Journal of Computers Vol. 33 No. 6, December 2022, pp. 73-82
doi: 10.53106/199115992022123306006

A Private Storage System of Medical Data Based on Blockchain

Shengjuan Guo, Mei Yang*

School of Primary Education, Wuhan City Polytechnic, Wuhan, China.
402596565@qq.com, 27778676@qq.com

Received 25 February 25 2022; Revised 9 March 2022; Accepted 10 March 2022

Abstract. At present, most medical institutions use electronic medical records (EMRs) to save medical and health data in the database, which take certain security risks on sharing the medical and health data. To solve this problem, this paper presents a novel storage method for health data using blockchain technology, which can share and avoid tampering the medical data between different hospitals and patients and keep the privacy at the same time. The schema of health data structure, the generation of block, the data access, peer management, system design and access controlling are presented in this paper.

Keywords: blockchain, medical data, information security, information sharing

1 Introduction

With medical field entering the era of digital and big data, it is hard to balance the personal privacy and the applications of medical and health data [1]. Usually the medical data are stored in electronic medical records (EMRs), which is not easy to share with patients and take the risk of data leakage and tampering the privacy data in some degree [2-3]. To avoid this problem, Esposito et al. introduced blockchain technology to medical cloud storage, which ensure the accuracy of data content and protect personal privacy security through consensus mechanism and data fuzzy encryption [4]. Azaria et al. presented a MedRec information sharing platform based on Ethereum technique [5], in which patients have the control of their personal medical records. Hospitals and other medical service providers record on the public chain, and each record will be given a specific digital currency as feedback. Xue et al. [6] proposed the MDSM Blockchain Medical System by improving DPOS consensus algorithm. Based on the practical Byzantine fault-tolerant algorithm, Zhang et al. [7] proposed an alliance block Chain Medical system to store medical data and ensure the security using role authorization.

However, in the above solutions, the patients difficultly hold and exchange the data with other hospitals and makes Information Silo effect. As a result, the patients can't share the personal medical data to other medical centers. When medical disputes arise, medical institutions can easily tamper with the patient medical records while patients can not find evidence easily.

To solve this problem, this paper aims to design a novel scheme to share the medical data and patient privacy using blockchain Technology, which can avoid the medical institution's modification on the medical records without the patient's agreement. Blockchain is a novel technology which includes distributed medical data storage, blocks in China, point-to-point data exchange, consensus algorithm, security technology and information applications. Based on blockchain, the health data storage is introduced in section 2, and peer management and system design are introduced in section 3, and the conclusion is in section 4.

2 Health Data Storage

Electronic Medical Records (EMR) are basic data for the medical information system, which include much patients privacy data such as the patients' personal information, medical reports, doctors suggestion, medical Images such as CT/MRI images, ultrasound images and electrocardiogram. The data is increased day and day, and how to design a schema to store these information efficiently and safely has become more and more important. In this section, a schema based on block chain is designed to storage the patient information. Private chain, consortium chain and public chain are the three main type of blockchain. In the private chain, a company or individual has absolute control of the blockchain. For the consortium chain, it is managed by the members of the authorized organizations, and the public chain is completely decentralized and anyone in the chain can participate in the transaction record and message query. In our schema, the block chain is controlled by the medical institutions such as hospitals.

* Corresponding Author

2.1 Health Data Structure

Transaction list is the basic structure and storage core of the entire medical blockchain, which store patient information, doctor information, hospital information, treatment information and permission information et al. The transaction list mainly includes transaction ID, transaction type, transaction content and transaction generation timestamp, public key of patients, public key of doctors and digital signature. Fig. 1 shows the components of transaction list.

Transaction ID is the result of hashing the transaction content and type of the whole transaction using SHA-256 or Base64. The transaction ID usually generate Merkle root and is used for quick retrieval. The transaction list type is mainly used to describe the information recorded in the exchange, including patient information, doctor information, medical information, health information, permission information, query information, modification information and deletion information. The timestamp is the time when the transaction was generated. When the patient and the doctor query the information transaction list, the patient's public key and the doctor's public key are not required Meanwhile. In the medical information transaction list, both the patient's public key and the doctor's public key are required, so that the patient belonging to the medical information transaction form can be quickly located. After the doctor encrypts the information with private key, it is convenient to verify whether the information has been tampered. Digital signature is the signature of the private key corresponding to the public key.

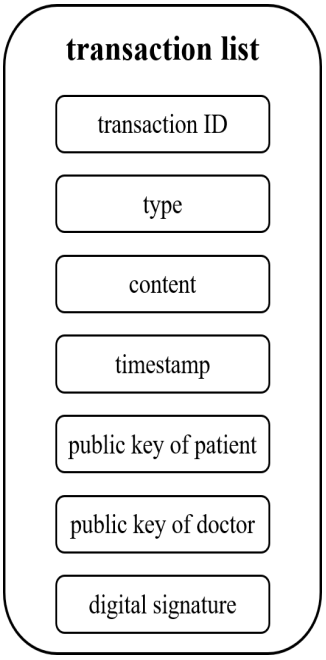

Fig. 1. Transaction list

2.2 The Generation of Blocks

A medical blockchain is composed of blocks that record the hash value of the previous block, while each block is composed of block head and block body. The block head contains the hash value of the current block, the hash value of the previous block, the size of the current block, the Merkle root and the timestamp. The block body is composed of the Merkle tree with the transaction order as the leaf node and the number of transaction orders contained in the current block. The specific structure of the block is shown in Fig. 2.

The current hash value is generated by the pre-hash value, block size, timestamp and Merkle root. Due to the irreversibility of the hash algorithm, the generated hash value has little probability of repetition in theory, and can be used to identify a block. The pre-hash value stores the block header hash value of the previous block, which logically forms a chain. The Merkle root is generated from the transaction orders contained in the current block. The Merkle tree is a binary tree structure and can quickly locate which transaction order has been tampered when data is forged using time complexity of verification is $O(\log_{f0}(n))$.

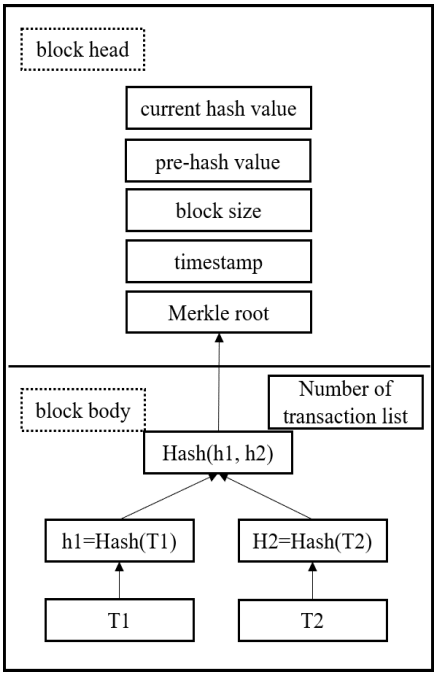

Fig. 2. Block structures

2.3 Data Share Schema and Retrieval

According to the basic framework and data standard of electronic medical records, EMRs mainly consist of medical record, clinical diagnosis, treatment records, inpatient diagnosis and treatment records, health examination records, referral records, Statutory medical certificates and hospital information, which are numbered from EMR01 to EMR16 [8].

Structured data is generally numbers and simple words, with a specific structure, relatively small amount of data, suitable for direct storage in the blockchain using a JSON string.

Unstructured data is generally large text, audio, images, such as B ultrasound, ECG. This kind of data has no specific structure, and the amount of data is relatively large, which is not suitable for direct storage in the blockchain. In order to achieve the security and sharing of data, the original file is stored in file system and the file paths are stored in the distributed database system (DDBS). The original file, the path and database records are hashed and store in the blockchain. When querying the data, the database record, file path and the original files are retrieved and generate the hash values which compare with the hash value stored in the system to verify the integrity of the file and prevent the lack and tampering of data as Fig. 3 shows.

To improve the efficiency of query, Hyperledger Fabric blockchain system are used to store the medical data, which supports state database, block index database and historical index database. The state database is used to record the number of data records of each user in the blockchain, the block index database stores the location information of each block, the historical index database is used to record the historical status information, which is easy to locate when the data is modified or deleted. Key-Value databases such as LevelDB and CouchDB is used to save transactions and blocks in the blockchain system.

As shown in Fig. 4, when a new health record is added to the blockchain, the status information is inserted into the status database, and the block number, offset and other index information of the record are insert into the block index database. When querying the data, the total amount related to the certain user is searched from status database, and next find the location information from block index database, and then obtain the data from the blockchain quickly. In the historical database, the modified record can be searched and inspected.

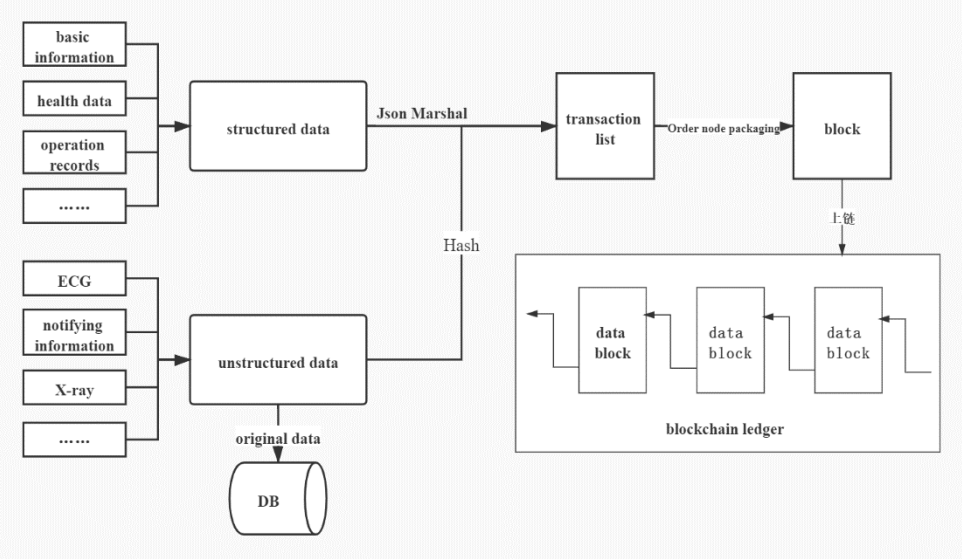

Fig. 3. Storage diagram for structured and unstructured data

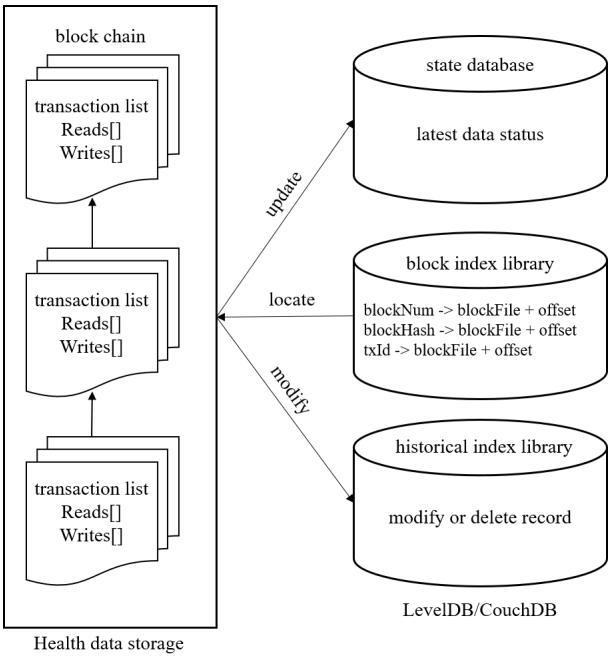

Fig. 4. Schematic of data query

3 System Architecture

3.1 The Schema Architecture

This proposed blockchain EMR system consists of patients, doctors, hospital manager and governors. The patients and doctors access the data in private cloud servers such as hospital server centers. the patients and administrators access the data in public cloud servers. The deployment server's schema is show in Fig. 5. The patient's and the governor's SDK tools are deployed in the public cloud, and doctor and hospital SDK tools are deployed in hospital private cloud. The schema servers and blockchain servers can be deployed in hospital private cloud or public cloud.

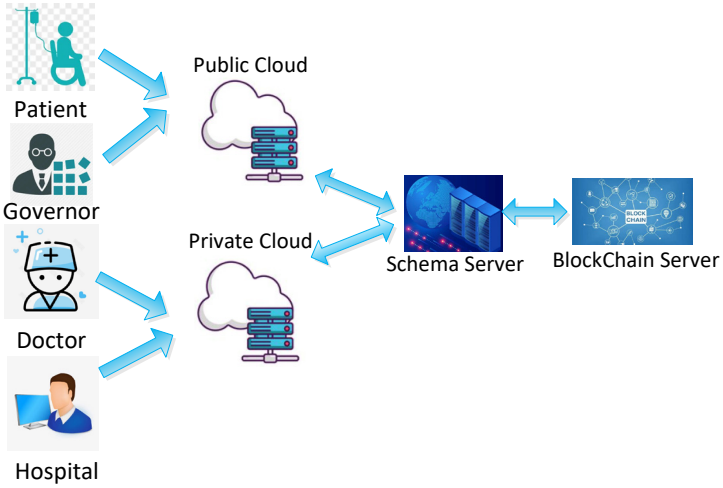

Fig. 5. The deployment schema

This schema is based on Hyperledger Fabric blockchain which is composed of multiple organizations such as hospitals and other medical institutions. The schema includes three layers of application, block chain and data storage. The application layer can operate the information in block chain layers and the medical information in data storage layers. The total Schema is shown in Fig 6.

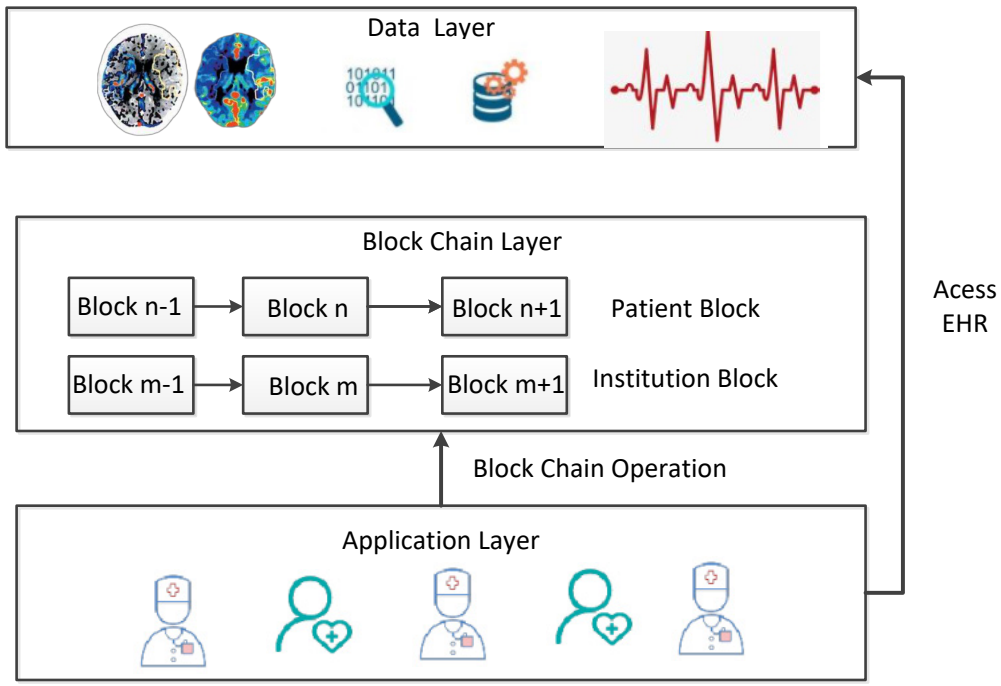

Fig. 6. The system architecture

An organization can consist of one or more nodes (PEERS), each of which contains the following four node types:

Leading peers: communicate with the Order sorting service cluster node on behalf of the current organization, the sorted and bundled data from the Order sorting service is retrieved from the cluster node and broadcast to the other nodes in the organization.

Anchor peers: undertake the task of communication with other organization nodes.

Endorsing peers: invoke different chain codes according to the instructions, simulate the results after execu-

tion, and endorse the simulation results.

Committing peers: insert the validated blocks to the ledger.

Nodes in blockchain system can collect transactions, verify the legitimacy of transactions and blocks. Similarly, endorsing peers and sorting service node have the same function in this system. The client and data storage process will be introduced in detail.

3.2 The Doctor Client and Patient Client

The client can be used to interact with the Fabric network for data increase and query, such as patient and doctor information entry, visit record entry, visit record query, etc. There are two types of clients: doctor client for data operation and patient client for authorized operation. With the user's authorization, the doctor client can operate the patient's medical data including creating, updating, deleting and retrieving the patient's information. Patient client can upload, modify, delete and lookup their personal information, health records through smart devices.

3.3 Data Operation Process

When the transaction is created, the client should start the data operation process, which includes verification, packaging, and up-linking. The PBFT is adopted as consensus algorithm due to the advantages of low energy consumption and large throughput as Fig. 7 shows.

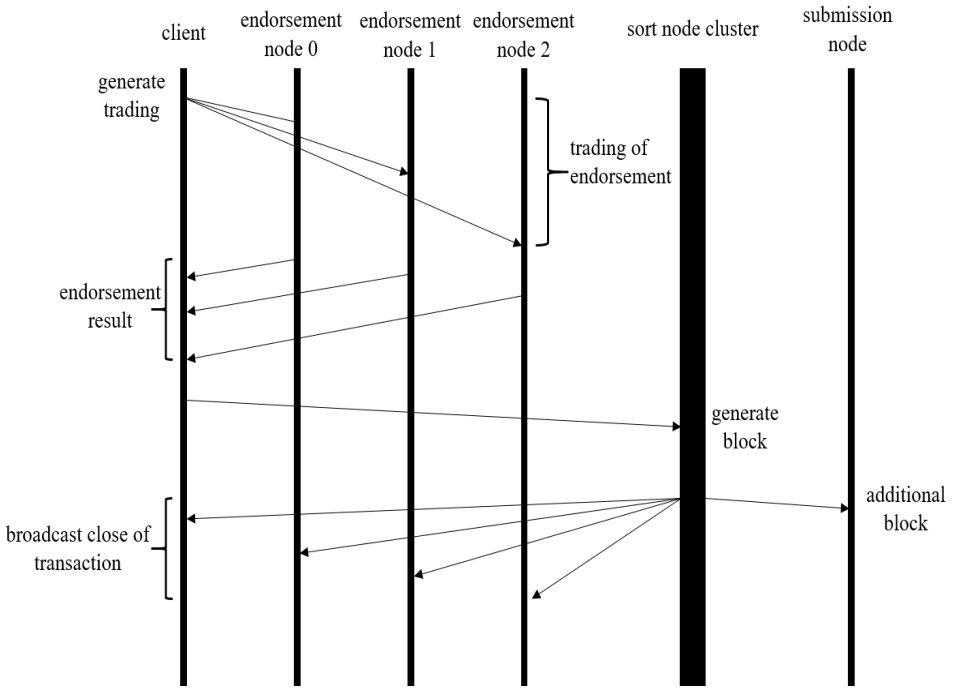

Fig. 7. Data operation process

The algorithm to operate the data is in Algorithm 1.

Algorithm 1. The data operation process

Input:

The ID of the Patient
The operation Type of EMR
The index of EMR

Output:

The result of Operation.

Procedures:

Step 1: client SDK generates a medical health transaction and confirm whether the content needs to be changed on the blockchain ledger.

Step 2: endorsing peers verify the signature and decide whether the submitter has the authority permission.

Step 3: endorsing peers return the result with the read-write set.

Step 4: the client verifies the signatures from each endorsing peer, confirms endorsement, and verifies the differences between different endorsing peers. If passing, the client sends the read-write set to the sorting service cluster.

Step 5: the ordering service node cluster continuously receives medical and health transactions from the network, generating healthcare blocks in chronological order of the transactions.

Step 6: healthcare blocks are broadcast to all nodes in the channel, committing verification to ensure that the endorsement strategy is satisfied and the state in the ledger has not changed.

Step 7: The nodes in each channel add healthcare blocks to the chain as storing to the state database, and then notify the client whether the transaction is inserted to the blockchain.

4. Smart Contract and User Management

In fabric chain, the smart contract depends on the business chaincode running in the network. Users can operate the chain code through the client SDK or command line to carry out the necessary operations to modify the status of blockchain distributed ledger.

4.1 The Chain-code life-cycle

The chain code updates the data in the ledger of the distributed database through the node. Users can modify various states in the life cycle of chain code through commands. As shown in Fig. 8, the life cycle includes initial state, waiting for installation, waiting for instantiation, running and stopping.

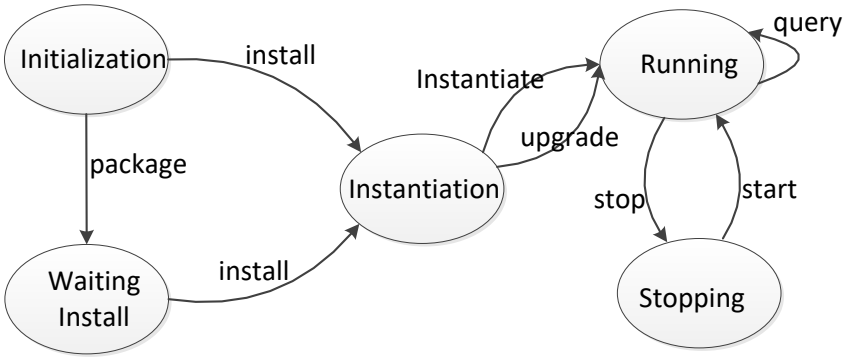

Fig. 8. The life cycle of chain code

4.2 Chain Code Implementation

From the block chain view, the most important operation is getting and setting data between the application layer and block chain layer. To access the data in block chain, two interface, *GetState* and *PutState*, are developed

to inquiry and reset the data from ledger. The algorithm *GetState* and *PutState* are shown in Algorithm 2 and Algorithm 3.

Algorithm 2. The operation of *GetState*

Input: the array args of ledger
Return: the status of operation
Procedures:

```
Function createEMR(stub shim.ChaincodeStubIntergace, args []string)(string, error)
if(len(args) != 2){
   return   fmt.Errorf("Parameter Error")
}
value := []byte(args[1])
err := stub.PutState(args[0], value)
if(err != nil){
   return "", fmt.Errorf("Writing Error:%s", args[0])
}
return args[1], nil
```

Algorithm 3. The operation of *PutState*

Input: the index of medical data
Return: the medical data
Procedures:

```
Function getEMR(stub shim.ChaincodeStubIntergace, args []string)(string, error)
 if(len(args) != 1){
    return  fmt.Errorf("Parameter Errors")
 }
 value, err := stub.GetState(args[0])
 if err != nil {
    return "", fmt.Errorf("Quering data Error, Error ID：%s", err)
 }
 if value == nil{
    return "", fmt.Errorf("No data in ledger, %s", args[0])
 }
 return string(args[0]), nil
```

4.3 User Authority Management

The system adopts a role-based access control (RBAC) model for authority management. Different permissions of roles are also different, which improves the security of the system and realizes different needs according to different roles. In this system, the main users are patients, governors, doctors and hospital managers. Account management is mainly used for all kinds of users in the system. Different roles have different permissions, which improves the security of the system and realizes different requirements.

Key pair is generated when the user registers in Hyperledger Fabric, and users can use the CA client or SDK to request the generation of certificates and pairs, the algorithm for generating the key pair is the elliptic curve digital signature algorithm (ECDSA). All components and members in Hyperledger Fabric must have certificates to be eligible to enter the network. Fabric provides Fabric CA components to quickly build certification authorities.

As shown in Fig. 9, Fabric CA is a typical C/S structure, Servers are divided into root servers and intermediate servers. The certificate of the root server can be issued to the intermediate server, and the certificate of the intermediate server can also be issued to other intermediate servers. The Fabric CA client is used to send certificate issuance and account management requests to the server, including enroll, register and revoke [9]. Server authenticates by invoking SDK and publishing the certificate to the node.

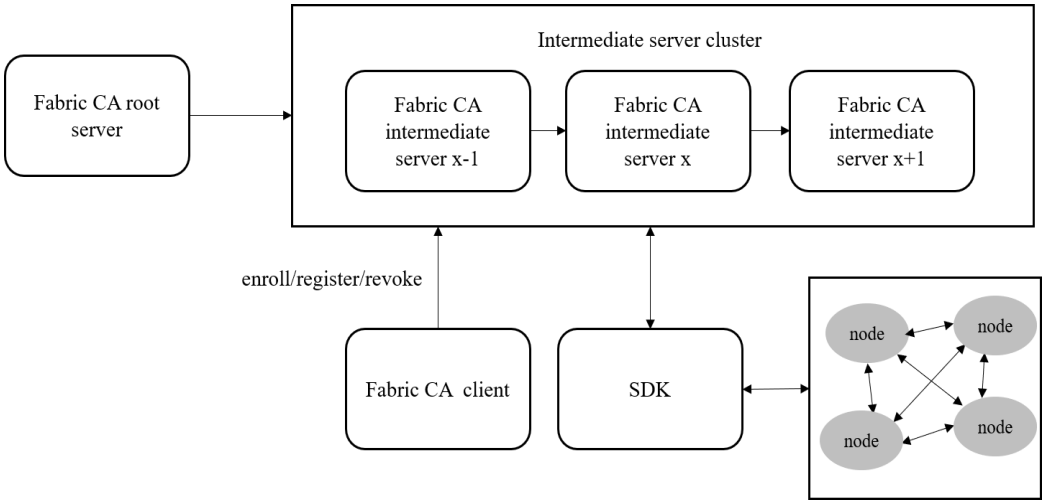

Fig. 9. Fabric CA certification structures

When patients or nodes in the organization need to join the network, they first fill in their application information through the client and apply for a certificate from CA. After verifying the applicant's application information, the CA generates a key pair and certificate, and then sends the key pair and certificate to the applicant. After the applicant receives the key, the key is encrypted and stored. When sending a message in the network, you need to attach your own certificate and public key, so that the message receiver can verify the validity of the message sender, and a node can join the network. The authentication service provided in fabric enables only the authenticated nodes to participate in the network. Although it has a certain centralization, it reduces the number of malicious nodes and further improves the security of the network.

4.4 Security Evaluation

Non-tamperability and privacy are two main features of information security. In the view of tampering, the traditional centralized data storage is difficult to guarantee. Hackers can modify the data of the database through SQL injection or maliciously modify the value of the database after obtaining permission. In the block chain system, the data is not stored in a centralized center, but distributed in each node in the network. When a node wants to modify the value stored in a block chain, other nodes can easily verify the forgery with their own backup. At the same time, due to the special structure of the block chain, when tampering with a block, it is necessary to modify not only the current block, but also the blocks behind the current block which make the tampering easy to detect. In terms of privacy security, the data stored in the block chain is encrypted, and the ownership of the data is the patient. Only the owner of the information can decrypt. Even if the data is leaked, the patient's information cannot be obtained without decryption key. And the role-based access control is introduced into this system. Only when the patient is authorized, other people can retrieve the patient's data, which further strengthens the patient's data privacy. Block chain can record the behavior of nodes. When a node conducts malicious behavior, such as tampering with patient information, illegally obtaining patient information, etc., these behaviors will be recorded in the block chain and cannot be denied. When the system finds a malicious node, it will revoke the certificate of the node and expel it from the blockchain network.

5 Conclusions

This paper analyzes the data security and sharing issues of traditional medical record systems and proposes a novel schema using blockchain which can be decentralized, traceable, and immutable. In the presented schema, the patient data is secured, shareable, and authorized by the patient's control. Data stored on the blockchain is encrypted and only the owner of the data has the right to query, which can make the data secure. The realization schema is also introduced in this paper, which can share data between different hospitals, thereby reducing repetitive examinations and giving more reasonable diagnosis and treatment plans.

References

[1] H. Gu, C. Jia, D. Wu, S. Gu, Researches on the Personal Privacy under the Background of Health Care Big Data Application, Chinese Health Service Management 38 (2)(2021) 117-120.
[2] H. Kristiina, S. Kaija, N. Pirkko, Definition, structure, content, use and impacts of electronic health records: a review of the research literature, International journal of medical informatics 77(5)(2008).
[3] Y. Yuan, F. Wang, Blockchain: The State of the Art and Future Trends, Acta Automatica Sinica 42(4)(2016) 481-494.
[4] C. Esposito, A. De Santis, G. Tortora, H. Chang, K.K. R. Choo, Blockchain: A Panacea for Healthcare Cloud-Based Data Security and Privacy? IEEE Cloud Computing (5)(1)(2018) 31-37.
[5] A. Azaria, A. Ekblaw, T. Vieira, A. Lippman, MedRec: Using Blockchain for Medical Data Access and Permission Management, in: Proc. 2016 2nd International Conference on Open and Big Data (OBD), 2016.
[6] T. Xue, C. Fu, C. Wang, X. Wang, A Medical Data Sharing Model via Blockchain, Acta Automatica Sinica 43(9)(2017) 1555-1562.
[7] C. Zhang, Q. Li, Z. Chen, Z. Li, Z. Zhang, Medical Chain: Alliance Medical Blockchain System, Acta Automatica Sinica 45(8)(2019) 1495-1510.
[8] Q. Shao, C. Jin, Z. Zhang, W. Qian, A. Zhou, Blockchain: Architecture and Research Progress, Chinese Journal of Computers 41(5)(2018) 969-988.
[9]P. Sharma, M. Borah, S. Namasudra, Improving security of medical big data by using Blockchain technology, Computers & Electrical Engineering 96(2021) 107529.