2019 Twelfth International Conference on Ubi-Media Computing (Ubi-Media)

An Electronic Medical Record Management System based on Smart Contracts

Yeong-Sheng Chen, Wei-Kai Yang, and Jie-Si Chen
Department of Computer Science, National Taipei University of Education, Taipei, Taiwan
yschen@tea.ntue.edu.tw, kevin05190519@gmail.com, jessyechen@gmail.com

***Abstract*—This study proposed a management system of the electronic medical records in the blockchain environment. In the proposed system, electronic medical records are first stored in the InterPlanetary File System (IPFS), and then the system generates the hash value, which will be sent to the smart contract to correlate with the patients' data. After that, if the medical staff's medical authority is confirmed by the system, the smart contract can send back the hash value that IPFS generated, and thus the system will present the complete electronic medical records. In the experimental simulation, the results showed that our proposed electronic medical record management and storage process not only can block forged or tampered electronic medical records via smart contracts, but also make it easy to integrate electronic medical records and share medical resources.**

Keywords—Blockchain, Smart Contracts, Electronic Medical Records

I. INTRODUCTION

With the development of Internet technology and the electronization of medical information, many medical institutions are using electronic medical records, which can facilitate the process of accessing medical records. However, there exist the problems of tampering and counterfeiting electronic medical records. Therefore, it is important to plan a management system for the electronic medical records, and how to thoroughly and accurately access the electronic medical records has become a topic worth discussing.

Blockchain [1][2] technology has received much attention in recent years. Virtual currency and smart contract applications have gained more and more attention not only in the rapidly growing financial technology but also in medical applications [3][4][5]. The untamperable nature of the blockchain is the key to the technology, which ensures the authenticity and traceability of the data while maintaining their transparency and anonymity. Smart Contract is the main core of Ethereum blockchain [6]. It can perform decentralized transactions and stylized contracts without relying on other third-party systems or databases.

Azaria *et al.* [5] propose a framework to limit and manage medical institutions' access to medical records via smart contracts. Based on such a framework, not only can medical records retain full anonymity, but the permissions of medical records by smart contracts can also be well managed. Inspection agencies can also review visiting records through blockchain. Because of these above advantages, this research is aimed at establishing a system to optimize its framework and improve the process by adding different technological skills in the system.

In the current electronic medical record exchange system, most hospitals or clinics do not retain the electronic medical records of all patients. After authenticating the agreement between medical staff's IC cards and the patients' health insurance IC cards, Electronic Medical Record Exchange Center (ECC) will retrieve the patients' medical records which are stored in other hospital databases [10]. The high costs of electronic medical record review process in the construction of the framework and the cumbersome access of the electronic medical record lead to the difficulty in interoperating the medical records and even raise the suspicion of tampering, which may cause false diagnosis and medical disputes.

To solve the problems mentioned above, this study uses smart contracts and InterPlanetary File System (IPFS) [7]. We apply blockchain to patients' data and medical records' hash values because of its irreversible nature and manage medical personnel's authority through smart contracts. Compared with the MedRec framework [5], we focus on the part of reviewing and authorizing electronic medical records. MedRec uses three smart contracts, RC, PPR, and SC, to exchange transaction records; while we incorporate different functions, such as registration and authorization, into smart contracts. These functions we devise allow the authority of the medical records' review to transfer through the system instead of being limited to the medical staff in a single institution, and patients have a higher degree of mastery of their own medical records. Because IPFS has the advantages of distributed storage with high storage security [8][9], we chose to apply IPFS to storing related medical image files. In a similar past study [10], the medical records converted to Json or Xml files were stored in smart contracts so that they were not affected by database failures or attacks. However, thus, transaction fees for the smart contracts are high because it takes longer time to review the large files of medical records. In this study, we store the hash values obtained after uploading medical records to the IPFS in the smart contract. As a result, the medical records will have multiple copies in several blocks and can be verified by the hash values of the file content, which can protect the data from being maliciously tampered. Compared with the current electronic medical record exchange system, this study can avoid any loss from centralized system data.

The process designed in this paper makes the framework decentralized, the sharing of medical records more convenient, and the authentication of the medical records effectively managed. Last but not least, the full transparency of the data simplifies the process of accessing medical records. Also the advantages that we mentioned above can prevent medical records from being tampered or forged.

The content of this paper is divided into five sections. In the first section, we will introduce the background, motivation and purpose of this paper. In the second section, we will mention technology and documents that are related to this paper. In the third section, we will explain the structure of the system built in this paper. In the fourth section, we will detail how we construct our system. In the last section, we will make conclusions and further discuss the future possibilities.

978-1-7281-2820-7/19/$31.00 ©2019 IEEE
DOI 10.1109/Ubi-Media.2019.00050

II. RELATED WORKS

A. Blockchain

Blockchain is a technical solution for storing, verifying, transmitting and communicating network data based on cryptography. The core concept is to rely on cryptographic and mathematical decentralized algorithms. Blockchain can make participants reach a consensus without the intervention from the third party. This can solve the problem of lack of trust and unreliable value delivery. Through a public key, both of a private key, which are generated through asymmetric encryption, and the account address, we can perform a transaction or transmit data on the blockchain. The transaction on the blockchain is verified by every block on the blockchain instead of the third party.

After a block verifies the transaction addresses, transaction messages, and the source of cryptocurrency, it will push them to other nodes for verification. Once the transaction is confirmed by all nodes on the blockchain, the transaction data will be recorded in the block. The transaction record can't be changed and has full anonymity. Each transaction will produce a unique hash value, and each block header contains the previous block hash value, which connects all blocks and forms a chain[11].

Compared with the traditional financial transaction systems, which has to rely on trusted or guaranteed third-party organizations, the blockchain effectively implements the process of decentralization and combines the network to ensure that transactions are collectively maintained and verified by each node and that transaction records are confirmed.

B. Ethereum

Ethereum is a public blockchain with the functionality of smart contracts, which have an open source platform. In addition to the usual cryptocurrency transactions, Ethereum features a decentralized Ethereum Virtual Machine (EVM) to deploy and apply smart contract. In addition to mainly conducting cryptocurrency transactions, Ethereum also provides environmental constructions of private blockchains and multiple blockchains for testing, such as Rinkeby and Ropsten etc., which provide developers with different environments for testing and development.

C. Smart Contract

Ethereum provides a programmable computer trading protocol called Smart Contract, which is designed to allow both parties to programmatically trade and have open but anonymous records. Smart contracts can be written in Solidity, Serpent, etc.

Through computer programs, we can implement different transaction contents through virtual currency flow. The account addresses and data must be transmitted to the function in the smart contract through a standard Application Binary Interface (ABI). Smart contracts will then perform currency transaction or process data according to the functions. In this way, the transaction records can be stored and verified on the blockchain.

D. Interplanetary File System

InterPlanetary File System (IPFS) proposed by Juan Benet is a network transport protocol for the decentralized storage and the sharing of files. It is content-based addressing rather than domain-based addressing. The same data content will generate the same hash value. From the comparison of the hash value, we can see whether the data blocks are consistent. The node itself uses a version control system, which is similar to Git, to manage local files and data. This version control system guarantees the low redundancy of the data and provides a traceable historical version. The IPFS node needs to use the blockchain technology to maintain the hash routing table and the consistency of the books. This can not only dynamically increase or decrease the contents but also make the nodes reach a consensus with the whole network. The node can both pull the required data from other nodes and store the new data in its own node for other nodes to download.

E. MedRec

Azaria *et al.* [5] propose a model named MedRec, which uses Ethereum smart contract technology on electronic medical records. Through three smart contracts, medical records can be registered, the authority between patients and medical staff can be established, the records of authorization can be maintained, and the data in the existing medical institution database can be interrelated. After the authority is confirmed, a copy of the medical records can be obtained from medical institutions that originally stored the medical record. In other words, the medical record of the patient can be accessed to different medical institutions. The model not only makes medical records difficult to tamper with but also reduces the occurrence of repeated medical records. Medical institutions can obtain information across agencies and provide further medical services.

III. RESERCH METHOD

This study proposes a system to improve the current system structure of electronic medical records. We divide this problem into three aspects, the authority management, the storage, and the access. For the authority management, we propose using the smart contract to register the verification of the medical staff and the medical records. As for the storage, we propose applying IPFS to the storage of medical records. As for the access, through smart contracts we get access to the reading of the whole medical records.

In the system proposed in this study, each medical institution is regarded as a node. Each node acts as an Ethereum client, transfers data, and registers on the blockchain through the smart contract. All the nodes will also become a node in IPFS to store data and share files.

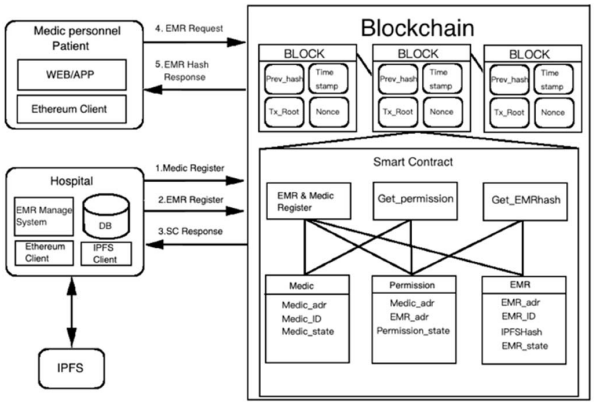

Fig. 1. System architecture.

The process of the proposed system mainly consists of three steps. In the first step, a medical staff member enters their personal information and register through the registration function in the smart contract. After our system confirms the registration status of the medical personnel's account address, we upload the patients' electronic medical records to IPFS. Then the hash value that IPFS generates from the patients' medical records is stored in the smart contract. When medical personnel or patients need to obtain an electronic medical record, they will have to make a request for the authorization of the smart contract. After the medical personnel gain the authorization, they will obtain the hash value of the patient's electronic medical record via the smart contract. The system will then present all the information, maps, etc. in the electronic medical record. The system architecture is shown as Fig. 1.

A. Authority Management

In terms of authority management, medical personnel register their Ethereum account address on the smart contract. The medical personnel use the information on the medical IC card to verify their identity. Therefore, the system transmits the medical personnel account address, ID and other information which are in the medical IC card to smart contracts so that we can identify medical personnel.

The medical personnel or other users must verify the identity of the uploader and the presence or absence of the medical record before uploading the IPFS. After confirming the result, a medical staff member or other users will register the status and content of the electronic file with the smart contract, such as medical record information and medical image files. In terms of access authorization, the uploaders, who are either medical personnel and the patients have the authority to access the electronic medical records, other registered medical staff members can also gain authorization access to medical records by using this system. In addition, patients can also place restrictions on the access of the medical records through smart contracts.

B. The Upload process for Medical records

When a new medical record needs to be archived, it needs to be uploaded to the medical record management system first, and then the medical personnel identity and related patient information are transmitted to the smart contract for verification. If the medical personnel have not registered or the data are incorrect, the upload will fail. If the identity of medical personnel is verified by the system and the medical record has not yet been uploaded, the medical record can be successfully uploaded to the IPFS. The returned file hash value from IPFS will then be added into the smart contract and the local database by the system. When the medical record needs to be edited, the system will first upload the already edited file to IPFS, and then add the newly returned file hash value to the smart contract to ensure that the medical record has been changed. Consequently, the medical institutions can get the correct information in real time.

C. Medical Record Acess

The medical personnel and patients who have obtained the authorization have the right to access the medical records. If the medical personnel and patients don't obtain the authorization, the system will transmit their account address to the smart contract. The smart contract will then determine their authorization of accessing the medical records according to the contents of the contract. After authorizing their access, the smart contract will acquire the hash value of the medical record generated by IPFS, and then the medical personnel can read the complete information related to the medical record through the IPFS gateway. Additionally, patients can also check their electronic medical records through the webpage or the application. The application or webpage will transmit the patients' account addresses to the smart contract. After being authorized by the smart contract, the IPFS hash value will be added to the smart contract. The complete medical record will be presented on the application or webpage through the IPFS gateway.

IV. SYSTEM CONSTRUCTION

In the system construction, we use ReactJS as the front end of the system, and Testrpc [12] is used as the blockchain environment for simulation. Besides, we use the online editor Remix Solidity IDE provided by Ethereum to write and test the smart contract. In the medical file upload section, we apply the IPFS API to uploading files to IPFS. The architecture of the proposed system is similar to that of MedRec. Both systems use smart contracts to register and obtain medical records, and utilize the Ethereum environment. However, in the proposed system, only hash values instead of the complete medical records are stored in the smart contracts. Comparison of the proposed approach with MedRec will be presented later.

A. Registration for Medical Personnel

By first concatenating the API provided by web3.js, we can query the status of Ethereum blockchain account, perform transactions, and access smart contracts through the web3 API. There is some information required for the registration of medical personnel, such as the account address, the name of the medical staff member, and the identity card number. Our system processes and stores the information through the web3 API and then send it to the registration function in the smart contract to register. Fig. 2 shows an example for registration on the webpage.

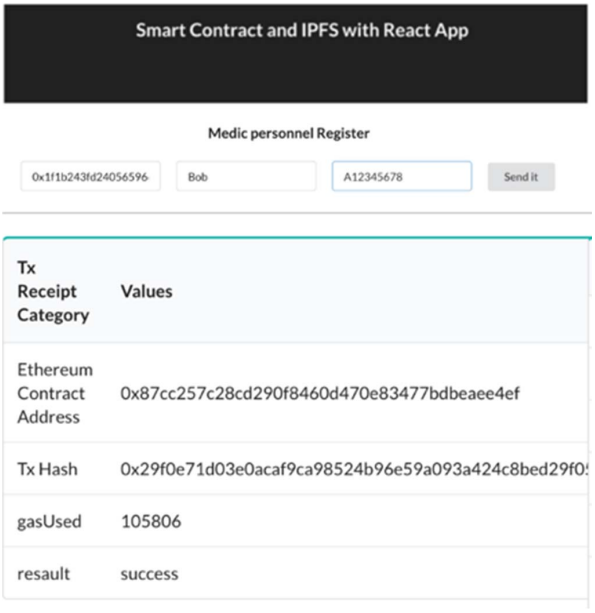

Fig. 2. Screenshot of registration of a medical staff member.

B. *The process of uploading medical records*

After the medical personnel complete the registration, they can upload the medical records. First of all, they can select the medical image files or related files that they want to upload, and then the system will confirm their authority. Next, the medical files will be sent to the IPFS through API. Our system will return a hash value that is generated from the medical record, enter patients' information, and save the hash value in the smart contract. Therefore, there is only the hash value of the medical record in the smart contract instead of the complete medical record. In this way, we can reduce the transaction fees for each transaction. Fig. 3 shows an example of uploading medical records to IPFS.

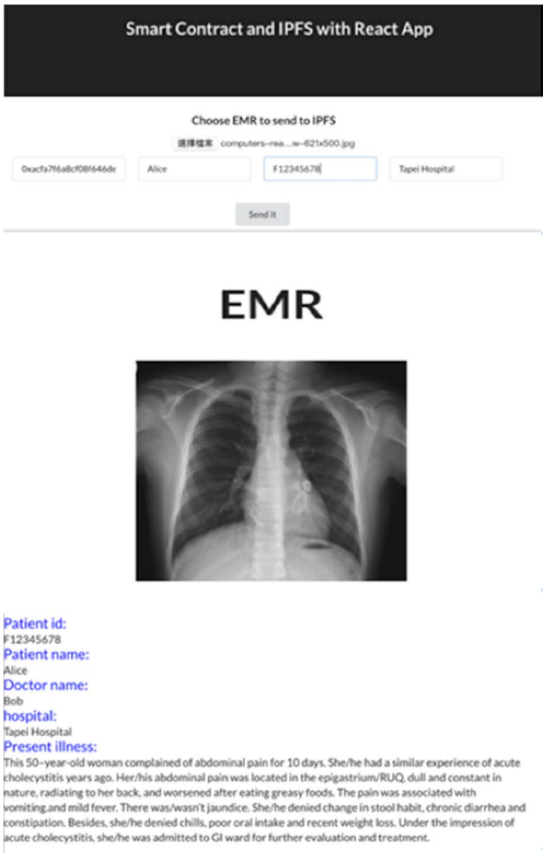

Fig. 3. Screenshot of uploading a medical record.

C. *Medical records authorization and access*

After the process of authorization, the system will return the transaction record of the authorization to the medical staff member. Then, the medical staff member can issue a request through the system to get the hash value, which is generated from the patient's medical record. The smart contract will first confirm the medical personnel's authority, and then reply to the corresponding hash value according to the patient's account address that the medical staff member enters to the system. When the system gets the hash value, it will display the whole medical record on the webpage.

In the proposed system, we designed the authorization mechanism using smart contracts. Smart contracts not only keep unauthorized persons from accessing medical records but also maintain the confidentiality and security of medical records. When transaction records are written into the blockchain, the network hackers need to change their target from the traditional database to the blockchain. Therefore, under the blockchain environment, our system can provide medical records with an authorization process that can't be tampered. Besides, both the medical staff and the patients have the right to read medical records encrypted on the blockchain. In our system, the patients can well manage their medical records and even plan for their health needs. Because the data and information on blockchain are open to everyone, whoever meets the requirements of smart contracts can query the blockchain data through the public interface. As a result, the data on the blockchain are open and highly transparent.

V. CONCLUSIONS

The medical record authority management method proposed in this study is adopted in the blockchain environment. It can protect against tampering, maintain information transparency, and decentralize the control of the authority. To achieve decentralization, we use IPFS to implement the medical record storage mechanism, making the sharing of medical records easier and safer. In this study, we only use Testrpc to simulate the blockchain environment. In the future, we will conduct the simulations of our system in the environment of private chains and public chains so that it can be closer to the actual applications.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Working Paper, Nov. 2008.

[2] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," 2017 IEEE International Congress on Big Data (BigData Congress), pp. 557-564, 2017.

[3] Pei-Shin He, "Applying Blockchain Technology on an Outpatient Electronic Medical Record System," Master Thesis, National Taipei University of Technology, 2017.

[4] Mei Ying, "Research on blockchain method for secure storage of medical records ,Jiang Xi Normal University," Science Journal Vol. 41(5), pp. 481-487, 2017.

[5] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "MedRec: Using blockchain for medical data access and permission management," International Conference on Open Big Data (OBD), pp. 25-30, Aug. 2016.

[6] Ethereum, https://www.ethereum.org

[7] J. Benet, "Ipfs-content addressed, versioned, p2p file system," arXiv preprint arXiv:1407.3561, Jul. 2014.

[8] Yin long, Wang Hong-wei, "Research on Distributed Data Sharing System Based on IPFS," Internet of Things Technology Vol. 6(6), pp. 60-62, 2016.

[9] Zheng, Qiuhong, et al., "An Innovative IPFS-Based Storage Model for Blockchain," 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI). IEEE, 2018.

[10] Yu-Syun Kao. "A Study on the Physician Behaviors in Affecting Usage with Electronic Medical Records Exchange," Zhongzheng University Information Management Department Medical Information Management Institute Dissertation, pp. 1-112, 2015

[11] Applications for Blockchain Grow in Energy, Transport, https://about.bnef.com/blog/applications-blockchain-grow-energy-transport/

[12] Narayan P. "Building blockchain projects: develop real-time DApps using Ethereum and JavaScript," Building Blockchain Projects: Building decentralized Blockchain applications with Ethereum and Solidity, 2017.