Security and Privacy of Electronic Health Records: Decentralized and Hierarchical Data Sharing using Smart Contracts

Ehab Zaghloul, Tongtong Li and Jian Ren

Abstract—**Over the last fifty years, medical treatment has seen remarkable advancement, however, the data management and storage systems of medical records has lagged in comparison. In addition, these systems are often inharmonious across platforms and do not put the privacy desires of patients first. While HIPAA and other laws are put in place to protect patient medical record security and privacy, these antiquated systems inherently hinder patient security and privacy. In this paper, we propose a novel data sharing and management scheme that empowers patients over their records by leveraging the security and privacy benefits of blockchain and smart contracts. In comparison to current methods for healthcare records management, our proposed scheme empower patients over their records and minimizes the dependencies on record-generating institutions. It also allows the patients to selectively share their records and disclose certain parts with specific data users based on the privacy preferences desired. In our security and privacy analysis, we show that patients can protect against potential threats to securely and privately share their records. Moreover, in our performance discussions, we show that smart contract design and development is key.**

Index Terms—**Blockchain, smart contract, healthcare records management.**

I. Introduction

The current applications utilized to manage and store electronic healthcare records are vulnerable to threats and lack the security, privacy and efficiency preferences desired by patients. Security and privacy vulnerabilities in these applications have resulted in significant revenue losses for medical institutions and the privacy issues of patients. In addition, these systems are often inefficient and lack interoperability with other systems, leading to a poor patient experience. Furthermore, the advancement of these applications remains slow-moving in contrast to the rate and diversity in format of records being generated.

In the U.S., regulations on patient health records are governed by the Health Insurance Portability and Accountability Act of 1996 (HIPAA) [1]. This ensures the protection and privacy of the health data of individuals, while still allowing the movement of health information necessary for the efficient administration of health services. However, from a security perspective, HIPAA falls behind in current security protocols and data encryption is considered as an *addressable* requirement. It does not dictate the ways in which to create and implement the systems currently being used. This leads to many variations in centralized systems used today and has prevented interoperability between medical institutions. As a result, the current systems lack security, prevent patient privacy, and are an inefficient way to transfer their health data, especially in cases of urgency. Furthermore, they lead to wasted resources where in the absence of patient data from a previous clinic, physicians will order duplicate testing.

Besides the lack of interoperability, current centralized electronic health record systems are liable to malicious attacks such as Distributed Denial of Service (DDoS) attacks [2] and social engineering attacks [3], [4]. A report by IBM and the Ponemon Institute in 2018 shows that healthcare breaches are the most costly among sectors such as business, banking, governments, etc, resulting in an average cost of $380 per record [5].

Recently, the decentralized blockchain technology has been proposed as a method to mitigate the security issues, privacy concerns and inefficiencies of various centralized platforms such as financial systems. In these decentralized systems, a network maintains a cryptographically secured public ledger of records and transactions ownership. The most common application of blockchain today is Bitcoin, a digital cryptocurrency developed in 2009 [6]. In contrast to centralized systems, blockchain-based systems do not suffer from a single point of failure and do not require a third trusted party to maintain the integrity of data ownership and flow. In fact, since the inception of Bitcoin, various blockchain-based systems continued to evolve aiming at providing enhanced features in terms of security, privacy and efficiency.

Today, smart contracts are one of the most well known and widely used features utilized by blockchain-based systems. They were initially introduced as a concept by Nick Szabo in 1994 [7] and were later implemented by multiple systems. Shortly after they were implemented by Ethereum [8], the second largest blockchain-based system that exists today, their popularity skyrocketed attracting dense research in their capabilities. In general, their main purpose is to digitally facilitate and enforce verified negotiations of contracts between two participating parties over the blockchain. This means, they are intended to allow non-trusting entities to share information while eliminating the need of a third trusted party. Since maintained over the blockchain, smart contracts are immutable, irreversible and can be tracked. Like any other transaction processed over the blockchain, they are based upon cryptographic primitives that ensure their integrity.

Smart contracts present a promising potential for data

978-1-5386-9223-3/19/$31.00 ©2019 IEEE

management. They can be utilized to provide patients access control over their records and eliminate the management services provided by the record-generating parties. These contracts are deployed over a decentralized blockchain which can help enhance record security, privacy and sharing efficiency. In addition, they can provide patients with fine-grained access-granting control to their records based on the their preferences. This allows adherence to the HIPAA requirement of sharing minimally reasonable data. Another benefit to these public and decentralized systems is open accessibility and increased efficiency by removing the requirement of needing compatible systems.

In this paper, we propose a secure, private and efficient electronic record sharing scheme that utilizes smart contracts deployed over a blockchain. Our proposed scheme empowers patients over their records allowing them to selectively share them with data users that satisfy their privacy preferences. The record-generating institutions can no longer delay or mitigate the sharing process once requested by another institution.

The rest of this paper is organized as follows. In section II, we discuss the preliminaries to our work. Next, in section III, we present our proposed scheme. Following that, in section IV, we discuss the security analysis of our work. In section V, we present some performance discussions. Finally, in section VI, we conclude the paper and discuss future work.

II. Preliminaries

A. Ciphertext Policy Attribute-Based Encryption

Ciphertext Policy Attribute-Based Encryption (CP-ABE) is a fine-grained access control and encryption scheme [9]. It allows data users to selectively share their data by utilizing access policies that are integrated into the ciphertexts during encryption. CP-ABE formally divides the process into four main functions.

Setup: A probabilistic function carried out by a key-issuer to generate a public key PK and master key MK.

Key Generation: A probabilistic function carried out by a key-issuer that utilizes the master key MK to generate a unique private key SK for a data user based on a set of attributes $\mathbb{A} = \{A_1, A_2 \ldots, A_n\}$ possessed by the user.

Encryption: A probabilistic function carried out by the data owner to encrypt data under an access policy $\mathcal{T}$ and generate ciphertext CT.

Decryption: A deterministic function carried out by the data user to decrypt a ciphertext CT using the uniquely generated private key SK for the user.

B. Privilege-based Multilevel Organizational Data-sharing

Privilege-based Multilevel Organizational Data-sharing scheme (P-MOD) [10] is an extension of CP-ABE that handles data sharing in complex hierarchical organizations. P-MOD partitions a data file into multiple segments based on user privileges and data sensitivity. Each segment of the data record is shared depending on data user privileges, the set of attributes that satisfy the hierarchical access policies.

Privilege-based Access Structure: The privilege-based access structure divides the data users of an organization into a hierarchy that consists of k levels, $\{\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$. Each level is associated with an access policy, $\{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_k\}$. An access policy $\mathcal{T}_i$ specifies a set of rules defined using logic gates and the different sets of attributes that can satisfy these rules. Data users in possession of a correct set of attributes that can satisfy a certain access policy $\mathcal{T}_i$ belong to level $\mathcal{L}_i$.

Data Partitioning and Encryption: The data owner partitions a data file $\mathcal{F}$ into a set of k record segments, that is $\mathcal{F} = \{F_1, F_2, \ldots, F_k\}$. Segment F_1 contains the most sensitive information of $\mathcal{F}$ that is to be shared with data users belonging to level $\mathcal{L}_1$. Segment F_k contains the least sensitive information of $\mathcal{F}$ that is to be shared with all data users belonging to any level of the hierarchy.

Next, the data owner generates a set of keys $\{sk_1, sk_2, \ldots, sk_k\}$ to symmetrically encrypt the corresponding segments of $\mathcal{F}$. The key sk_1 is first randomly selected by the data owner. The remaining keys are then derived using a cryptographic hash function h as follows

$$sk_{i+1} = h(sk_i). \tag{1}$$

A privileged data user that can satisfy access policy $\mathcal{T}_i$ belongs to level $\mathcal{L}_i$ and is granted key sk_i. Given sk_i, the data user can derive the keys $\{sk_{i+1}, \ldots, sk_k\}$ belonging to levels $\{\mathcal{L}_{i+1}, \ldots, \mathcal{L}_k\}$ as defined in equation (1). However, given the properties of hash function h, sk_i cannot be used to derive any of the keys $\{sk_1, \ldots, sk_{i-1}\}$.

Each $F_i \in \mathcal{F}$ is then symmetrically encrypted with its corresponding generated private key sk_i. Finally, each sk_i is encrypted with CP-ABE under its corresponding access policy $\mathcal{T}_i$.

C. Blockchain and Smart Contracts

Blockchain: The blockchain is a public ledger that stores all the cryptographically processed transactions performed over a peer-to-peer network. Initially implemented by Bitcoin [6], it provides its users with transaction confirmations to track ownership rights of BTC. The Bitcoin blockchain is based on a distributed consensus, Proof of Work (PoW), that allows any past and present online transaction to be verified. It consists of blocks $\{\mathsf{B}_0, \mathsf{B}_1, \cdots, \mathsf{B}_n\}$, each carrying a set of validated transactions, where B_0 represents the first block and B_n represents the most recent block attached to the blockchain. Each block B_i incorporates the cryptographic hash of its preceding block B_{i-1} to form the complete blockchain.

Smart Contracts: Smart contracts are pieces of code that enable users to write their own arbitrary rules for ownership, transaction formats and state transition functions. This means, two parties can digitally interact following a set of customized rules (defined by one of the parties) without the need of a third trusted party to secure the transaction. The deployment and/or interaction with smart contracts are immutable, permanent and irreversible. The process of deploying or interacting with an already deployed smart contract over the blockchain requires the users to connect to the peer-to-peer network through a client. Nodes in the network execute the smart contracts in return for a reward that keeps them incentivized. Rewards are referred to as *gas*, an execution fee willingly paid by nodes deploying or triggering the smart contracts.

III. The Proposed Scheme

A patient that visits a certain medical institution interacts with multiple of its staff members during the medical treatment process. In order to provide the patient with efficient treatment, staff members may request access to the previous records of the patient. These records may have been generated at the same/different institutions. By being able to access such information, patient safety is enhanced while saving time and resources. However, it is important to maintain the privacy preferences of the patient and only share this information with certain staff members as the patient desires. In addition, the patient should not have to disclose the entire records when sharing them. In fact, the patient should be able to grant access to specific parts of the records as desired.

The challenge is to provide a secure and efficient method for the patients to share their previously generated records based on the desired privacy preferences they please to maintain. Patients should also be able to selectively share certain parts of their record with the healthcare providers they select. In addition, they should not be concerned with the availability nor the guaranteed secure storage of their records provided by the record generating institutions.

A. Design Goals

To provide a record sharing scheme that satisfies the previously discussed challenges, we have the following design goals:

Data Ownership: A record belongs to its patient, not the record generating institution.

Fine-grained access control: Granting access to parts of the record is define and controlled by the patient according to the desired privacy preferences the patient wishes to maintain.

Collusion resistant: Staff members cannot combine their access permissions to access parts of the records they are not authorized to access individually.

Data confidentiality: Data integrity and security is protected. This includes the cloud server storing the records.

Data access trace-ability: The patient can recover the history of accesses to the shared record and track how the record has been accessed.

B. System Components

The system consists of six main actors and entities.

Patients: A set of patients $\{p_a \in \mathcal{P} \mid 1 \leq a \leq \infty\}$ which receive treatment from different medical institutions. Based on privacy preferences, each patient p_a can selectively choose how to share his/her records when visiting any medical institution.

Medical institutions: A set of medical institutions $\{m_b \in \mathcal{M} \mid 1 \leq b \leq \infty\}$ that provide patients in $\mathcal{P}$ with the necessary medical treatment and generate records that document the entire treatment results.

Staff members: The set of staff members $\{s_{b,l} \ \forall \ 1 \leq l \leq \infty\}$ employed at medical institution m_b that require access to the previous records of the visiting patient in order to provide enhanced treatment. Every staff member possesses a set of attributes $\mathbb{A}$ that can be used to categorize them into groups of similar characteristics, hence, rank them into a hierarchy. Attributes are identifiers that are selected from an infinite pool set. They may be as general as the role of a staff member and could be as unique as a bio-metric.

Key-issuer: A semi-trusted entity that generates access keys for eligible staff members which are granted access to certain parts of the record based on the access rights predefined by the patient.

Cloud server: A non-trusted entity used to store the patient records.

Blockchain P2P network: A non-trusted peer-to-peer network which maintains a blockchain that regulates access permissions of patient records to the staff members of medical institutions.

C. System Orchestration

For explanation purposes, we denote the record as $\mathcal{R}$ and the record attributes as $\{R_j \in \mathcal{R} \mid 1 \leq j \leq n\}$.

Partition and encrypt record: Based on the work presented in [10], the patients defines a privilege-based access structure that consists of k levels $\{\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_k\}$ and their corresponding access policies $\{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_k\}$. Following that, the patient partitions the record $\mathcal{R}$ into k segments such that, $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_k\}$. Each segment $\mathcal{R}_i \in \mathcal{R}$ represents one or more record attribute(s) and is of a different sensitivity value. Sensitivity is define by the patients themselves to give them the power of sharing their records as they please. Next, the patient generates generates k symmetric keys $\{sk_1, sk_2, \ldots, sk_k\}$ as defined by equation 1 and encrypts each corresponding $R_i \in \mathcal{R}$, that is

$$ER_i = \textsf{Sym-Enc}_{sk_i}(R_i), \tag{2}$$

where Sym-Enc is a symmetric encryption algorithm such as the Advanced Encryption Algorithm (AES) [11]. Finally, following the work presented in [9], each sk_i is encrypted under its corresponding $\mathcal{T}_i$ defined in the privilege-based access structure such that

$$Esk_i = \textsf{CPABE-Enc}(sk_i, \mathcal{T}_i), \tag{3}$$

where CPABE-Enc is the CP-ABE encryption function. The generated ciphertexts, $\{ER_1, ER_2, \ldots, ER_k\}$ and $\{Esk_1, Esk_2, \ldots, Esk_k\}$, are then stored by the patient in the cloud server.

Access policy: In order to grant access to staff members, the patient incorporates the set of access policies $\{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_k\}$ into a smart contract and deploys it over the blockchain.

Patient Visit: Later, the patient requires extended medical treatment, therefore, pays a visit to another medical institution. Staff members at that institution provide the patient with the required treatment.

Staff member requesting access permissions: In order for staff members at the institution to access any part of the record, they must possess a correct set of attributes $\mathbb{A}$ that satisfies any of the access policies defined by the patient. Staff members interact with the smart contract deployed by the patient and send along their possessed attributes. The smart contract is responsible for verifying the input attribute sets, announcing whether the staff members are privileged to access any parts of $\mathcal{R}$ and which level level they belong to.

Granting keys to staff members: A key-issuer continuously watching the blockchain observes an announcement that verifies a certain staff member. The key-issuer then generates a private key SK for the staff member that corresponds to the set verified attributes. Next, the issuer encrypts the generated key SK with the public key of the staff member such that

$$ESK = \mathsf{Asym\text{-}Enc}_{pub}(SK), \tag{4}$$

where Asym-Enc is the asymmetric encryption function. Finally, the key-issuer shares the encrypted private key ESK with the staff member.

Staff member private keys retrieval: The staff member can obtain the private key SK by decrypting ESK with his/her private key pr that corresponding to the public key used during encryption as

$$SK = \mathsf{Asym\text{-}Dec}_{pr}(ESK), \tag{5}$$

where Asym-Dec is the asymmetric decryption function.

Accessing record partitions: The patient initially fetches the encrypted record partitions $\{ER_1, ER_2, \ldots, ER_k\}$ and symmetric keys $\{Esk_1, Esk_2, \ldots, Esk_k\}$ from the cloud server. Next, using the derived private key SK from equation 5, the staff member decrypts the symmetric key Esk_i that belongs to level $\mathcal{L}_i$ such that

$$sk_i = \mathsf{CPABE\text{-}Dec}_{SK}(Esk_i), \tag{6}$$

where CPABE-Dec is the CP-ABE decryption function. The staff member can also derive the symmetric keys $\{sk_{i+1}, \ldots, sk_k\}$ as discussed in equation 1. Finally, the staff member decrypts the corresponding encrypted record partitions $\{ER_i, \ldots, ER_k\}$ as

$$R_i = \mathsf{Sym\text{-}Dec}_{sk_i}(ER_i), \tag{7}$$

where Sym-Dec is the symmetric decryption function.

D. Smart Contracts

We develop our proposed scheme by implementing smart contracts deployed over the Ethereum public testnet blockchain [12]. In general, our scheme is divided into two main smart contracts: i) staff member registration and ii) access verification.

Staff member registration: Staff Member Registration (SMR) is a global smart contract that registers staff members of medical institutions before they can interact with the system and request access to patient records. The staff members must visit initially visit a certified registering institution that physically verifies any attributes the members claim to possess. For example, a staff member must physically show the registering institution that he/she possesses a drivers license with a unique number to add it to his/her set of registered attributes. Once verified, the registering institution transacts with the SMR contract and sends it the attributes of the staff member to be stored over the blockchain.

Access verification and permission announcements: Access Verification and Permission Announcements (AVPA) is a smart contract developed and deployed by the patients over the blockchain to selectively define how to share a record among the registered staff members. A registered staff member can interact with it in order to request access to certain parts of the record. The smart contract consists of a twofold verification process.

Initially, the staff member sends a digital signature along with a challenge address. The smart contract derives the address used in the digital signature and compares it to the challenge address sent by the staff member. If a match is found, the AVPA verifies that the staff member is in possession of the challenge address as claimed.

Once verified, the AVPA fetches the stored attributes of the staff member from the SMR contract to be used in the second verification process. In this process, the fetched attributes are check to satisfy the access policies $\{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_k$ defined by the patient starting from the highest level. If they do, the AVPA contract publicly announces permission rights of the staff member to the specific level $\mathcal{L}_i$ of the hierarchy over the blockchain. At that point, the key-issuer will observe the announcement and proceed with generating the private key SK for the staff member as discussed in section III-C.

IV. Security and Privacy Analysis

In this section, the security and privacy of our proposed model is analyzed. We assume our proposed scheme utilizes a blockchain that is maintained by a large number of nodes, for example, Ethereuem. In such environments, tampering with already processed transactions is expensive and infeasible.

A. Security Analysis

We first discuss how our proposed scheme provide secure against potential threats.

a) Replay attacks: Staff members that wish to successfully interact with an AVPA smart contract to obtain access to certain parts of a record must provide a digital signature and a challenge address they claim to possess. Given that the blockchain and all interactions are public, attackers can fetch the digital signatures and attempt to reuse them along with the challenge addresses sent by the honest staff members. Therefore, to avoid such replay attacks, staff members are required to time stamp their digital signatures such that

Digital Signature $=$ $\mathsf{Sign}(M\|T)$, where M is the message being signed by the Sign and T is the time at which the signature is generated. However, in order to accept time-stamped signatures, a copy of each utilized digital signature must be maintained over the blockchain.

b) Collusion resistance: Two or more dishonest and registered users are not capable of bringing together their sets of attributes to obtain access to parts of records they are not eligible to access independently. This is achieved through the initial verification process performed by the AVPA smart contract. The contract will only accept a single digital signature at once. If valid, the AVPA contract will fetch the corresponding registered attributes which are then tested against the specified access policies. As a result, our proposed scheme protects collusion attacks.

B. Semi-trusted key-issuer

In the traditional CP-ABE [9] schemes, a completely trusted key-issuer is required to validate the attributes of staff member before generating private keys for them. In our proposed scheme, the key-issuer is no longer required to be completely trusted since attribute verification is performed by the AVPA smart contracts. The patient would only require the key-issuer to generate a private key for the permissioned staff member when an announcement is observed over the blockchain. The patient can easily track accesses to his/her records and learn if the key-issuer tries to cheat by generating keys for unpermission staff members. In that case, the patient can simply re-delegate the key-issuing process to another entity and re-deploy another AVPA smart contract to redefine access rights.

C. Privacy Analysis

One of the primary goals of our proposed scheme is to protect the privacy of the patients in order to comply with HIPAA. However, in our adversarial setting, we hold no guarantees of what information may be leaked about the staff members and their access patterns. All the information of staff members and their attributes is publicly available on the blockchain to be used in the initial verification process of the AVPA smart contracts. However, this feature is beneficial to the patients themselves. It allows them to continuously monitor how and when their records are being accessed. If the patient recognizes undesired access permissions, the patient can simply construct a new AVPA contract and deploy it over the blockchain to redefine the access permissions. This can also be performed in the case of revoking access rights from particular staff members.

V. Performance Discussions

The performance of our proposed scheme can be measured as the number of smart contract computations and their complexity, executed by a node upon receiving a transaction request. This directly correlates to the amount of gas paid by the transacting party to incentivize contract execution.

In general, an SMR smart contract includes functions to store the attributes of new staff members and return them when requested by the AVPA smart contract. Therefore, performance in this case is mainly based on the number of attributes being stored/returned such that

$$\text{Performance} \propto \text{Number of attributes.} \tag{8}$$

On the other hand, an AVPA smart contract is generally more complex in terms of number of computations and their complexity. Performance of such contracts mainly depends on three main factors: (i) a cryptographic computation to validate an input digital signature, (ii) the number of attributes fetched from the SMR contract, and (iii) the complexity of access policies being tested. Cryptographic computations are usually the most expensive to execute and require a lot of gas. In addition, the number of attributes stored/fetched directly depend on how a patient defines access policies. Therefore, it is necessary for patients to optimize access policies to improve the overall performance.

VI. Conclusion and Future Work

In this paper, we presented a secure and private medical record sharing and management scheme. Our proposed scheme empowers patients over their records and eliminates the reliance on the record-generating institutions when records are requested to be shared. We showed that utilizing a blockchain and smart contracts, patients can selectively share their records in a secure manner that preserves the desired privacy. We also proved that the process of granting access privileges can be performed in a decentralized manner, hence, eliminate the need of a completely third trusted party. In the future, we intend on formalizing our proposed model and security analysis. We also plan on presenting a feasibility study of our proposed scheme by testing it under various conditions.

References

[1] U. D. of Health and H. Services, "Heath insurance portability and accountability act." https://www.hhs.gov/hipaa/index.html, 2018.

[2] F. Lau, S. H. Rubin, M. H. Smith, and L. Trajkovic, "Distributed denial of service attacks," in *Systems, Man, and Cybernetics, 2000 IEEE International Conference on*, vol. 3, pp. 2275–2280, IEEE, 2000.

[3] R. Anderson, *Security engineering*. John Wiley & Sons, 2008.

[4] H. K. Patil and R. Seshadri, "Big data security and privacy issues in healthcare," in *Big Data (BigData Congress), 2014 IEEE International Congress on*, pp. 762–765, IEEE, 2014.

[5] IBM and Ponemon, "2018 cost of data breach study: Global overview." https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=55017055USEN&, 2018.

[6] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[7] N. Szabo, "Smart contracts," 1994.

[8] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project yellow paper*, vol. 151, pp. 1–32, 2014.

[9] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *2007 IEEE symposium on security and privacy (SP'07)*, pp. 321–334, IEEE, 2007.

[10] E. Zaghloul, K. Zhou, and J. Ren, "P-mod: Secure privilege-based multilevel organizational data-sharing in cloud computing," *arXiv preprint arXiv:1801.02685*, 2018.

[11] J. Daemen and V. Rijmen, *The design of Rijndael: AES-the advanced encryption standard*. Springer Science & Business Media, 2013.

[12] "Ropsten (revival) testnet." https://ropsten.etherscan.io, year = 2018.