in charge of reaching consensus. Committers

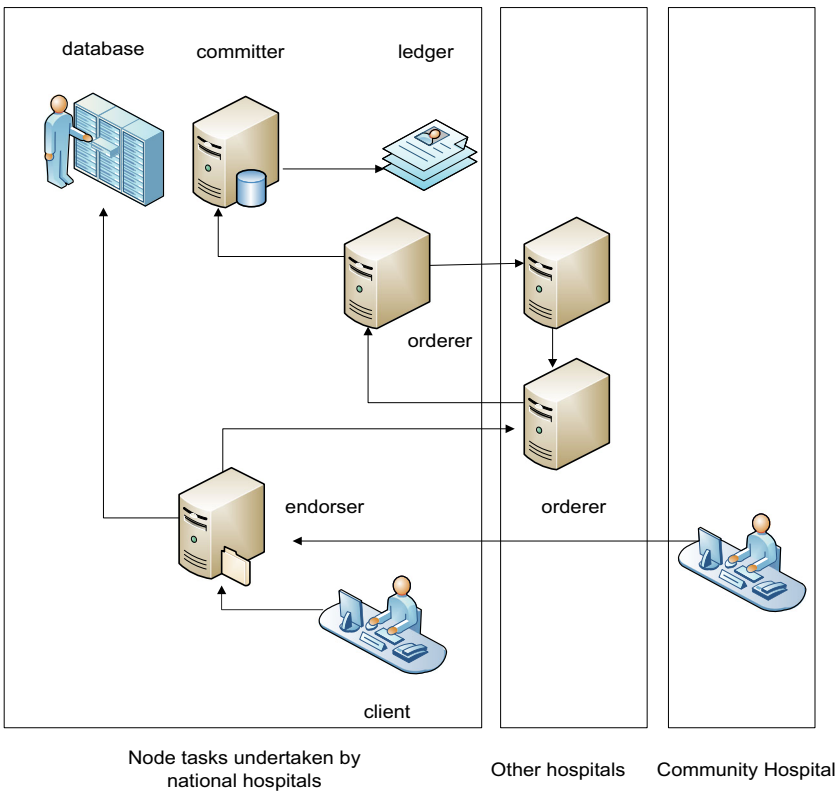

Fig. 2 Schematic diagram of tasks undertaken by different hospitals

are responsible for adding the data to ledger based on consensus. Meanwhile, committers also need to be responsible for the consistency of the ledger by broadcasting the hash value of the ledger to the whole network periodically. Committers need to find out the problematic nodes to keep the consistency of the whole network ledgers.

National hospitals need to undertake the tasks of all nodes, while community hospitals can only become endorsers or optional orderers. The complete process is as follows (shown as Fig. 3):

Step1: After collecting the users' EMRs and organizing the summary M, the client encrypts it with the patients' public key. And then the client uses the private key of the patient and the private key of the department to sign the for assuring the information is correct. Adding the hash value of the EMRs to the top of, the client sends the to endorser.

Step2: The endorser checks whether the signature of the is complete. If completed, the endorser saves data to the local cache and sends the receipt to the client. After this, the client continues to wait for the receipt of the orderers.

Step3: The endorser sorts all the uploaded and packs sorted into blocks according to the upload time. When it is the endorser' turn to become the primary, the endorser would send proposals of adding blocks to orderers.

Step4: Consensus nodes reach a consensus based on consensus algorithm and send the consensus to committers.

Step5: After collecting enough confirming receipts, the endorser sends the successfully uploaded information to the client.

Step6: The committer adds the blocks into the ledger according to the consensus result.

Step7: If the client has not received a confirmation receipt for a long time, it could choose another endorser to initiate the request again.

Step8: When all the blocks to be uploaded have been confirmed, endorser will broadcast information to the whole network, so that the next endorser will become the primary.

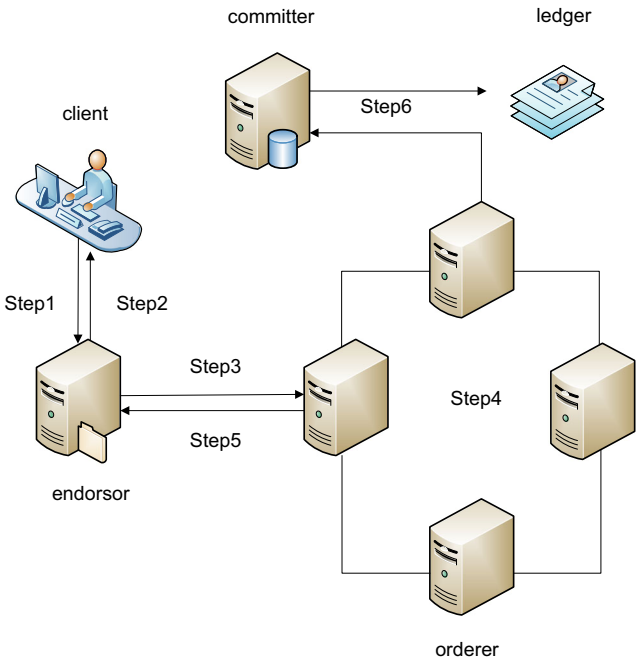

Fig. 3 The function of each node and the process of adding blocks

Consensus mechanism

In order to avoid excessive consumption of energy and centralization of power, the mechanism of traditional Practical Byzantine Fault Tolerance and Delegated Proof of Stake are not suitable. We have developed an efficient hybrid-consensus mechanism based on the actual situation. Like a board vote, nodes within the same region vote to determine a node as the endorser of the region, acting on behalf of them in a responsible position for verification and sending proposals. The selected node is granted a view ((K_{pub}; K_{pri}; $View_{num}$) to mark the node so that the entire network receives the nodes' information. If more than half of the nodes assume that the existing endorser is already crashed, they may re-initiate the election to elect a new endorser. All endorsers will submit the proposal in turn according to a certain order. If we choose to submit medical data in real time, it is obvious that peaks and valleys of data traffic may arise. After all, few people visit the hospital from midnight to dawn. Compared to the system efficiency, real-time data upload is not very important. When an endorser becomes the primary, MedBlock needs to select the consensus nodes under the current primary. We use the beacon continuously generate random numbers to determine which nodes can be orderers. These orderers also need to recalculate when the primary node is changed. When the orderers receive the request, they will reach a consensus based on the Practical Byzantine Fault Tolerance Algorithm.

The algorithm allows data to be uploaded in turn and effectively avoids network congestion caused by patients visiting the hospital in a centralized time. If a new joining node wants to become an endorser or orderer, the node needs to be authenticated first, which ensures that most of the nodes in the system are trustworthy. If most nodes of the system are honest, the system can reach the correct consensus. Using the hybrid consensus mechanism, we can avoid unnecessary waste of resources and achieve green communication.

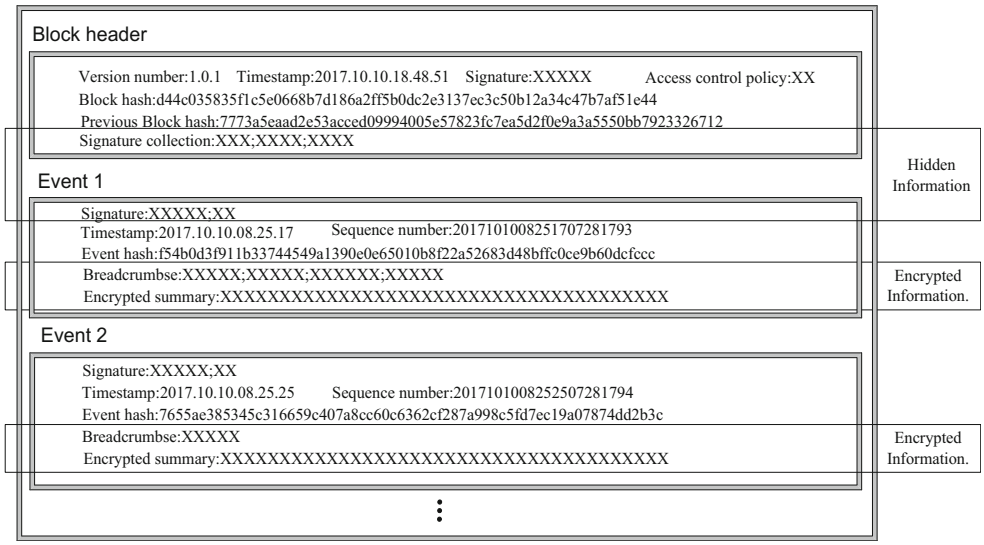

Fig. 4 The Structure of MedBlock

Ledger structure

We show the format of a block containing data in Fig. 4. Just like the Bitcoin, our block is a Merkle Tree-based structure. The first structure is block header. The items included in the header are as follows:

Version number

Signature: The digital signature of endorser to assure the source of the block.
Signature collection: Signatures of events on the block to improve the efficiency of information retrieval.
Access control protocol: A policy to filter illegal users.
Block hash: The SHA256 hash of the current block. The value is calculated by hashing all hash values of events to ensure the immutability of the block.

$$Hash_{block} \leftarrow Hash(Hash_{event1} + Hash_{event2} + Hash_{event3} + \ldots) \quad (1)$$

Previous block hash: The hash value of previous block, used to connect and verify.
Timestamp: Signifying the time that the block is legitimate to add to the blockchain. The timestamp is added by orderers.

A block will contain multiple events which are independent and have nothing with each other. We will introduce the format of events including the following sections.

Timestamp: A timestamp of when the summary was received by the endorser.

Signature: The digital signatures of patient and provider to assure the source and authenticity of data.
Sequence number: This value is a unique index of data.
Event hash: The SHA256 hash of the encrypted summary. The integrity and authenticity of the encrypted information can be verified by the hash value.
Encrypted summary.

Finally, we discuss the composition of the encrypted summary, which is also the most important part of the ledger. The encrypted summary is made up of following modules.

Diagnostic information is written in plaintext to reduce the size of data which contains disease description, examination results and treatment plan.

Pointer of record and the hash value of EMRs. This is a record to find the true storage address of information and ensure the EMRs not tampered.

Bread crumbs mechanism

Bread crumbs mechanism is also an important part of the ledger. However, the problem of how to efficiently find the encrypted information that users add to the ledgers needs to be solved. By comparing the encrypted keywords, users can find the information they are interested in, but it needs to retrieve the entire ledger, which is inefficient. We can make a retrieval directory for the users' past EMRs by classifying the patients' past encrypted summaries based on the departments of hospital and recording the location of the data. Used to improve index efficiency, Bread crumbs records the hash value of the patient-related blocks which are classified according to the hospital departments. If the patient has not visited some

departments, the relevant hash value record is null. After that, the patient only needs to update the relevant hash value records, which brings great efficiency improvement for data querying. When a user wants to look for some information about a past EMR, he can quickly find the corresponding block based on the crumbs. The bread crumbs record hash values are in a tabular format. (Table 1).

Data downing

This section describes how to download and read information. Openness of the ledger means everyone can supervise and browse information. But they can obtain the signature collection and encrypted summary only if users can satisfy access control protocol on the public blockchain ledger. Authorized users can view the latest blocks related to them and use their private key to decrypt the encrypted data. Users can quickly query the information they want based on the bread crumbs records and judge whether to view the EMR based on the diagnostic information. According to the pointer of record, patients can send a request to query the corresponding database for his EMRs. (Fig.5)

Access control protocol

In addition to caring whether the system is easy to use, users are also concerned about the privacy contained in the records. So the formulate suitable access control protocol is needed to avoid unauthorized users from getting sensitive information.

The openness of the blockchain means that everyone can view the content on the block. However, due to privacy reasons, we will conceal the signatures collection in the title of the ledger, the bread crumbs, the signature of event and the encrypted summaries to unauthorized users. For unauthorized users, we only allow them to verify the hash value to ensure that the information has not been tampered with.

We require visitors to provide their signature as the mainly identification of their identity. The system traverses the block until it finds the right block by comparing the signature with signature collection on the ledger. Whether the user can see the encrypted content on the block depends on the result of the comparison. After the user is authorized, the system allows the user to view the hidden information. If the user wants to view the encrypted summary, they can use their private key to decrypt the data. Although, this access control protocol is relatively simple, its security, efficiency and granularity have met our needs. We can achieve the same effect of ring signature and zero knowledge proof.

Access control protocol

Initialization:

$getAction, getHashvalue, AccessControl;$

$getAccessSig,\ getCollectionSig;$

Ensure: *Setting up functions;*

$func(getHashvalue);$

$func(getAction);$

$func(getAccessSig);$

$func\ (getCollectionSig);$

$func\ (accessControl);$

for $(func\ (getAction) == decrypted\ data)$

do $func(accessControl\ );$

$H \leftarrow func(getHashvalue);$

$Sig_{acc} \leftarrow func(getAccessSig);$

$Sig_{collec} \leftarrow func(getCollectionSig);$

end for

if $Sig_c \in Sig_c$

Allow access to block H;

else

Access denied;

end if

Performace analysis

In this section, we discuss the security and efficiency of the system.

Security analysis

If a user wants to access data successfully, he must meet the access control protocol and decrypt the data. In our scenario, we suppose that user never exposes his identity credential to others, and the key cannot be recovered by the adversary. We assume that A is a user, B is a ledger server, S is an authentication server and C is an adversary. K is defined as a public-private key, $E_{k_a^{-1}}(m)$ is defined as a signed message, $E_{k_a}(m)$ is a message encrypted by K_a. C(A) indicates that the C is disguised as A to send a message.

Attack on security protocols.

Table 1 Record of bread crumbs mechanism

Number	Department	Hash Value
I	Oncology	Null
II	Orthopaedics	c3929c7ea117fe3cce028b7b87491a99c ae014746c41d2b6436e1363490bbd25
III	Dentistry	239dd2fec70f90c8c84d235c9190a824 b7fff731bb4c2ff2467187d9b36f2ae2
...	…	…

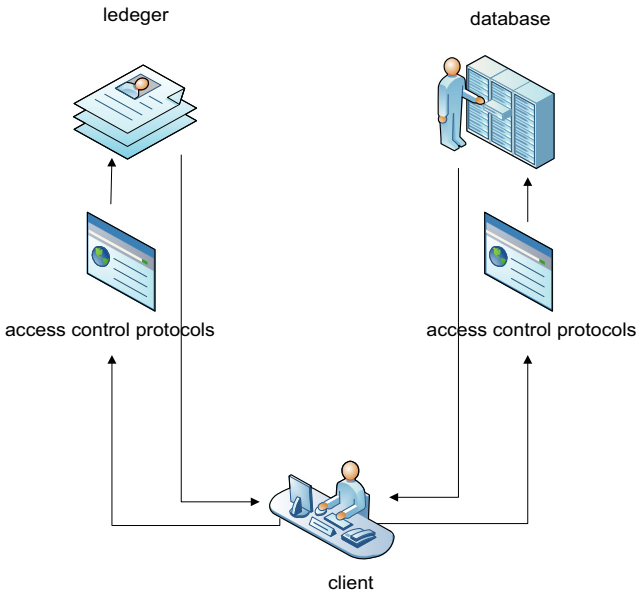

Fig. 5 User access to data flow schematic

Situation 1:

$$C \rightarrow B : E_{\left(K_c^{(-1)}\right)}\left(E_{K_b}(N_a, C)\right); \tag{2}$$

If the adversary without identity credential tries to access the ledger, he will not see the signature collection and encrypted summary. It is no benefit to the adversary.

Situation 2:

$$A \rightarrow C(B) : E_{\left(K_a^{(-1)}\right)}\left(E_{K_b}(N_a, A)\right); \tag{3}$$

$$C(A) \rightarrow B : E_{\left(K_a^{(-1)}\right)}\left(E_{K_b}(N_a, A)\right); \tag{4}$$

$$B \rightarrow C(A) : E_{\left(K_b^{(-1)}\right)}\left(E_{(K_a)}(M)\right); \tag{5}$$

We allow adversaries to intercept messages sent by users and perform replay attacks. And we assume that C can successfully deceived B,which leads B to regard C as A. Also, they can query the information on the blockchain. But as a result, they can only get a piece of encrypted information without decryption method.

$$A \rightarrow C(S) : A, B, N_a; \tag{6}$$

Table 2 List of system resistance attack

SCHEME	IDENTITY DISGUISE	REPLAY ATTACK	BINDING ATTACK	FORWARD SECURITY
MedBlock	Y	Y	Y	N

Table 3 Comparison between proposed system and other systems

SCHEME	TAMPER PROOF	ANONYMOUS	PRIVACY	ATTACK RESISTANCE
MedRec	Y	N	N	Y
Medshare	Y	Y	N	Y
DACC	Y	N	N	Y
MedBlock	Y	Y	Y	Y

$$C(A) \rightarrow S : A, C, N_a; \tag{7}$$

$$S \rightarrow C(A) : S, E_{k_S^{-1}}(S, A, N_a, C, K_c); \tag{8}$$

$$C(S) \rightarrow A : S, E_{K_S^{-1}}(S, A, N_a, C, K_c); \tag{9}$$

Assuming that the public key of B is K_b, and the public key of the attacker C is K_c. And the attacker wants the user A to believe that the public key of B is K_c, so that the binding attack is implemented. However, the returned information contains C identity information, A will find that the target is in consistent and avoid binding attacks.

Situation 3:

$$A \rightarrow C(B) : E_{\left(K_a^{(-1)}\right)}\left(E_{K_b}(N_a, A)\right); \tag{10}$$

$$C \rightarrow B : E_{\left(K_a^{(-1)}\right)}\left(E_{K_b}(N_a, A)\right); \tag{11}$$

$$B \rightarrow C(A) : E_{\left(K_b^{(-1)}\right)}\left(E_{(K_a)}(M)\right); \tag{12}$$

$$C(B) \rightarrow A : E_{\left(K_c^{(-1)}\right)}\left(E_{(K_a)}\left(M'\right)\right); \tag{13}$$

Even if the adversary wants to send false information to the patient, it will be judged as false information due to the lack of authentication information of ledger server.

Analyzing the overall security of the system, we can start with two parts.

A is a user, B is a ledger server, K is defined as a public-private key, m is defined as a signed message.

Authentication stage:

$$\frac{B\ \text{Received}\ m_1\ \ SignedWith\ K_a^{-1}, x\ in\ m_1, B\ IsTrustedOn\ K_a}{B\ CanProve(A\ \ Says\ \ x)} \tag{14}$$

Reception stage:

$$\frac{A\ \text{Receives}\ m_2 SignedWith\ K_b^{-1}; A\ CanProve(K\ \ Authenticates\ \ B)}{A\ \ CanProve(B\ \ Says\ \ m_2)} \tag{15}$$

$$\frac{A\ \ IsTrustedOnB; A\ \ Can\text{Prove}(B\ \ Says\ \ m_2); A\ \ Can\text{Prove}(B\ \ IsTrustedOn\ \ m_2)}{A\ \ Can\text{Trust}\ \ m_2} \quad (16)$$

The conclusion is that the system can resist identity disguise, replay attack, binding attack and so on (Table 2).

Attack on blockchain

Blockchain is the core of acquiring data and ensures integrity and reliability of information. It has tamper-proof and open class verification features to ensure that the information on the block cannot be tampered. Even if the adversary tampered with some of the ledger information, it would be quickly corrected by the system.

Adversary may try to submit a large number of requests to endorsers in order to cause network congestion. It's similar to denial of service attacks. It's pointless because endorser only handles requests from clients by checking the signature of the data. The cost is enormous but the effect is minimal.

Nodes of the system may also be attacked, crashed or even become adversaries. Consensus mechanism and endorsers' election mechanism can ensure the stability of the system so as to ensure that opponents will not cause great damage.

The access control mechanism on the ledger can realize the anonymity of data and achieve the same effect of ring signature and zero knowledge proof. This is a very effective way to protect the privacy of patients.

Table 3 below compares our MedBlock system to other existing systems. The result shows that proposed scheme is outstanding in privacy and security.

Efficiency analysis

The efficiency of the system is mainly reflected by three aspects.

In our scheme, we adopt bread crumbs to enhance information retrieval efficiency. If a user wants to retrieve some specific information, he can directly find the corresponding block according to the records of bread crumbs. The original search method needs to traverse the data on the block until finding the useful data. Although the bread crumbs will bring additional amount of data, compared to the traditional way of data retrieval, its efficiency increases too much. We compare our scheme with some new schemes such as MedRec [15] and Medshare [16]. With the number of access increases, MedRec uses less time. The results show that the efficiency of data retrieval is greatly improved (Table 4 and Fig. 6).

When the number of users is small, the effective date of each user accounts for a relative high proportion of all data. The original search method can also quickly find relevant information. However, as the number of users' increases, the advantages of MedBlock over the original methods become more and more obvious. The bread crumb records can directly guide the users to find the corresponding blocks. Even if the proportion of valid information is low, it will not be a constraint on efficiency.

When an endorser sends a proposal to add blocks to the system, we change the method from real-time upload to alternate upload. The time that patients visit hospitals is relatively concentrated. If an endorser chooses to upload the data in real time, the system will bear significant high load, which may cause data congestion. Avoiding this situation is helpful to improve the stability of the system. Through the comparison we can easily find the improvements of our scheme in this aspect (Table 5 and Fig. 7).

Uploading data asynchronously makes the system load smoother and helps to avoid data congestion in the system. The analysis and simulation results show that the scheme is effective to avoid significant high load that may cause data congestion.

Table 4 Latency of service provider requests

NUMBER	LATENCY(SEC)	
	MEDBLOCK	MEDSHARE
5	83.4	53.4
10	122.51	145.26
15	146.78	178.24
20	188.75	226.78
30	304.51	351.36
40	374.73	447.94
50	459.31	553.81
100	925.12	1286.73

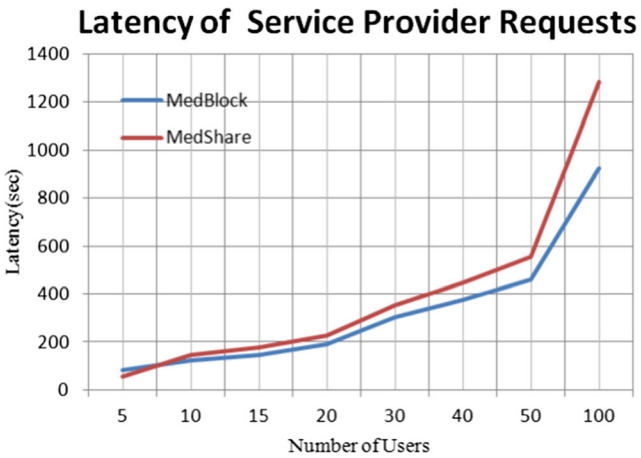

Fig. 6 Comparison of data delay

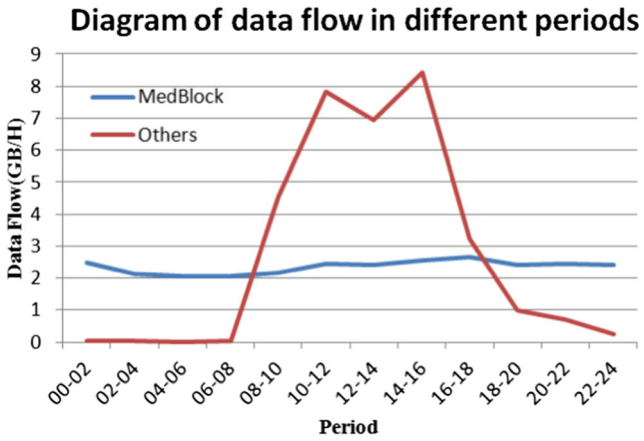

Fig. 7 Diagram of data flow in different periods

Compared with CP-ABE-based access control, our strategy is more appropriate. We analyze the reasons as the following aspects: First, the overhead of revocation in ABE scheme is too large to be ignored. In addition to the need to perform cryptographic operations, all the ledgers need to be changed when new patches need to be added. However, these problems in our strategy do not exist. Second, after the users get the encrypted information, they only need one exponential operation to decrypt ciphertext that doesn't contain encrypted information about attributes. It is a kind of efficiency improvement for users to obtain information.

Conclusion

Leveraging blockchain technology, MedBlock successfully resolved the problem of large-scale data management and sharing in an EMR system. Patients can easily access the EMRs of different hospitals through the MedBlock avoiding the previous medical data being segmented into different databases. Data sharing and collaboration via blockchain can help hospitals get a prior understanding of patients' medical history before the consultation.

We propose an efficient privacy-preserving and sharing scheme based on blockchain, which can guarantee users' privacy contained in his data by utilizing the combination of access control protocol and encryption technology. This method of uploading no data to a semi-trusted third party ensures that other agencies cannot access the original medical data of patients. The design which employs bread crumbs on the ledger can quickly retrieve the location of encrypted information and improves the efficiency of system.

Table 5 Diagram of data flow in different periods

PERIODS	DATA FLOW(GB/H)	
	MEDBLOCK	OTHERS
00–02	2.478	0.041
02–04	2.122	0.021
04–06	2.045	0.017
06–08	2.045	0.022
08–10	2.145	4.545
10–12	2.437	7.822
12–14	2.411	6.953
14–16	2.542	8.421
16–18	2.642	3.211
18–20	2.423	0.978
20–22	2.437	0.721
22–24	2.425	0.245

Funding This study was funded by the National Key R&D Program of China (No. 2017YFB0802300), the National Natural Science Foundation of China (No. 61772403 and No. U1401251), Natural Science Basic Research Plan in Shaanxi Province of China (No. 2017JM6004), and National 111 Program of China B16037 and B08038.

Compliance with Ethical Standards

Conflict of Interest Kai Fan declares that he has no conflict of interest. Shangyang Wang declares that he has no conflict of interest. Yanhui Ren declares that he has no conflict of interest. Hui Li declares that he has no conflict of interest. Yintang Yang declares that he has no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Perera, G., Holbrook, A., Lehana, T. et al., Views on health information sharing using electronic medical records. *Int. J. Med. Inform.* 80, 2011. https://doi.org/10.1016/j.ijmedinf.2010.11.005.
2. Kish, L. J., and Topol, E. J., Unpatients– why patients should own their medical data. Nat. Biotechnol. 33(9):921–924, 2015. https://doi.org/10.1038/nbt.3340.
3. Wang, Y., Li, P.-F. et al., A shared decision-making system for diabetes medication choice. IEEE Journal of Biomedical and Health Informatics. 21(5):1280–1287, 2017. https://doi.org/10.1109/JBHI.2016.2614991.
4. Lee, S. J., Larson, E. B., Dublin, S., Walker, R. L., Marcum, Z., and Barnes, D. E., Electronic medical record (EMR) predictors of undiagnosed dementia. Alzheimer's and Dementia. 13(7):1040–1041, 2017. https://doi.org/10.1016/j.jalz.2017.06.1469.
5. Thilakanathan, D., Chen, S., Nepal, S., Calvo, R. A., Liu, D., and Zic, J., Secure multiparty data sharing in the cloud using hardware-based TPM devices. In: Proc. IEEE 7th Int. Conf. on Cloud Comput. (CLOUD), pp. 224–231, 2014. https://doi.org/10.1109/CLOUD.2014.39.
6. Khan, A. N., Kiah, M. L. M., Ali, M., Madani, S. A., Khan, A. U. R., and Shamshirband, S., BSS: Block-based sharing scheme for secure data storage services in mobile cloud environment. J. Super Comput. 70(2):946–976, 2014. Springer US. https://doi.org/10.1007/s11227-014-1269-8.
7. Jena, D., Mishra, B., et al. Securing Files in the Cloud. Presented at 2016 IEEE International Conference on, 2016. 10.1109/

CCEM.2016.016. Available: http://ieeexplore.ieee.org/document/7819669/
8. O'Driscoll, A., Daugelaite, J., and Sleator, R. D., 'Big data', Hadoop and cloud computing in genomics. J. Biomed. Inform. 46(5):774–781, 2013. https://doi.org/10.1016/j.jbi.2013.07.001.
9. The Economist Intelligence Unit of IBM Institute for Business Value. Healthcare rallies for Blockchains: Keeping patients at the center. Healthcare and Blockchain Executive Report. 2017. Available: http://www.ibm.biz/blockchainhealth.
10. Fan, K., Ren, Y., Wang, Y., Li, H., and Yang, Y., Blockchain-based efficient privacy preserving and data sharing scheme of content-centric network in 5G. *IET Commun.* 12(5):527–532, 2018. https://doi.org/10.1049/iet-com.2017.0619.
11. Shen, Z., Shu, J., and Xue, W., Keyword search with access control over encrypted cloud data. IEEE Sensors J. 17(3):858–868, 2016. https://doi.org/10.1109/JSEN.2016.2634018.
12. Liu, Z., Li, T. et al., Verifiable searchable encryption with aggregate keys for data sharing system. Futur. Gener. Comput. Syst. 78:778–788, 2017. https://doi.org/10.1016/j.future.2017.02.024.
13. Kim, K., and Zhang, F., ID-based blind signature and ring signature from pairings. International Conference on the Theory & Application., 2002. https://doi.org/10.1007/3-540-36178-2_33.
14. Salazar, J. L., Tornos, J. L., and Piles, J. J., Efficient ways of prime number generation for ring signatures. Information Security, IET. 10, 2016. https://doi.org/10.1049/iet-ifs.2014.0547.
15. Hardjono, T., and Smith, N., Cloud-based commissioning of constrained devices using permissioned blockchains. In: Proc. 2nd ACM Int. WorkshopIoT Privacy, Trust, Secur. (IoTPTS), pp. 29–36, 2016. 10.1145/2899007.2899012
16. Zyskind, G., Nathan, O., and Pentland, A., Decentralizing privacy: Using blockchain to protect personal data. Proceedings of IEEE Security and Privacy Workshops:180–184, 2015. https://doi.org/10.1109/SPW.2015.27.
17. Lippman, A., Vieira, T., Ekblaw, A., Azaria, A., et al., MedRec: Using blockchain for medical data. Presented at *International Conference on Open & Big Data*. 2016. Available: http://ieeexplore.ieee.org/document/7573685/
18. Xia, Q., Sifah, E. B. et al., MeDShare: Trust-Less Medical Data Sharing via Blockchain. IEEE Access. 5, 2017. https://doi.org/10.1109/ACCESS.2017.2730843.
19. Esposito, C., Santis, A. D. et al., Blockchain: A panacea for healthcare cloud-based data security and privacy? IEEE Cloud Computing. 5(1):31–37, 2018. https://doi.org/10.1109/MCC.2018.011791712.
20. Li, M., Yu, S., Zheng, Y., Ren, K., and Lou, W., Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption. IEEE Transactions on Parallel & Distributed Systems 24(1):131–143, 2013. https://doi.org/10.1109/TPDS.2012.97.
21. Li, W.-M., Li, X.-L. et al., Flexible CP-ABE Based Access Controlin in Hybrid Cloud System. J. Comput. Sci. Technol. 32, 2017. https://doi.org/10.1007/s11390-017-1776-1.
22. Goyal, V., Pandey, O. et al., Attribute-based encryption for fine grained access control of encrypted data. In: Proc. 13th ACM conf. on Computer and communications security. pp. 89–98, 2006. https://doi.org/10.1145/1180405.1180418.
23. Gu, K., Jia, W., Wang, G., and Wen, S., Efficient and secure attribute-based signature for monotone predicates. Acta Informatica 54(5):521–541, 2017. Springer Berlin Heidelberg. https://doi.org/10.1007/s00236-016-0270-5.
24. Guo, R., Shi, H., Zhao, Q., and Zheng, D., Secure attribute-based signature scheme with multiple authorities for Blockchain in electronic health records systems. IEEE, 2018. https://doi.org/10.1109/ACCESS.2018.2801266.
25. Ferdous, S., Margheri, A., Paci, F., and Sassone, V., Decentralized runtime monitoring for access control systems in cloud federations. *Proc. IEEE Int. Conf. Distrib. Comput.*:1–11, 2017. https://doi.org/10.1109/ICDCS.2017.178.