Electronic Health Record Sharing Scheme With Searchable Attribute-Based Encryption on Blockchain

SHUFEN NIU, LIXIA CHEN, JINFENG WANG, AND FEI YU
College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China
Corresponding author: Shufen Niu (sfniu76@nwnu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61562077, Grant 61662071, Grant 61662069, and Grant 61772022.

ABSTRACT With the digitization of traditional medical records, medical institutions encounter difficult problems, such as electronic health record storage and sharing. Patients and doctors spend considerable time querying the required data when accessing electronic health records, but the obtained data are not necessarily correct, and access is sometimes restricted. On this basis, this study proposes a medical data sharing scheme based on permissioned blockchains, which use ciphertext-based attribute encryption to ensure data confidentiality and access control of medical data. Under premise of ensuring patient identity privacy, a polynomial equation is used to achieve an arbitrary connection of keywords, and then blockchain technology is combined. In addition, the proposed scheme has keyword-indistinguishability against adaptive chosen keyword attacks under the random oracle model. Analysis shows that the scheme has high retrieval efficiency.

INDEX TERMS Electronic health records, data sharing, searchable attribute-based encryption, blockchain.

I. INTRODUCTION

Electronic health records (EHRs) reduce the cost of traditional medical data storage. EHRs are designed to allow patients to manage their own medical data. Data users have limited access to EHRs. Meanwhile, patients do not share their private medical data with data users, indicating the value given by patients to EHR safety and privacy.

From the perspective of using electronic health records, an encrypted retrieval of such records must be implemented. A searchable encryption scheme has been proposed to achieve retrieval on ciphertext [1]. Searchable encryption can be symmetric or asymmetric. Symmetric searchable encryption has higher encryption efficiency compared with the asymmetric one [2], [3]; however, its private key management is more complicated during data sharing. To solve the key management problem, Boneh et al. [4] introduced public key encryption with keyword search. Public key searchable encryption is suitable for multi-user data sharing.

To achieve multi-user data sharing, [5] proposed attribute-based encryption.

With the rapid development of electronic health records, distributed storage platforms have received extensive attention through a network of numerous storage devices. With the increasing popularity of these storage platforms, distributed platforms that store data have become a target of cyber attacks. To reduce the risk of data leakage, [6] and [7] proposed data storage on blockchain.

Blockchain decentralization eliminates the concentration of cloud storage servers and solves the security flaws caused by network attacks. A blockchain is created by the nodes of the mining block in the system, and each block contains the hash function value of the previous block header. This structure makes the blockchain difficult to tamper. With the development of the bitcoin blockchain [8], other types of blockchains have also attracted widespread attention. The ethereum blockchain [9], [10] could execute smart contracts. A smart contract is a function that could be written on a blockchain and executed by a node. Users could not run programs on the system indefinitely due to the cost of mining. Therefore, ethereum allow users to create private or
permissioned blockchains that could be managed and controlled by small users. A permissioned blockchain increases the ability to control privacy and modify mining costs.

A. RELATED WORK

In recent years, blockchains have received extensive attention in the fields of data storage, data sharing, and privacy protection. Blockchain are transparent and not tampered. Therefore, data stored in the blockchain must be encrypted. Searchable encryption solves the search problem on the ciphertext. Reference [11] built a secure index using bloom filter, but the search results are partially inaccurate. For the problem of symmetric searchable encryption key management, Boneh et al. [4] proposed a public key searchable encryption scheme. By narrowing down the search through keywords, Golle et al. [12] proposed a multi-keyword search scheme, and [13], [14] and [15] proposed other multi-keyword searchable encryption schemes with special functions, which extended multi-keyword retrieval applications.

In the context of information sharing, attribute encryption extends the application scenario to a one-to-many situation. Qiu et al. [16] proposed an attribute searchable encryption that could resist the internal keyword guessing attack, but requires the data owner to be online in real time, such that a new data user who joins the system negotiates the search key with data owner, and the data owner is considerably burdened. Li and Xi [17] proposed attribute encryption based on a key strategy to implement a single keyword-searchable encryption scheme. To retrieve files efficiently, Song et al. [18] proposed an attribute-based searchable encryption scheme that supports user revocation, implementing extended fine-grained search authorization, which allows owners to encrypt and outsource their data to the cloud server independently. Sun et al. [19] used inert re-encryption technology to implement attribute revocation.

All the above solutions are based on cloud server retrieval, but most cloud servers are semi-honest and curious. Blockchain technology can solve the semi-honesty problem of cloud servers and freely access and share data [20]–[23]. Healthbank uses blockchain technology to ensure absolute data storage security. Dagher [24] provided a data storage structure on a blockchain but did not provide an efficient search method. Li et al. [25] implemented data retrieval on a blockchain, but the search results were inaccurate, and the search efficiency was low. Zhang et al. [26] achieved the verifiability of server-side data. Xia et al. [27] proposed a blockchain-based health data sharing model. The system for sharing EHRs in the blockchain uses the blockchain as an aid for data sharing instead of a primary tool for data storage, management, and sharing. In addition, the existing blockchain medical data retrieval solution does not provide detailed solutions for data owners and data users.

B. CONTRIBUTION

In this paper, we propose an electronic health record (EHR) sharing scheme based on the combination of each hospital server and permissioned blockchain on the premise of not violating patient privacy. Our contributions are as follows:

- Without the introduction of third-party search, using keywords and ciphertext stored separately on permissioned blockchain and hospital cloud storage servers, which not only solves the problem of semi-honest search on cloud storage server, but also solves the problem of limited blockchain storage space. It also achieves the immutability of electronic health record and the sharing of electronic health record.
- The ciphertext-based attribute encryption is used to satisfy the multi-user retrieval and realize fine-grained access control of users.
- Arbitrary connection of multiple keywords is achieved by using polynomial equation, which not only supports multi-keyword search by users, but also improves the accuracy of data retrieval.
- The data public audit mechanism is adopted to achieve the identity authentication of doctors, prevent the attending doctors from uploading the electronic health records of patients on the hospital server with a false identity, and ensure the authenticity and reliability of the uploaded data.

C. ORGANIZATION

The rest of the paper is organized as follows: Section II is a review of preliminary knowledge. Section III presents the system and security models of the proposed scheme. Section IV describes the EHR sharing scheme based on a permissioned blockchain. Section V presents the safety and efficiency analyses. Section VI provides the conclusion.

II. PRELIMINARIES

A. BILINEAR GROUPS

Let $G_1$ and $G_2$ be two cyclic groups of the same prime order $p$; let $g_1$, $g_2$ be the generator of $G_1$. A bilinear map $e : G_1 \times G_1 \rightarrow G_2$ must satisfy the following properties:

- **Bilinear:** For any $g_1, g_2 \in G_1$, $a, b \in Z_p^*$, we have $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$.
- **Non-degenerate:** $g_1, g_2 \in G_1$ exists such that $e(g_1, g_2) \neq 1$.
- **Computable:** For any $g_1, g_2 \in G_1$, an efficient algorithm computes $e(g_1, g_2)$.

B. HARDNESS ASSUMPTION

**Definition 1 (Decisional Diffie-Hellman Problem, DDH [?]):** Let $G_1$ be a group in which prime order is $p$; let $g$ be a generator of $G_1$ and give a tuple $(g, g^a, g^b)$, where $a, b \in Z_p^*$. The DDH assumption holds if no polynomial time algorithm has a non-negligible advantage to distinguish if $g^c$ equals $g^{ab}$ or a random element of $G_1$.

Under the DDH assumption, the probabilistic polynomial time adversary $A$ can distinguish tuple $(g, g^a, g^b, g^{ab})$ from tuple $(g, g^e, g^f, g^{ef})$ with a non-negligible advantage. If $Pr[C(g, g^a, g^b, g^{ab}) = C(g, g^a, g^b, g^c)] \geq \varepsilon$, then challenger $C$ has an advantage $\varepsilon$ solving DDH.
C. PERMISSIONED BLOCKCHAINS

Public chains are blockchains where anyone can access the system at any time to read data, send transactions, and compete for accounting. Public chains generally encourage participants to compete for accounting by encrypting digital currencies, such as bitcoins and ethereum to ensure data security. Public chains are completely decentralized. Permissioned blockchains allow each participant node in the blockchain system to be permissioned (e.g., private and alliance chains). Some permissioned blockchains do not have a digital currency mechanism because some nodes are designated in the system for accounting without the need to encrypt currency. Nodes on a permissioned chain are allowed to perform specific functions, such as reading, accessing, and writing information on the blockchain. An increasing number of businesses are choosing permissioned blockchain networks because such networks could be configured. The network selectively adds necessary restrictions and controls the roles that each participant needs in each application. The storage of medical data and the determination of participants on the permissioned chain are depicted in Figure 1.

D. ACCESS STRUCTURE

Attribute characteristics are used to describe a doctor and a patient. A tree is used to represent the access control structure [5]. The internal node relationship of the access tree is represented by AND and OR gates, and the leaf nodes of the access tree represent attributes. Only the attributes that satisfy the access structure can access the electronic health record.

The tree contains a \( v \) (the access structure can access the electronic health record. access tree represent attributes. Only the attributes that satisfy is represented by AND and OR gates, and the leaf nodes of the structure [5]. The internal node relationship of the access tree represents a patient. A tree is used to represent the access control

III. PROBLEM FORMULATION

This section describes an EHR system based on a permissioned blockchain, which comprises four entities, namely, patient, doctor, cloud server (medical institution), and searcher. The formal algorithm and security model of the proposed scheme are also introduced.

A. EHR SYSTEM MODEL

To enhance medical data safety, the proposed scheme builds the medical platform shown in Figure 2 based on medical institutions that share EHRs in a certain region. The doctor stores the encrypted electronic health records on the servers of the respective hospitals, and the server administrator stores some keywords of these records (in the form of transactions) in the permissioned blockchain to achieve electronic health record sharing among hospitals. The six entities are described as follows:

1) SYSTEM MANAGER

The system manager is in charge of the entire system, and every patient, doctor, and data user needs to register before entering the system. The system manager generates public keys and private keys for them. Furthermore, the system manager distributes and revokes attribute as attribute authority.

2) EHR SYSTEM

This entity refers to a user who provides medical services to a medical institution. Typically, each hospital has a server and multiple computer clients. Each computer client records the patient’s health information. The server administrator then broadcasts the keywords of the electronic health record to the permissioned blockchain. The server registers users and doctors in the system. When the doctor uploads an electronic health record to the server, the server must verify the identity of doctor.

3) PATIENT

When a patient visits a hospital, he or she must register at the hospital first. During registration, the patient is provided a visit token, which is used as a pass for the patient to see the doctor. The authorized doctor generates an electronic health record of the patient, and stores the encrypted electronic health record on the hospital server. The server manager stores some keywords of the electronic health record on the permissioned chain.
4) DOCTOR
The doctor generates a patient’s electronic health record, encrypts the electronic health record using the patient’s access structure, and uploads the encrypted electronic health record to EHR System.

5) DATA USER
When third-party organizations or individuals (called data users) exist other than the hospitals and patients who need to access patients’ data, obtaining searching trapdoors generated by patients is necessary to perform keywords searching in the blockchain.

6) BLOCKCHAIN
Participants in the permissioned blockchain consist of medical institutions, and the searchers comprise participants specified by the permissioned blockchain system. Participants verify the transaction records broadcasted by each server to the blockchain and implement the data search function on the permissioned blockchain.

B. ALGORITHM DEFINITION
The EHR sharing scheme on the permissioned blockchain comprises six algorithms, which are defined as follows:

1) SYSTEM INITIALIZATION
   • Setup(\(\lambda\)) \(\rightarrow\) (pp, msk): The setup algorithm is run by the system manager. It takes as input a security parameter \(\lambda\), and then output the public parameters pp and master secret key msk.

2) KEY GENERATION
   • KeyGen(msk, pp, S) \(\rightarrow\) sk: The secret key generation algorithm is run by the system manager. It takes as input the master secret key msk, public parameters pp and the attribute set S of a user, then outputs a secret key sk.

3) DATA STORAGE
   • Enc(pp, T, W, M) \(\rightarrow\) (I, C): The encryption algorithm is run by the doctor. It takes as input the public parameters pp, access control structure T of patient P, a keywords set W of the shared electronic health records and electronic health records M, then output the keywords index I and a ciphertext C.

4) DATA QUERY
   • Trapdoor(pp, W', sk) \(\rightarrow\) T_{W'}: The trapdoor generation algorithm is run by the user. It takes as input the public parameters pp, search keywords set W’ and secret key sk, then outputs the search trapdoor T_{W’}.
   • Search(I, T_{W'}) \(\rightarrow\) C: The search algorithm is run by participants in permissioned blockchain. It takes as input the keywords index I and search trapdoor T_{W’}, then outputs a ciphertext C.

5) DATA DECRYPTION
   • Decrypt(pp, sk, C) \(\rightarrow\) M: The decryption algorithm is run by the user. Taking public parameters pp, secret key sk and ciphertext C, this decryption algorithm outputs the message M.

C. SECURITY DEFINITIONS
Definition 1: If an adversary without probability polynomial time can attack the scheme with a non-negligible advantage, then the scheme has indistinguishable security under an adaptive keyword attack.

Attack games between adversary A and challenger C under adaptive selection keywords are as follows:
Adversary initialization: The adversary A chooses the access control structure $T'$ and informs the challenger C.

System initialization: The challenger C runs the system initialization algorithm $\text{Setup}(\lambda)$ and sends the public parameters $pp$ to the adversary A.

Query phase 1:
- Private key query: The adversary A asks the challenger C for the private key $sk$ of some attributes $S$. The challenger C runs $\text{KeyGen}(pp, S)$ and sends the private key $sk$ to the adversary A and maintains a list $L_{sk}$ of the private keys.
- Trapdoor query: The adversary A can adaptively ask the challenger C for the trapdoor $T_{W'} = \text{Trapdoor}(pp, sk, w)$ for any keyword $w \in \{0, 1\}^*$ from keywords set $W'$.

Challenge phase: The adversary A sends the challenger C two keywords $w_0, w_1 \in W$. Then, the challenger C randomly selects one of the keywords $w_i$ to encrypt, and sends the encrypted keyword to the adversary A.

Query phase 2: The adversary A repeats query phase 1 and continues to ask for keywords $w$, where $w \neq w_0, w_1$.

Guess phase: Eventually, adversary A outputs $c' \in \{0, 1\}$ and wins the game if $c' = c$.

The advantage of adversary A attacking the scheme is defined as the following function of the security parameter $k$.\[
\text{Adv}_{A,w}(k) = |Pr(c') - \frac{1}{2}|
\]

IV. OUR SCHEME

A. ELECTRONIC MEDICAL RECORD RETRIEVAL SCHEME BASED ON PERMISSIONED BLOCKCHAIN

1) SYSTEM INITIALIZATION

The proposed scheme consists of three stages: system setup, data generation and storage, and data search and access.

Phase 1: System setup
- $\text{Setup}(\lambda)$: Input security parameter $\lambda$; let $G_1$ and $G_2$ be two multiplicative cyclic groups with generators $p$, and $g_1, g_2$ are two generators of $G_1$. Let $e: G_1 \times G_1 \rightarrow G_2$ be an admissible bilinear map. The system randomly selects $\alpha, \beta \in Z_p^*$, computes $g_2^\alpha, g_2^\beta, g_1^\alpha, g_1^\beta$. Select four hash functions $H_1: \{0, 1\}^* \rightarrow Z_p^*$, $H_2: G_1 \rightarrow Z_p^*$, $H_3: Z_p^* \rightarrow G_2$, $H_4: G_2 \rightarrow \{0, 1\}^*$. The system parameters $PP = (p, e, g_1, g_2, g_2^\alpha, g_2^\beta, g_1^\alpha, g_1^\beta, G_1, G_2, H_1, H_2, H_3, H_4)$ is published, the master secret key $msk$ keeps secret.

$$\text{msk} = (\alpha, \beta)$$ (1)

- $\text{KeyGen}(\lambda)$: The system manager chooses $r \in Z_p^*$, and computes $R = g_1^r, R' = g_1^r g_2^r$. For each attribute $a \in S$ randomly chooses $t_a \in Z_p^*$ and computes $X_a = g_2^{t_a}, Y_a = g_1^{t_a} g_1^{H_1(a)}$. Lastly, the attribute authority sends secret key $sk$ to the user, as shown in (2).

$$sk = (R, R', \{X_a, Y_a\}_{a \in S})$$ (2)

Phase 2: Data generation and storage

When patient $P$ comes to the hospital for treatment, the hospital server randomly chooses $h \in Z_p^*$, computes $\mu = H_3(h)$ and server reserves $\mu$. The $h$ is then sent to the patient by a safe channel, and the patient is simultaneously designated a doctor $D$. When the patient $P$ sees the doctor and presents $h$, and chooses a tree $\Gamma$ as the access structure, and $h$ is used as a secret shared value of access structure $\Gamma$.

The patient $P$ runs $\text{Share}((\Gamma, h) \rightarrow \{h_i(w)\}_{w \in \text{LV}(\Gamma)})$ for each leaf node in the access structure $\Gamma$ to obtain a secret value and calculates $A_v = 8 g_2^{h_i(0)}, B_v = 8 g_1^{H_1(s_v)h_i(0)}$ for each leaf node attribute $s_v$. of the access control structure $\Gamma$, and then send to Doctor $D$. Doctor $D$ runs encryption algorithm as follows:

- $\text{Encrypt}(m, w)$:
  (1) Doctor $D$ generates health records $M \in \{0, 1\}^*$ about patient $P$ and uses access structure $(\Gamma, h)$ to calculate $C_0 = 8 g_2^{h_i(0)}, C_1 = M \cdot H_4(e(g_1, g_1)^h)$. (2) Doctor D extracts keyword sets from the ciphertext C, and randomly selects $a, b \in Z_p^*$ to construct a polynomial equation about keywords $W = \{w_1, w_2, \ldots, w_m\}$, as shown in (3).

$$g(x) = a(x-h_1(w_1))(x-h_1(w_2)) \ldots (x-h_1(w_m)) + b = a_m x^m + a_{m-1} x^{m-1} + \ldots + a_1 x + a_0$$ (3)

For each $a_j, j \in \{1, 2, \ldots, m\}$ calculates $L_j = g_2^{a_j}, F_0 = g_2^{h_i(h)}; F_1 = g_2^{h_i(h) + b}$.

The encryption algorithm output is expressed as:

$$C_M = (\Gamma, C_0, C_1)$$ (4)

$$I_w = (L_j, F_0, F_1, (A_v, B_v)_{v \in \Gamma})$$ (5)

The doctor D uploads ciphertext $C_M$ and $I_w$ on the database server of the hospital(EHR system). When the doctor uploads the encrypted health records and keyword sets, the cloud server must verify the identity of the doctor, the doctor D randomly chooses $x_1, r_2 \in Z_p^*$, for each attribute $a \in S$, randomly chooses $t_a \in Z_p^*$ and calculates $\eta_1 = g_1^{r_2} g_1^{H_1(a)}, \eta_2 = g_1^{r_2} g_1^{H_1(a)}, h' = H_2(\eta_1^{r_2}) \oplus h$. Then the doctor D sends $(\eta_1, \eta_2, h')$ to the EHR server, which calculates $h'' = H_2(\eta_1^{r_2}) \oplus h'$ and determines whether the equation $H_3(h'') = \mu$ is true.

Correctness :

$$H_3(h'') = H_3(H_2(\eta_1^{r_2} H_1(a)) \oplus h') = H_3(H_2(g_1^{r_2} g_1^{H_1(a)})) \oplus h' = H_3(H_2(g_1^{r_2} H_1(a)) \oplus h') = H_3(h) = \mu$$

The manager of EHR system extracts the block $ID_b$ ($ID_b$ is the identity of a hospital in the EHR system), patient’s identity $ID_p$, and secure keyword $I_w$. The server of EHR
system broadcasts the new transaction \( TX_{N} = (ID_{b}, ID_{p}, I_{w}) \) to the blockchain.

**Phase 3: Data search and access**

- **Trapdoor**: The data users construct the trapdoor of the keywords \( W' = \{w'_1, w'_2, \ldots, w'_{n}\} \), where \( n \leq m \), and randomly chooses \( r_3 \in Z_p^* \) to calculate

\[
K_0 = y_1^{ar_3}, \quad K_1 = R_3 = x_1^{\frac{e}{r_3}}
\]

For \( a \in S \), calculate:

\[
X'_a = X_3^{r_3}\alpha, \quad Y'_a = Y_3 = g_1^{\frac{e}{s}\cdot H_i(a)}
\]

\[
E_i = g_1^{\frac{e}{s}\cdot H_i(a)h_0(0)} = (g_1, g_2)^{\delta r_3 h_0(0)}
\]

Then, the trapdoor \( T_{W'} \) of the keyword is set as shown in (6).

\[
T_{W'} = (K_0, K_1, E_i, (X'_a, Y'_a)|a \in S) \quad (6)
\]

- **Search**: The searcher on the permissioned blockchain according to the attributes of the trapdoor and calculates the secret value of the leaf node as shown in (7).

\[
E_v = e(Y'_a, A_v) = e(g_1^{\frac{e}{s}\cdot H_i(a)h_0(0)}, g_2^{\frac{e}{s}\cdot H_i(a)h_0(0)})
\]

\[
e = e(g_1^{\frac{e}{s}\cdot H_i(a)h_0(0)}, g_2^{\frac{e}{s}\cdot H_i(a)h_0(0)})
\]

If the attributes of user satisfy access tree structure, then the searcher determines whether (8) is equal; if equal, then the searcher broadcasts the correct search result in the form of a transaction.

\[
e(F_0, K_0) = E_v \cdot e(F_1, K_1) \prod_{i,j=0}^{m} e(L_i, E_i) \quad (8)
\]

**Correctness**:

\[
e(F_0, K_0) = e(g_1^{\frac{e}{s}\cdot H_i(a)h_0(0)}, g_2^{\frac{e}{s}\cdot H_i(a)h_0(0)})
\]

\[
\prod_{i,j=0}^{m} e(L_i, E_i) = \prod_{i,j=0}^{m} e(g_1^{\frac{e}{s}\cdot H_i(a)h_0(0)}, g_2^{\frac{e}{s}\cdot H_i(a)h_0(0)})
\]

\[
e(g_1, g_2)^{\delta r_3 h_0(0)}
\]

\[
E_v \cdot e(F_1, K_1) = e(g_1, g_2)^{\delta r_3 h_0(0)}
\]

\[
e(g_1, g_2)^{\delta r_3 h_0(0)}
\]

\[
e(g_1, g_2)^{\delta r_3 h_0(0)}
\]

\[
e(g_1, g_2)^{\delta r_3 h_0(0)}
\]

- **Decryption**: The data user obtains the ciphertext \( C = (T, C_0, C_1) \) from the hospital server, decrypts the ciphertext with its own private key, and then calculates

\[
E' = e(Y'_a, A_v) = e(g_1, g_2)^{\delta}, \quad M = C_1 / H_4(e(R, C_0) / E'_v)
\]

**V. SAFETY AND PERFORMANCE ANALYSIS**

**A. SAFETY ANALYSIS**

**Theorem 1**: If the DDH problem is difficult, then the scheme has keyword-indistinguishable security under an adaptive keyword attack in the random oracle model.

**Proof**: If a polynomial time adversary \( A \), which could win the game with a non-negligible advantage \( \epsilon \), then challenger \( C \) could solve the DDH problem with an advantage \( \frac{\epsilon}{2} \).

The attack game between adversary \( A \) and challenger \( C \) is as follows:

1) **ADVERSARY INITIALIZATION**

The adversary \( A \) chooses the access control structure \( T' \) and informs the challenger \( C \).

2) **SYSTEM INITIALIZATION**

Challenger \( C \) randomly chooses \( \alpha, \beta \in Z_p^* \) and calculates

\[
s_2^\alpha \cdot s_1^\beta \cdot g_1^\beta, \quad g_2^{\delta} = g_2^\beta \cdot s_1^\beta \cdot g_1^\beta \]

where \( s_2^\alpha = g_2^\beta \). Challenger sends the public parameter \( PP = (p, e, g_1, g_2, g_1^\beta, g_2^{\delta}, G_1, G_2, H_1, H_2, H_3, H_4) \) to the adversary \( A \).

3) **QUERY PHASE 1**

- **H1 query**: The challenger \( C \) maintains an empty initial list \( L_{H1} = (s_i, c_i, h_i) \). The adversary \( A \) asks for the \( H_1 \) value of the attribute \( s_i \). If the attribute \( s_i \) is in the list, then the challenger \( C \) returns \( h_i \) to the adversary \( A \). If the attribute \( s_i \) is not in the list \( L_{H1} \), the challenger \( C \) selects a random number \( c_i \in Z_p^* \) to calculate \( h_1 = c_i \) and adds the attribute to the list \( L_{H1} \).

- **H2 query**: The challenger \( C \) maintains an initial list \( L_{H2} = (s_i, c_i, h_1, \eta_i, h_2, h_2) \). The adversary \( A \) asks the challenger \( C \) for the value of \( H_2 \). The challenger \( C \) first checks, whether \( H_2 \) already exists in the list \( L_{H2} \); if \( H_2 \) exists, then challenger returns the previously stored
value or randomly chooses \( d \in \mathbb{Z}_p^* \) and adds \( d = h_2 \) in the list \( L_{H_2} \) and then returns \( d \) to the adversary \( A \).

- **\( H_3 \) query:** The challenger \( C \) maintains an initial list \( L_{H_3} \). The adversary \( A \) queries \( H_3 \). The challenger \( C \) randomly chooses \( d_1 \), returns \( H_3(h) \) to the adversary \( A \), and adds \((h, d_1)\) to the initial list \( L_{H_3} \).

- **\( H_4 \) query:** The adversary \( A \) queries the challenger \( C \), who first checks if \( H_4 \) is in the list \( L_{H_4} \); if \( H_4 \) exists, then return previously stored value to \( A \), or randomly select \( d_2 \in \mathbb{Z}_p^* \) and add \( d_2 \) to list \( L_{H_4} \), then return \( d_2 \) to \( A \).

- **Private key query:** The adversary \( A \) queries the challenger \( C \) for the private key of some attributes \( s \). The attribute \( s \) does not satisfy the access structure \( T \), and the challenger \( C \) calculates \( R = g_1^{a_2w} \), \( R' = g_1^{a_2w} \) according to the system parameters. For each attribute \( a_i \in \mathbb{Z}_p^* \) is randomly selected, and \( X_a = g_1^{a_i}, Y_a = g_1^{a_i \cdot H_1(a)} \) is calculated. The challenger \( C \) sends the private key \( sk = (R, R', X_a, Y_a) \) to the adversary \( A \) and maintains a list \( L_{dk} \) of private keys.

- **Trapdoor query:** When the adversary \( A \) asks the challenger \( C \) for the trapdoor of the keyword \( w_i \), the challenger \( C \) randomly chooses \( r_i \) and queries the list \( L_{dk} \) to obtain the private key \( sk \). The challenger \( C \) calculates

\[
K_0' = g_1^{ar_i}, \quad K_1' = g_1^{(a_2-r_1w_1)}, \quad X_a' = g_1^{r_1w_1},
\]

\[
Y_a' = g_1^{r_1w_1}, \quad E_i' = g_1, \quad E_i' = g_1^{r_1w_1}
\]

and the challenger \( C \) returns the trapdoor \( T_{w_i} = (K_0', K_1', X_a', Y_a') \) of the keyword \( w_i \) to the adversary \( A \).

4) **CHALLENGE PHASE**

The adversary \( A \) randomly chooses two keywords \( w_0 \) and \( w_1 \) and sends them to the challenger \( C \), \( w_0 \) and \( w_1 \) are not queried in phase 1. The Challenger \( C \) randomly chooses keyword \( c = \{0, 1\} \) for encryption. The challenger \( C \) randomly chooses \( a, b \in \mathbb{Z}_p^* \) to construct the equation \( g(x) = a(x - H_1(w_i)) + b \) of the keyword \( w_i \), and uses the secret shared value \( h \) of the access tree to calculate \( L^x = g_1^{a \cdot h}, F_0 = g_1^{b \cdot h}, F_1 = g_1^{b \cdot h} \), and sends the encrypted keyword \((L^x, F_0, F_1)\) to \( A \).

5) **QUERY PHASE 2**

Adversary \( A \) could adaptively repeat the inquiry of the phase 1 to challenger \( C \), \( w \neq w_0, w_1 \).

6) **GUESS PHASE**

The adversary \( A \) outputs \( c' \in \{0, 1\} \), if \( c = c' \), then the adversary \( A \) wins the game.

If the challenger \( C \) solves the difficult problem, then the adversary \( A \) obtains the ciphertext of keywords as a result of the random encryption of \( C \) and guesses whether \( C_c \) is \( w_0 \) or \( w_1 \). The advantage of guessing is \( \frac{1}{2} \).

The advantage of the challenger \( C \) in solving the DDH is \( v' = Pr[c = c'] - \frac{1}{2} = Pr[c = c'|c = 1] \cdot Pr[c = 1] + Pr[c = c'|c = 0] \cdot Pr[c = 0] - \frac{1}{2} = (\frac{1}{2} + \epsilon) \cdot \frac{1}{2} + (\frac{1}{2} - \epsilon) \cdot \frac{1}{2} = \frac{\epsilon}{2} \). Assuming the DDH problem is difficult to resolve, adversary \( A \) cannot break solution under the selective access structure \( T' \). This scheme is indistinguishable under an adaptive choose-keyword attack.

B. PERFORMANCE EVALUATION

In this section, the performance and efficiency of the proposed scheme are analyzed and compared with schemes [16], [18], [19] through a numerical simulation experiment.

1) **THEORY ANALYSIS**

The performance and efficiency of a scheme have an important effect on the sharing of electronic health records. Table 1 shows the performance characteristics and calculations of the proposed scheme, whose efficiency is compared with that of the classic scheme. The notations used in this section is introduced as shown: \( P \) and \( M \) denote the exponential and multiplication operations in the cyclic group, respectively; \( E \) is a bilinear pair operation; \( m \) is the number of keywords; \( n \) is the number of search keywords; and \( v \) is the number of attributes.

The performance characteristics of the proposed scheme are shown in Table 1. From the perspective of data search, [16] and [19] support single-keyword search, and the accurate retrieval of data transactions could not be achieved. [18] and the proposed scheme support multi-keyword search, but the latter uses polynomial equations to achieve an arbitrary connection of keywords. From the perspective of index storage, [16],[18], and [19] store the index on the cloud storage server and retrieve keywords using the cloud server. The proposed
scheme stores the index on the permissioned blockchain and retrieves the data from the blockchain to obtain reliable and non-tamperable search results. From the access control structure of the attribute analysis, [16], [18], and [19] use the AND gate access structure to implement the access permission settings of a user. By contrast, the proposed scheme uses the access structure of the tree to implement fine-grained access control of data users.

The calculation of the proposed and the existing scheme is shown in Table 1. The index calculation amount of the proposed and the other scheme is related to the number of attributes and encryption keywords. The index calculation of the proposed scheme is slightly lower than that of [16], [18] and [19]. The proposed scheme uses a polynomial equation, thus the encryption calculation is low, but the arbitrary connection of the keywords is achieved. In the trapdoor calculation, the proposed scheme has a higher calculation amount compared with other references. The calculation of the trapdoor is determined by the number of attributes and the number of search keywords. In the amount of search calculation, the number of attributes affects the amount of search calculation, but the number of keywords in the proposed scheme also affects the amount of search calculation. In this scheme, the keywords are arbitrarily connected, thus the search must be retrieved in accordance with the polynomial equation. The number of attributes in [16], [18], and [19] affects the amount of search calculations.

2) NUMERICAL EXPERIMENT ANALYSIS

The actual performance of the proposed scheme was analyzed through a simulation on a Windows system with an Intel Core i7 CPU at 2.8GHz and 16GB RAM. The implementation was based on Java pairing-based cryptography. Simulations were performed on [16], [19], and the proposed scheme, and experiments were conducted under the same equipment conditions. The experimental results are shown in Figures 3, 4 and 5.

As shown in Figure 3 the index generation time of the proposed scheme has a linear relationship with the number of attributes. The index calculation time of the proposed scheme is slightly smaller than that of [16] and [19]. The proposed scheme supports the arbitrary encryption of keywords, whereas [16] and [19] support the encryption of individual keywords. When the influence of attributes on index generation time is considered, the index calculation of the proposed scheme is one index less than that of [16], whereas that of [19] is one double-pair operation more than the scheme. The operation of a bilinear pair is larger than the exponential operation. With regard to the experimental multi-keyword encryption, the index generation time is slightly higher than that of [16] and [19].

As shown in Figure 4, the trapdoor calculation is not affected by the change of the number of attributes in the proposed scheme because the trapdoor is encrypted by the key generated by the attribute authority. However, the number of keywords affect the trapdoor. The calculation of trapdoors in [16] and [19] has a linear relationship with the number of attributes. With increase number of attributes, the calculation of trapdoors in [16] and [19] increases linearly. [16] and [19] support single-keyword search, but the trapdoor generation time in [19] is lower than that in [16].

As shown in Figure 5, without considering the data retrieval platform, the data retrieval time in [16], [19], and
the proposed scheme increases linearly with the number of attributes, and the search efficiency of the proposed scheme is higher than that of [16]. The search efficiency is lower than that of [19]. The proposed scheme uses polynomial equations to implement an arbitrarily connected keyword search. In data retrieval, the searcher must calculate polynomial equations, thus the search time of the proposed scheme is higher than [19].

VI. CONCLUSION

Aiming at the problem of malicious doctors uploading false electronic health records and semi-honest search of cloud storage servers, this paper proposes an electronic health record sharing scheme based on keyword search on permissioned blockchains. Secure keyword search and EHRs sharing of encrypted are achieved using permissioned chains. Moreover, the encrypted electronic health record and keyword index are respectively stored in the server and the permissioned blockchain of the hospital, thus achieving privacy protection of electronic health records. The proposed scheme also reduces the storage cost of electronic health records. The scheme not only achieves fine-grained access control but also prevents malicious doctors from uploading false information through the authentication process. In addition, electronic health records sharing among medical institutions is achieved through blockchain technology. The full use of medical information enables doctors to make quick and accurate diagnosis plans for patients and improve the efficiency of hospital work.

REFERENCES

[1] D. X. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on encrypted data,” in Proc. IEEE Symp. Secur. Privacy (S&P), May 2000, pp. 44–55.

[2] D. Cash, S. Jarecki, C. Jutla, H. Krawczyk, M. Rosu, and M. Steiner, “Highly scalable searchable symmetric encryption with support for boolean queries,” in Lecture Notes in Computer Science, vol. 8042. Berlin, Germany: Springer, 2013, pp. 353–373.

[3] S. Jarecki, C. S. Jutla, H. Krawczyk, M. Rosu, and M. Steiner, “Outsourced symmetric private information retrieval,” in Proc. CCS, Nov. 2013, pp. 873–888.

[4] D. Boneh, G. Crescenzo, and R. Ostrovsky, “Public key encryption with keyword search,” in Proc. EUROCRYPT, Interlaken, Switzerland, 2004, pp. 506–522.

[5] V. Goyal, O. Pandey, A. Sahai, and B. Waters, “Attribute-based encryption for fine-grained access control of encrypted data,” in Proc. 13th ACM Conf. Comput. Commun. Secur., Alexandria, VA, USA, 2006, pp. 89–98.

[6] A. Ekblaw, A. Azaria, J. D. A. Halamka, and A. Lippman, “A case study for blockchain in healthcare: MedRec prototype for electronic healthrecords and medical research data,” in Proc. IEEE Open Big Data Conf., Aug. 2016, pp. 1–13.

[7] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, “Integrating blockchain for data sharing and collaboration in mobile healthcare applications,” in Proc. IEEE 28th Annu. Int. Symp. Pers., Indoor, Mobile Commun. (PIMRC), Oct. 2017, pp. 1–5.

[8] S. Nakamoto. Bitcoin:A Peer-to-Peer Electronic Cash System. Accessed: 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[9] G. Wood. Ethereum:A Secure Decentralized Transaction Ledger. Accessed: 2014. [Online]. Available: http://gavwood.com/paper

[10] G. Wood. Ethereum: A Secure Decentralized Generalised Transaction Ledger Eip-150 Revision. Accessed: 2014. [Online]. Available: http://gavwood.com/paper

[11] R. Curtmola, J. Garay, and S. Kamara, “Searchable symmetric encryption: Improved definitions and efficient constructions,” in Proc. ACM Conf. Comput. Commun. Secur., 2006, pp. 79–88.

[12] P. Golle, J. Staddon, and B. Waters, “Secure conjunctive keyword search over encrypted data,” in Proc. 2nd Int. Conf. Appl. Cryptogr. Netw. Secur., Huangshan, China, 2004, pp. 31–45.

[13] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy-preserving multi-keyword ranked search over encrypted cloud data,” in Proc. IEEE INFOCOM, Shanghai, China, Apr. 2011, pp. 829–837.

[14] B. Zhang and F. Zhang, “An efficient public key encryption with conjunctive-subset keywords search,” J. Netw. Comput. Appl., vol. 34, no. 1, pp. 262–267, 2011.

[15] L. Ballard, S. Kamara, and F. Monrose, “Achieving efficient conjunctive keyword searches over encrypted data,” in Proc. 7th Int. Conf. Inf. Comput. Secur., Beijing, China, 2005, pp. 414–426.

[16] U. Qiu, J. Liu, and Y. Shi, “Hidden policy ciphertext-policy attribute-based encryption with keyword search against keyword guessing attack,” Sci. China Inf. Sci., vol. 60, no. 5, 2017, Art. no. 052105.

[17] S. Li and M. Xu, “Attribute-based public encryption with keyword search,” Chin. J. Comput., vol. 37, no. 5, pp. 1017–1024, Jun. 2014.

[18] S. Yan, H. Wei, and C. Dong, “Attribute encryption scheme supporting keyword arbitrary connection search,” J. Commun., vol. 37, no. 8, pp. 77–85, 2016.

[19] W. Sun, S. Yu, W. Lou, Y. T. Hou, and H. Li, “Protecting your right: Verifiable attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud,” IEEE Trans. Parallel Distrib. Syst., vol. 27, no. 4, pp. 1187–1198, Apr. 2016.

[20] Q. Zheng, S. Xu, and G. Atmiesse, “VABKS: Verifiable attribute-based keyword search over outsourced encrypted data,” in Proc. IEEE INFOCOM, Apr./May 2014, pp. 522–530.

[21] S. Wang, Y. Zhang, and Y. Zhang, “A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems,” IEEE Access, vol. 6, pp. 38437–38450, 2018.

[22] S. Wang, D. Zhang, Y. Zhang, and L. Liu, “Efficiently revocable and searchable attribute-based encryption scheme for mobile cloud storage,” IEEE Access, vol. 6, pp. 30444–30457, 2018.

[23] Q. Zhang, S. Wang, D. Zhang, J. Wang, and Y. Zhang, “Time and attribute based dual access control and data integrity verifiable scheme in cloud computing applications,” IEEE Access, vol. 7, pp. 137594–137607, 2019.

[24] G. D. Dagher, J. Mohler, M. Milojkovic, and P. B. Marella, “Anacle: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology,” Sustain. Cities Soc., vol. 39, pp. 283–297, May 2018.

[25] L. Huige, T. Haibo, F. Zhang, and J. He, “Blockchain-based searchable symmetric encryption scheme,” Comput. Electr. Eng., vol. 73, pp. 32–45, Jan. 2018.

[26] Y. Zhang, R. H. Deng, J. Shu, K. Yang, and D. Zheng, “TKSE: Trustworthy keyword search over encrypted data with two-side verifiability via blockchain,” IEEE Access, vol. 6, pp. 31077–31087, 2018.

[27] Q. Xia, E. B. Sifah, A. Smahi, S. Amofa, and X. Zhang, “BBDs: Blockchain-based data sharing for electronic medical records in cloud environments,” Information, vol. 8, no. 2, p. 44, 2017.

SHUFEN NIU was born in Tongwei, Gansu, China, in 1976. She received the B.Sc. degree from Northwest Normal University, in 2000, the M.Sc. degree in operations research and cybernetics from Shanghai University, in 2007, and the Ph.D. degree in cryptography from Northwest Normal University, in 2013. From March 2018 to January 2019, she was a Visiting Scholar with the Department of Computer Science, Georgia State University. She is currently a Teacher with the School of Computer Science and Engineering, Northwest Normal University. She is also an Associate Professor and a Master’s Tutor. In recent years, she has published many excellent articles and hosts several National Natural Science Foundation projects. Her research direction is cryptography. She presided over the Northwest Normal University Young Teacher Research Promotion Program. She received the second prize of scientific and technological progress in colleges and universities in Gansu Province, in 2012, and the third prize of scientific and technological progress in colleges and universities in Gansu Province, in 2013.
LIXIA CHEN was born in Nanchang, Jiangxi, China, in 1992. She received the bachelor’s degree from the School of Information and Electronic Engineering, East China University of Technology, in 2018. She is currently pursuing the master’s degree with the School of Computer Science and Engineering, Northwest Normal University. Her research direction is cryptography.

JINFENG WANG was born in Tianshui, Gansu, China, in 1992. She received the bachelor’s degree from the School of Information and Computer Science, Lanzhou University of Finance and Economics, in 2016. She is currently pursuing the master’s degree with the School of Computer Science and Engineering, Northwest Normal University. Her research direction is cryptography.

FEI YU was born in Jining, Shandong, China, in 1997. He received the bachelor’s degree from the School of Software, Jishou University, in 2019. He is currently pursuing the master’s degree with the School of Computer Science and Engineering, Northwest Normal University. His research direction is cryptography.

* * *