IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON CYBER-THREATS AND COUNTERMEASURES IN THE HEALTHCARE SECTOR

Received May 14, 2018, accepted July 1, 2018, date of publication July 4, 2018, date of current version July 25, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2852784

Secure Identity-Based Data Sharing and Profile Matching for Mobile Healthcare Social Networks in Cloud Computing

QINLONG HUANG, (Member, IEEE), WEI YUE, YUE HE, AND YIXIAN YANG

School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Qinlong Huang (longsec@bupt.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB0800605, in part by the National Natural Science Foundation of China under Grant 61572080, in part by the Key Program of Joint Funds of the National Natural Science Foundation of China under Grant U1736212, and in part by the China Scholarship Council under Grant 201806475007.

ABSTRACT Cloud computing and social networks are changing the way of healthcare by providing real-time data sharing in a cost-effective manner. However, data security issue is one of the main obstacles to the wide application of mobile healthcare social networks (MHSNs), since health information is considered to be highly sensitive. In this paper, we introduce a secure data sharing and profile matching scheme for the MHSN in cloud computing. The patients can outsource their encrypted health records to cloud storage with an identity-based broadcast encryption technique, and share them with a group of doctors in a secure and efficient manner. We then present an attribute-based conditional data re-encryption construction which permits the doctors who satisfy the pre-defined conditions in the ciphertext to authorize the cloud platform to convert a ciphertext into a new ciphertext of an identity-based encryption scheme for specialist without leaking any sensitive information. Furthermore, we provide a profile matching mechanism in the MHSN based on identity-based encryption with an equality test, which helps patients to find friends in a privacy-preserving way and achieves flexible authorization on the encrypted health records with resisting the keywords guessing attack. Moreover, this mechanism reduces the computation cost on the patient side. The security analysis and experimental evaluation show that our scheme is practical for protecting the data security and privacy in the MHSN.

INDEX TERMS Conditional proxy re-encryption, data security, encryption, health information management, profile matching.

I. INTRODUCTION

Mobile healthcare is an innovative combination of mobile devices and mobile communication technologies, for it can provide necessary health information, routine care improvements, potential infectious disease prevention, health interventions, etc. It is getting more and more widely to apply the emerging cloud computing technology into the fields of mobile healthcare. By using mobile healthcare system, the electronic health record (EHR) can be transmitted over the network to the cloud service provider (CSP) for remote storage. Moreover, the healthcare providers can read it from an end device or access it remotely using a mobile device to provide real-time medical treatment [1]. Meanwhile, people tend to share and disseminate the healthcare information via social networks, since social media is an extension of the healthcare professional and patient relationship. Consequently, mobile healthcare social networks (MHSN) are created for connecting patients so that they could share healthcare information using their mobile devices, and also connecting doctors and specialists for better healthcare. For example, people in MHSN can communicate and interact with each other before making healthcare decision.

However, data security issues are the major obstacles to the application of MHSN [2]. As we all know, health information such as treatment and drug information is considered to be highly sensitive. If these data are outsourced to the CSP, the patients cannot directly control the software or hardware platform for storing data. Without careful consideration, patients may suffer serious medical information leakage from the cloud. For example, millions of EHRs have been

compromised in recent years. Hence, it is significant that the EHRs should be stored in an encrypted form. Even if the CSP is untrusted or compromised, the data maintains security and privacy. Simultaneously, the encrypted records should be shared and accessed in a reasonable way.

Currently, there are many techniques utilized to protect data security in MHSN, such as public-key encryption (PKE), identity-based encryption (IBE), identity-based broadcast encryption (IBBE) [3] and attribute-based encryption (ABE) [4]. In an IBBE system, broadcaster can dynamically select a specific group of users, and then encrypt the message thus only the selected users can decrypt. In an ABE system, secret key and ciphertext are associated with a set of attributes or an access policy. Although there are some schemes which apply ABE to encrypt the EHRs in MHSN, it is not convenient because of the heavy encryption and decryption cost. Let us consider a MHSN system with IBE in cloud computing, Alice may encrypt healthcare information using doctor Bob's identity and then outsource the ciphertext to the CSP, while Bob can obtain his secret key from the trusted authority (TA) and then access Alice's health data. In this way, IBE can achieve simple but efficient access control over the sensitive health data. However, when doctor Bob encounters an uncertain problem, he may have a consultation with a specialist for advice or treatment. Since the relevant records stored in the cloud were encrypted, the specialist cannot directly decrypt the ciphertexts. In order to solve this issue, proxy re-encryption (PRE) was employed [5], which can transform a ciphertext under Alice's identity into a ciphertext for Bob. By using identity-based PRE (IBPRE), the specialist in MHSN can access the EHRs without the CSP getting any useful information.

The MHSN also provides strong social interconnection functions since the patients can communicate with others. Specially, patients can find similar patients with the profile matching mechanism and communicate their illness symptoms and medications. However, the patients may disclose their sensitive health information to other users, including the users being matched with. Therefore, it is essential to protect the patients' personal information during the match process, otherwise malicious users may easily collect and use this information. Recently, many researchers have applied the equality test technique to achieve profile matching in cloud and social networks. However, there might be keywords guessing attack, especially in the medical system with limited keywords. Therefore, the attack is more likely to be successful in MHSN and may cause serious privacy leakage.

In order to protect data confidentiality and availability, and also preserve the patients' privacy in MHSN, encryption techniques must be adopted. In this study, a secure and efficient data sharing and profile matching scheme for MHSN in cloud computing is introduced. Our contributions are summed up as follows.

(1) We propose a secure identity-based data sharing scheme for MHSN, which allows patients to outsource their encrypted health records to CSP with IBBE technique, and share them with a group of doctors in a secure and efficient manner.

(2) We present an attribute-based conditional data re-encryption construction, which permits doctors who satisfy the pre-defined conditions in the ciphertext to authorize the CSP to re-encrypt the ciphertext for specialist, without leaking any sensitive information.

(3) We provide an efficient profile matching mechanism in MHSN based on IBE with equality test (IBEET), that helps patients to find friends in a privacy-preserving manner, and achieve flexible authorization on the encrypted health records with resisting the keywords guessing attack.

The rest of the paper is organized as follows. We introduce related work in section II. Then, the preliminaries are presented in section III. Next, we introduce the system model and system definition in section IV and give detailed construction of our scheme in section V. We present the security analysis and performance evaluation in section VI and VII. Finally, we summarize the paper in section VIII.

II. RELATED WORK

A. HEALTH RECORDS ENCRYPTION

A fundamental security requirement of MHSN is that EHRs should be encrypted to guarantee data confidentiality. Many encryption schemes were proposed to protect data security in mobile healthcare system. Li *et al.* [6] presented an access control framework over EHRs, that utilizes ABE to encrypt each patient's data. Barua *et al.* [7] proposed ESPAC which also utilizes ABE to achieve patient-centric access control. Yu *et al.* [8] exploited key-policy ABE (KP-ABE) technique to protect the EHRs in cloud computing. Although ABE can encrypt the data and achieve fine-grained access control over the ciphertext, it suffers from the inconvenience of heavy computation cost in encryption and decryption phases. It becomes even worse in the case of resource-limited healthcare devices, such as wearable devices and mobile terminals. Liu *et al.* [9] introduced an outsourced EHR access control scheme which allows data owner to complete most of encryption computation in advance and then generate the ciphertext with very low computation cost. Similar with this scheme, the recent ABE-based schemes [10], [11] also outsourced most of the expensive cryptographic computations to the CSP to reduce computational overhead of user-side. However, extra communication cost is inevitably brought in these schemes.

In order to guarantee both secure EHRs sharing and high comprehensive performance, many IBE-based schemes were proposed in MHSN system, since the IBE mechanism can use any valid string such as unique id as the public key, and reduce the computation cost of patient. Li *et al.* [12] employed IBE and identity-based signature techniques to protect healthcare data in cloud computing. Tan *et al.* [13] developed a lightweight IBE scheme suitable for sensors for healthcare monitoring. Wang *et al.* [14] constructed a new IBE scheme for secure and cost-effective EHRs sharing in mobile healthcare system in cloud computing.

However, these schemes may encounter repeated encryption when there are a group of accessors.

B. IDENTITY-BASED PROXY RE-ENCRYPTION

The cryptographic algorithm PRE was proposed by Blaze *et al*. [15] for secure data dissemination. Especially, IBPRE allows a proxy to transform a delegator's ciphertext into a delegatee's ciphertext. The first IBPRE was established by Green and Ateniese in [5], which is proved to be chosen ciphertext attack (CCA) secure. Matsuo [16] proposed a new PRE systems which can convert a ciphertext encrypted using a traditional PKE scheme to a ciphertext encrypted by IBE scheme. Zhou *et al*. [17] proposed an IBPRE construction which allows the proxy to convert a ciphertext of an IBBE scheme into a ciphertext of an IBE scheme. Recently, Wang *et al*. [14] showed how to integrate IBPRE into healthcare system in cloud computing, in which the doctors can delegate a key to the CSP so that the stored ciphertext can be transformed into a new one for the intended specialist.

However, the above mentioned PRE-based schemes could not control the process of data re-encryption. Weng *et al*. [18] proposed the first conditional PRE (CPRE) construction, that encrypts data with a key condition, and re-encrypts the ciphertext only if the key meets this defined condition. Xu *et al*. [19] proposed a conditional identity-based broadcast PRE scheme in cloud computing, which can transform an IBBE ciphertext into another IBBE ciphertext if the condition is satisfied. In order to support expressive conditions rather than keywords, attribute-based CPRE was proposed. Liang *et al*. [20] proposed an attribute-based PRE scheme in which if the original access policy is satisfied, the proxy can convert a ciphertext under an access policy to another ciphertext under a new access policy. Yang *et al*. [21] deployed an access policy in PKE-encrypted ciphertext and generated the re-encryption key with attribute set. This scheme allowed the ciphertext to be re-encrypted only if the access policy is satisfied.

C. PROFILE MATCHING IN CLOUD AND SOCIAL NETWORKS

Profile matching is an efficient method of comparing different users' personal profiles in cloud and social networks. However, the user's profile may contain sensitive information, so attention should be paid to ensure that private information is not leaked. Two mainstreams of ways were proposed. The first way considers the user profile as a set of attributes. It uses private set intersection to achieve attribute matching based on secret sharing and homomorphic encryption. In order to exchange the minimal private information of participating users, Li *et al* [23] utilized secret sharing technique to help the user to find friend whose profile best matches with her from a group of users. The second way measures the social proximity by taking the user profile as a vector. Zhang *et al*. [24] proposed a private matching protocol in mobile social networks, which allows subtle difference between users and supports a wide range of matching schemes for matching metrics. Zhang *et al*. [25] proposed a new privacy-preserving configuration profile matching mechanism based on symmetric encryption without any trusted third party.

However, all these strategies were based on plaintext and most of them were based on the overall similarity of users' data. To achieve the goal of profile matching over encrypted data, Qiu *et al*. [26] utilized identity-based cryptosystem to allow the CSP to conduct private matching by partially decrypting the ciphertexts to an intermediate form with authorized token. For the first time, Ma *et al*. [27] employed IBE and PKE with equality test (PKEET), and proposed the concept of IBEET that enables the CSP to store the ciphertexts encrypted by IBE scheme and perform equality test on them. To reduce the computational costs of encryption and test phases, Wu *et al*. [28] proposed an efficient IBEET scheme in smart city by reducing the time-consuming hash computation. In this scheme, the trapdoor is generated with a particular keyword, which indicates that it cannot be used to match any messages. Further, Wu *et al*. [29] adopted the IBEET scheme to protect EHRs in cloud computing, which can do search on outsourced encrypted EHRs and determine whether two different ciphertexts contain the same record.

However, little attention has been devoted to privacy protection of equality test. If an attacker can do the test on any ciphertexts without permission, it may leak more information about the ciphertexts. Aiming to support flexible authorization, Ma *et al*. [30] proposed a novel PKEET scheme which permits the user to control the comparison of its ciphertexts with others'. For example, the ciphertext to ciphertext authorization allows that a specific ciphertext could be compared with another ciphertext. Hence, server needs to execute the corresponding test algorithm according to authorization policy.

III. PRELIMINARIES

A. BILINEAR MAP

Let $\mathbb{G}_0$ and $\mathbb{G}_T$ be two groups with the prime order p. A map $e : \mathbb{G}_0 \times \mathbb{G}_0 \to \mathbb{G}_T$ with the following properties is said to be bilinear [31]:

(1) Bilinearity: For all $g, h \in \mathbb{G}_0$ and $a, b \in \mathbb{Z}_p$, we have $e(g^a, h^b) = e(g, h)^{ab}$.

(2) Non-degeneracy: There exists $g, h \in \mathbb{G}_0$ such that$e(g, h) \neq 1$.

(3) Computability: There is a polynomial time algorithm to compute $e(g, h) \in \mathbb{G}_T$ for any $g, h \in \mathbb{G}_0$.

B. IDENTITY-BASED ENCRYPTION

An IBE scheme consists of the following algorithms [32].

(1) $(MK, PK) \leftarrow Setup(1^\lambda)$. The setup algorithm inputs a security parameter λ, and outputs the master secret key MK and public parameters PK.

(2) $SK \leftarrow KeyGen(MK, PK, ID)$. The key generation algorithm inputs the master secret key MK, public parameters PK and an identity ID. It outputs a secret key SK.

(3) $CT \leftarrow Enc(ID, PK, M)$. The encryption algorithm inputs an identity ID, public parameters PK and a plaintext M. It outputs a ciphertext CT.

(4) $M \leftarrow Dec(SK, PK, CT)$. The decryption algorithm inputs a secret key SK, PK and a ciphertext CT, and outputs M.

C. ACCESS TREE

Let T be a tree representing an access policy, in which each non-leaf node x represents a threshold gate [31]. The functions and terms of T are defined as follows.

(1) num_x. It denotes the number of children of a node x.

(2) k_x. It denotes the threshold value of a node x.

(3) $attr_x$. It denotes an attribute associated with a leaf node x.

(4) $parent(x)$. It represents the parent node of a node x.

(5) $index(x)$. It returns the index value of node x.

(6) T_x. It denotes the sub-tree rooted at node x in T. If the attribute set S satisfies T_x, we denote it as $T_x(S) = 1$. Then it computes as follows. If x is a leaf node and $attr_x \in S$, $T_x(S)$ returns 1. If x is a non-leaf node, it computes $T_n(S)$ for all children n of node x, and returns 1 if at least k_x children return 1.

D. ATTRIBUTE-BASED ENCRYPTION

The main algorithms of an ABE system with tree-based access policy are summarized as follows [31].

(1) $(MK, PK) \leftarrow Setup(1^{\lambda})$: The algorithm inputs a security parameter λ, and outputs a master secret key MK and a public key PK.

(2) $AK \leftarrow KeyGen(MK, PK, S)$: The algorithm inputs the master secret key MK, public key PK and a set S of attributes. It outputs an attribute key AK corresponding to the attribute set.

(3) $CT \leftarrow Enc(T, PK, M)$: The encryption algorithm inputs an access policy T, the public key PK and a message M. It outputs a ciphertext CT.

(4) $M/\perp \leftarrow Dec(AK, PK, CT)$: The algorithm inputs an attribute key AK, the public key PK and a ciphertext CT with an access policy T. It outputs message M if $S \in T$ or $\perp$.

IV. SCHEME OVERVIEW

A. SYSTEM MODEL

Our proposed secure identity-based data sharing and profile matching model for MHSN in cloud computing is shown in Fig. 1, including five entities: central authority, CSP, patient, doctor and specialist.

1) CENTRAL AUTHORITY

The central authority is trusted for initializing the system and generating attribute keys and secret keys for participating users.

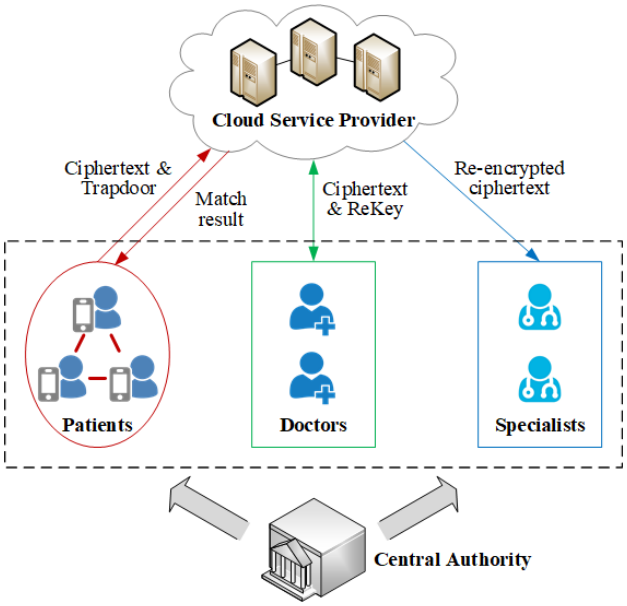

FIGURE 1. System model.

2) CSP

The CSP is responsible for data storage and can be acted as a proxy as it is semi-trusted. Besides, the CSP performs the profile matching for patients.

3) PATIENT

The patients register the system to obtain their secret keys with their identities. They encrypt the EHRs using IBBE algorithm and outsource the ciphertexts to CSP, hence only authorized doctors could decrypt them. Simultaneously, patients with the same symptom can generate trapdoors and form social relationships according to their wills.

4) DOCTOR

The authorized doctors can decrypt the patients' ciphertext that stored in the CSP. When encountering a problem that needs to negotiate with a specialist, the doctor can generate a re-encryption request, thus the CSP converts the ciphertext into an IBE-encrypted data for specialist if the doctor satisfies the pre-defined conditions in the ciphertext.

5) SPECIALIST

The specialist could decrypt the re-encrypted ciphertext with the secret key and then assist doctors for advice.

During the processes of data sharing in MHSN, the patients may want to make friends with others who have the same symptoms through the CSP, and they do not want to leak their private information. For example, Alice and Bob have similar symptoms and may want to link up to exchange the symptom information or encourage each other. Alice and Bob first encrypt their EHRs respectively and share with their doctors. Meanwhile, they generate the trapdoors, and the CSP could

execute the profile matching algorithm without knowing the specific information. The CSP sends the match result to Alice and Bob, so they can establish social relationship.

B. FLEXIBLE AUTHORIZATION

In order to withstand the keywords guessing attack and strengthen the privacy protection, flexible authorization is considered in our scheme. We describe three types of authorization as follows.

(1) User to user authorization. Alice and Bob generate trapdoors on their all ciphertexts respectively.

(2) User to ciphertext authorization. Alice generates a trapdoor on her all ciphertext, while Bob generates a trapdoor on his specific ciphertext. Suppose that Bob suffers from headache and stomachache, but he only wants to find a social friend who has a stomachache. Then he will generate a trapdoor on the stomachache and send it to the CSP.

(3) Ciphertext to ciphertext authorization. Alice and Bob may have more than one symptom. They generate a trapdoor on one of their ciphertexts according to their inclinations. It also means that they may prefer to find some social friends with part of their symptoms.

C. SYSTEM DEFINITION

Definition 1: With the above system model and authorization type, our scheme consists of the following algorithms.

(1) *Setup*(1^λ): The algorithm inputs a security parameter λ, outputs a master secret key MK and a public key PK.

(2) *KeyGen*-1(MK, PK, ID): The algorithm inputs the master secret key MK and public key PK, the identity ID of a user. It returns the secret key SK of this user.

(3) *KeyGen*-2(MK, PK, S): The algorithm inputs the master secret key MK and public key PK, a set S of attributes. It returns the attribute key AK for the doctors.

(4) *Enc*(U, T, PK, M): The algorithm inputs a set U of doctors' identities, an access policy T, public keyPK and a data M. It returns an initial ciphertext CT.

(5) *ReKeyGen*(SK, AK, ID', PK, ID_s): The algorithm inputs the secret key SK and attribute key AK of a doctor with identity ID', public keyPK and a specialist's identity ID_s. It returns a re-encryption key RK.

(6) *ReEnc*(RK, ID', PK, CT): The algorithm inputs the re-encryption key RK of a doctor with identity ID', public keyPK and an initial ciphertext CT. It returns a re-encrypted ciphertext CT'.

(7) *Dec*-1(SK, ID', PK, CT): The algorithm inputs a secret key SK for a doctor with identity ID', public keyPK and an initial ciphertext CT. It returns M if ID' is included in the identity set U.

(8) *Dec*-2(SK, ID_s, PK, CT'): The algorithm inputs a secret key SK for a specialist with identity ID_s, public key PK and a re-encrypted ciphertext CT'. If ID_s is included in the identity set U, it returns M.

(9) *TrapGen*-1(SK, PK): The algorithm inputs a patient's secret key SK and public keyPK. It returns a trapdoor TD for all the ciphertexts of the patient.

(10) *TrapGen*-2(SK, PK, CT): The algorithm inputs a patient's secret key SK, the public keyPK and an initial ciphertext CT. It returns a trapdoor TD' for the patient.

(11) *Test*-1(TD_a, CT_a, TD_b, CT_b): The algorithm inputs a trapdoor TD_a of a patient with ID_a, a ciphertext CT_a of a patient with ID_a, a trapdoor TD_b of a patient with ID_b, a ciphertext CT_b of a patient with ID_b. It returns true if CT_a and CT_b contain the same data.

(12) *Test*-2(TD_a, CT_a, TD'_b): The algorithm inputs a trapdoor TD_a of a patient with ID_a, a ciphertext CT_a of a patient with ID_a, a trapdoor TD'_b associated with a patient with ID_b and a ciphertext CT_b. It returns true if CT_a and CT_b contain the same data.

(13) *Test*-3(TD'_a, TD'_b): The algorithm inputs a trapdoor TD'_a associated with a patient with ID_a and a ciphertext CT_a, a trapdoor TD'_b associated with a patient with ID_b and a ciphertext CT_b. It returns true if CT_a and CT_b contain the same data.

V. CONSTRUCTION

A. SYSTEM SETUP

Let $e : \mathbb{G}_0 \times \mathbb{G}_0 \to \mathbb{G}_T$ be a bilinear map, $\mathbb{G}_0$ and $\mathbb{G}_T$ be two groups with prime order p. The central authority runs *Setup* algorithm to choose $g, h, u, v \in \mathbb{G}_0$, γ, $\beta \in \mathbb{Z}_p$ randomly. Three cryptographic hash functions $H_1 : \{0,1\}^* \to \mathbb{Z}_p^*$, $H_2 : \{0,1\}^* \to \mathbb{G}_0$ and $H_3 : \mathbb{G}_T \to \mathbb{G}_0$ are used, and a maximum number of doctors N is chosen. The central authority outputs a master secret key $MK = (g, \gamma, \beta, \lambda)$ and a system public key $PK = (g^\gamma, e(g,h), e(g,h)^\gamma, h, h^\gamma, \ldots, h^{\gamma^N}, u, u^\gamma, \ldots, u^{\gamma^N}, h^\beta, u^\beta, h^\lambda, v, e(v,h))$.

B. KEY GENERATION

The central authority generates the secret key SK for each entity with identity ID by running *KeyGen*-1 algorithm.

$$SK = (K_0 = g^{1/(\gamma + H_1(ID))}, K_1 = v^{1/(\lambda + H_1(ID))})$$

For the doctor with attribute set S, the central authority runs *KeyGen*-2 algorithm to choose a random $\alpha \in \mathbb{Z}_p$. Then central authority random chooses $r_j \in \mathbb{Z}_p$ for each attribute $j \in S$, and outputs the attribute key AK.

$$AK = (D_0 = g^{(\gamma+\alpha)/\beta}, \{D_j = g^\alpha H_2(j)^{r_j}, D'_j = h^{r_j}\}_{j \in S})$$

C. DATA ENCRYPTION

In order to protect the security of data M in MHSN, the patient runs *Enc* algorithm and outsources the result to the CSP. First, the patient chooses a random EK to encrypt M with symmetric encryption algorithm SE, and generates result $C_0 = SE_{EK}(M)$. Then the patient chooses a set U of doctors' identities, and picks a random $k \in \mathbb{Z}_p$ to protect the EK based on IBBE [3]. It computes as follows.

$$C_1 = EK \cdot e(g,h)^k, \quad C_2 = g^{-\gamma k},$$
$$C_3 = h^{k \cdot \prod_{ID_i \in U} (\gamma + H_1(ID_i))}, \quad C'_4 = u^{k \cdot \prod_{ID_i \in U} \frac{\gamma + H_1(ID_i)}{H_1(ID_i)}}$$

Then the patient customizes an access policy T and chooses a polynomial p_x for each node x (including the leaves) in the access tree. We set the degree d_x of polynomial p_x to be one less than the threshold value k_x of node x, that is $d_x = k_x - 1$. These polynomials are chosen in a top-down manner. Beginning with root node R, the patient chooses a random t and sets $p_R(0) = t$, and then chooses d_R other points of the polynomial p_R randomly to define it completely. For any other node x, it sets $p_x(0) = p_{parent(x)}(index(x))$ and randomly chooses d_x other points to completely define p_x. Let Y be the set of leaf nodes in T, the patient computes

$$C_4 = u^{\beta t} \cdot C_4', C_5 = h^{\beta t}, \quad C_6 = e(g,h)^{\gamma t},$$
$$C_7 = \{\tilde{C}_y = h^{p_y(0)}, \quad \tilde{C'}_y = H_2(attr_y)^{p_y(0)}\}_{y \in Y}$$

The patient chooses a random $l \in \mathbb{Z}_p$, and then generates $C_8 = h^{l(\lambda + H_1(ID))}$, $C_9 = v^l H_2(M) \oplus H_3(e(v,h)^l)$.

Finally, the patient outputs the initial ciphertext CT.

$$CT = (C_0, C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9)$$

D. RE-ENCRYPTION KEY GENERATION

If the doctor needs to collaborate with the specialist with identity ID_s, he then runs *ReKeyGen* algorithm to pick $s \in \mathbb{Z}_p$ randomly and computes the following with his identity ID' and secret key SK.

$$R_1 = K_0 \cdot u^{s/H_1(ID')} = g^{1/(\gamma + H_1(ID'))} \cdot u^{s/H_1(ID')},$$
$$R_2 = IBE.Enc(ID_s, PK, h^s), \quad R_3 = D_0 \cdot u^s = g^{(\gamma+\alpha)/\beta} \cdot u^s$$

Then the doctor generates the result with attribute key AK.

$$R_4 = \{\tilde{R}_j = D_j = g^{\alpha} H_2(j)^{r_j}, \tilde{R'}_j = D'_j = h^{r_j}\}_{j \in S}$$

Finally, the doctor outputs the re-encryption key $RK = (R_1, R_2, R_3, R_4)$.

E. DATA RE-ENCRYPTION

The CSP runs *ReEnc* algorithm to re-encrypt CT with the re-encryption request from the authorized doctor. First, the CSP generates

$$\begin{aligned} C_1' &= C_1 \cdot (e(C_2, h^{\Delta_\gamma(ID',U)}) \cdot e(C_3, R_1))^{\frac{-1}{\prod_{ID_i \in U \wedge ID_i \neq ID'} H_1(ID_i)}} \\ &= C_1 \cdot e(g,h)^{-k} \\ &\quad \cdot e(u^{s/H_1(ID')}, h^{k \cdot \prod_{ID_i \in U} (\gamma + H_1(ID_i)) \frac{-1}{\prod_{ID_i \in U \wedge ID_i \neq ID'} H_1(ID_i)}}) \\ &= EK \cdot e(u^s, h^{-k})^{\prod_{ID_i \in U} \frac{\gamma + H_1(ID_i)}{H_1(ID_i)}} \end{aligned}$$

Then, the CSP runs *DecryptNode* algorithm which takes CT, RK associated with attribute set S, and a node x in T as input. If the node x is a leaf node, then we let $a = attr_x$ and compute as bellow. If $a \notin S$, we have $DecryptNode(RK, CT, x) = null$. Otherwise, the algorithm is computed as follows.

$$\begin{aligned} DecryptNode(RK, CT, x) &= \frac{e(\tilde{R}_i, \tilde{C}_x)}{e(\tilde{R'}_i, \tilde{C'}_x)} \\ &= \frac{e(g^{\alpha} H_2(a)^{r_i}, h^{p_x(0)})}{e(h^{r_i}, H_2(a)^{p_x(0)})} \\ &= e(g,h)^{\alpha p_x(0)} \end{aligned}$$

On the other hand, if x is a non-leaf node, the algorithm $DecryptNode(RK, CT, x)$ is defined as follows. For all children n of node x, it runs $DecryptNode(RK, CT, n)$ and denotes the result as F_n. Let S_x be an arbitrary k_x-sized set of child nodes n such that $F_n \neq null$. If S_x does not exist, then $F_n = null$. Otherwise, we compute

$$\begin{aligned} F_x &= \prod_{n \in S_x} F_n^{\Delta_{j,S'_x}(0)}, \quad where \ {}^{j=index(n)}_{S'_x = \{index(n): n \in S_x\}} \\ &= \prod_{n \in S_x} (e(g,h)^{\alpha p_{parent(n)}(index(n))})^{\Delta_{j,S'_x}(0)} \\ &= \prod_{n \in S_x} e(g,h)^{\alpha p_x(j) \cdot \Delta_{j,S'_x}(0)} \\ &= e(g,h)^{\alpha p_x(0)}. \end{aligned}$$

If S satisfies the whole access tree, the CSP generates the result $A = DecryptNode(RK, CT, R) = e(g,h)^{\alpha t}$ and computes

$$C_2' = \frac{e(C_5, R_3)}{C_6 \cdot A} = \frac{e(h^{\beta t}, g^{(\gamma+\alpha)/\beta} \cdot u^s)}{e(g,h)^{\gamma t} \cdot e(g,h)^{\alpha t}} = e(h^{\beta t}, u^s)$$

Finally, the CSP outputs the re-encrypted ciphertext.

$$CT' = (C_0' = C_0, C_1', C_2', C_3' = R_2, C_4' = C_4)$$

F. DATA DECRYPTION

If the ciphertext is an initial ciphertext CT, the doctor computes the following by running the *Dec*-1 algorithm if his identity ID' is included in U.

$$\begin{aligned} K &= (e(C_2, h^{\Delta_\gamma(ID',U)}) \cdot e(K_0, C_3))^{\frac{1}{\prod_{ID_i \in U \wedge ID_i \neq ID'} H_1(ID_i)}} \\ &= (e(g^{-\gamma k}, h^{\Delta_\gamma(ID',U)}) \\ &\quad \cdot e(g^{1/(\gamma + H_1(ID'))}, h^{k \cdot \prod_{ID_i \in U} (\gamma + H_1(ID_i))}))^{\frac{1}{\prod_{ID_i \in U \wedge ID_i \neq ID'} H_1(ID_i)}} \\ &= e(g,h)^k \end{aligned}$$

Then, the doctor recovers symmetric key $EK = C_1/K$ and decrypts data M using algorithm SE. On the other hand, if the ciphertext is a re-encrypted one, specialist runs *Dec*-2 algorithm to compute $h^s = IBE.Dec(K_0, PK, C_3')$, and then computes the following

$$\begin{aligned} EK &= \frac{C_1' \cdot e(h^s, C_4')}{C_2'} \\ &= \frac{EK \cdot e(u^s, h^{-k})^{\prod_{ID_i \in U} \frac{\gamma + H_1(ID_i)}{H_1(ID_i)}} \cdot e(h^s, u^{\beta t + k \cdot \prod_{ID_i \in U} \frac{\gamma + H_1(ID_i)}{H_1(ID_i)}})}{e(h^{\beta t}, u^s)} \end{aligned}$$

Then, the specialist could recover data M.

G. TRAPDOOR GENERATION

In order to match the profile, the patient generates a trapdoor according to his authorization type. If the patient prefers to match his all ciphertexts, he runs *TrapGen*-1 algorithm to generate a trapdoor with his secret key as follows.

$$TD_i = K_{i,1} = v^{1/(\lambda+H_1(ID_i))}$$

Otherwise, the patient runs *TrapGen*-2 algorithm to compute the following with his secret key and a specific ciphertext.

$$E_i = e(K_{i,1}, C_{i,8}) = e(v^{1/(\lambda+H_1(ID_i))}, h^{l_i(\lambda+H_1(ID_i))}) = e(v,h)^{l_i}$$

Then the patient generates a trapdoor $TD'_i = E_i$.

H. PROFILE MATCHING

The CSP determines whether the ciphertexts of two patients contain the same data according to the authorization type.

(1) User to user authorization. The CSP runs *Test*-1 algorithm with $TD_a = v^{1/(\lambda+H_1(ID_a))}$ and $TD_b = v^{1/(\lambda+H_1(ID_b))}$, and computes E_a and E_b with chosen ciphertexts CT_a and CT_b.

$$\begin{aligned} E_a &= e(TD_a, C_{a,8}) \\ &= e(v^{1/(\lambda+H_1(ID_a))}, h^{l_a(\lambda+H_1(ID_a))}) = e(v,h)^{l_a} \\ E_b &= e(TD_b, C_{b,8}) \\ &= e(v^{1/(\lambda+H_1(ID_b))}, h^{l_b(\lambda+H_1(ID_b))}) = e(v,h)^{l_b} \end{aligned}$$

Then, the CSP computes

$$\begin{aligned} X_a &= C_{a,9} \oplus H_3(E_a) = v^{l_a} H_2(M) \\ X_b &= C_{b,9} \oplus H_3(E_b) = v^{l_b} H_2(M) \end{aligned}$$

The CSP outputs 1 if $E_a \cdot e(h, X_b) = E_b \cdot e(h, X_a)$ holds.

(2) User to ciphertext authorization. The CSP runs *Test*-2 algorithm with $TD_a = v^{1/(\lambda+H_1(ID_a))}$ and $TD'_b = E_b$, and computes E_a with chosen ciphertext CT_a.

$$E_a = e(TD_a, C_{a,8}) = e(v^{1/(\lambda+H_1(ID_a))}, h^{l_a(\lambda+H_1(ID_a))}) = e(v,h)^{l_a}$$

Then, the CSP computes

$$\begin{aligned} X_a &= C_{a,9} \oplus H_3(E_a) = v^{l_a} H_2(M) \\ X_b &= C_{b,9} \oplus H_3(E_b) = v^{l_b} H_2(M) \end{aligned}$$

The CSP outputs 1 if $E_a \cdot e(h, X_b) = E_b \cdot e(h, X_a)$ holds.

(3) Ciphertext to ciphertext authorization. The CSP runs *Test*-3 algorithm with $TD'_a = E_a$ and $TD'_b = E_b$, and computes

$$\begin{aligned} X_a &= C_{a,9} \oplus H_3(E_a) = v^{l_a} H_2(M) \\ X_b &= C_{b,9} \oplus H_3(E_b) = v^{l_b} H_2(M) \end{aligned}$$

The CSP outputs 1 if $E_a \cdot e(h, X_b) = E_b \cdot e(h, X_a)$ holds.

VI. SECURITY ANALYSIS

Theorem 1: If an adversary $\mathcal{A}$ breaks our scheme, we can build an algorithm B which interacts with A to break the selective CCA-security of IBBE scheme [3].

Proof: The adversary $\mathcal{A}$ can query the re-encryption for the chosen identity sets. In order to respond to the *ReKeyGen* queries of $\mathcal{A}$, algorithm B needs to call the key generation oracle of IBBE scheme and then run the *ReKeyGen* algorithm with the output keys to get the requested re-encryption keys. With the proof in [17] and the security of ABE [4], B cannot respond by giving the queried key if it is queried by $\mathcal{A}$ to generate a re-encryption key for the challenge identity. If $\mathcal{A}$ has an advantage in breaking the CCA-security of our scheme, B can break the security of IBBE with this advantage.

Theorem 2: Our scheme is collusion-resistant against colluding doctors based on the security of ABE.

Proof: For the purpose of re-encrypting the ciphertext stored in CSP, the authorized doctor must recover $e(g,h)^{\alpha t}$. If an attacker does not hold enough attributes, he may run *DecryptNode* algorithm with some colluding patient's re-encryption key *RK*. However, the *RK* is generated with a random and unique α defined by trusted central authority. Hence, the attacker cannot generate the correct C'_2 and the re-encrypted ciphertext by collusion attack.

Theorem 3: Our scheme is one-way chosen-ciphertext secure against a chosen identity attack (OW-ID-CCA).

Proof: As proved in [27], if $\mathcal{A}$ is an OW-ID-CCA adversary that has advantage against our scheme, then there is an one-way chosen-ciphertext security (OW-CCA) adversary B that has advantage against PKE scheme. Hence, the OW-ID-CCA attack on our scheme can be converted to an OW-CCA attack on PKE scheme [33]. However, the PKE scheme is OW-CCA secure under bilinear Diffie-Hellman assumption, thus our scheme is OW-ID-CCA secure, which guarantees that the ciphertexts cannot be decrypted by CSP during the test process.

VII. PERFORMANCE EVALUATIONS

A. FUNCTIONALITY COMPARISON

We first compare our scheme with several PRE-based data sharing schemes and two recent representative IBEET schemes in terms of data confidentiality, conditional re-encryption, re-encrypted data confidentiality, profile matching and flexible authorization etc. The result is shown in Table 1.

First of all, Zhou *et al.* [17], Xu *et al.* [19] and our scheme adopt IBBE to share confidential data with a group of users efficiently. With the PRE technique, the compared schemes can share ciphertext with other users by re-encrypting the ciphertext via proxy. However, Zhou *et al.* [17], Wang *et al.* [14] and Liang *et al.* [20] are not practical in MHSN, since the doctor authorized by a patient can re-encrypt all the healthcare data of this patient. Xu *et al.* [19], Yang *et al.* [21] and our scheme support conditional data re-encryption. Specially, Yang *et al.* [21]

TABLE 1. Functional comparison.

Functionality	Zhou et al. [17]	Wang et al.[14]	Liang et al. [20]	Xu et al. [19]	Yang et al. [21]	Wu et al. [29]	Qiu et al. [26]	Ma et al. [27]	Our scheme
Data confidentiality	IBBE	IBE	ABE	IBBE	PKE	IBE	IBE	IBE	IBBE
Conditional Re-encryption	No	No	No	keyword	Access policy	No	No	No	Access policy
Re-encrypted data confidentiality	IBE	IBE	ABE	IBBE	PKE	-	-	-	IBE
Profile matching	-	-	-	-	-	Yes	Yes	Yes	Yes
Flexible authorization	-	-	-	-	-	No	Yes	Yes	Yes

and our scheme adopt ABE technique which supports complex operations to represent flexible condition set in MHSN. Further, Wu *et al*. [29], Qiu *et al*. [26], and Ma *et al*. [27] and our scheme all support profile matching on ciphertexts. Although Qiu *et al*. [26] achieves flexible authorization, the authorization token is generated by two negotiated users, which may not applicable in MHSN. In our scheme and Ma *et al*. [27], the user can choose the data which to be matched according to their wishes by defining different trapdoors.

B. PERFORMANCE ANALYSIS

Let T_{exp}, T_{pair}, T_{hash}, N_a, N_u denote the computation cost of exponentiation operation in multiplicative groups, the computation cost of pairing operation, the computation cost of hash operation H_3, the number of attributes in access policy, the number of doctors, respectively. For simplicity, we ignore the symmetric encryption, general hash and multiplication operations.

Frist, we discuss the comparison during the data encryption, re-encryption and decryption phases. Table 2 shows the results. In the data encryption phase, Xu *et al*. [19] grows linearly with N_u at the slowest pace. However, it can only support simple keyword condition. Liang *et al*. [20] costs $(2N_a + 6)T_{exp}$ to encrypt data, which has almost the same cost with Yang *et al*. [21]. Unfortunately, the former scheme cannot support data conditional re-encryption and the latter one cannot support multiple receivers which is not practical in MHSN. In our scheme, the encryption computation cost is relevant to two factors, that are N_a and N_u, since the patient encrypts data with a set U and pre-defines an access policy to restrict which intended users can re-encrypt the ciphertext.

From Table2, we can learn that during the data re-encryption phase, the computation cost of our scheme grows slower than that in Liang *et al*. [20], but faster than that in Xu *et al*. [19] and Yang *et al*. [21], due to that our scheme made some sacrifices to support more essential functionalities. In the data decryption phase, it is clear that computation cost of our scheme and Xu *et al*. [19] are the same when decrypting the initial ciphertext encrypted by IBBE. Liang *et al*. [20] and Xu *et al*. [19] cost $(4N_a + 4)T_{pair} + 4N_aT_{exp}$ and $3T_{pair} + (N_u + 2)T_{exp}$ to decrypt the re-encrypted ciphertext, which both increase linearly with N_a or N_u respectively. Conversely, decryption cost of re-encrypted ciphertext keeps constant in our scheme, which is irrelevant to the above factors.

TABLE 2. Computation efficiency in secure data sharing.

Schemes	*Enc*	*ReEnc*	*Dec*-1	*Dec*-2
Liang et al. [20]	$(2N_a+6)T_{exp}$	$(2N_a+3)T_{pair}+(3N_a+5)T_{exp}$	$(2N_a+2)T_{pair}+2N_aT_{exp}$	$(4N_a+4)T_{pair}+4N_aT_{exp}$
Xu et al. [19]	$(N_u+6)T_{exp}$	$2T_{pair}+(N_u+2)T_{exp}$	$2T_{pair}+(N_u+2)T_{exp}$	$3T_{pair}+(N_u+2)T_{exp}$
Yang et al. [21]	$(2N_a+4)T_{exp}$	$(2N_a+1)T_{pair}+T_{exp}$	T_{exp}	$T_{pair}+T_{exp}$
Wu et al. [29]	$2T_{exp}$	-	$2T_{pair}$	-
Ma et al. [27]	$6T_{exp}$	-	$2T_{pair}+2T_{exp}$	-
Our scheme	$(2N_a+2N_u+7)T_{exp}$	$(2N_a+3)T_{pair}+(N_u+2)T_{exp}$	$2T_{pair}+(N_u+2)T_{exp}$	T_{pair}

Next, we compare our scheme with several profile matching schemes to evaluate the trapdoor generation and test algorithms. Table 3 indicates the comparison results. In terms of computation complexity of trapdoor generation, our scheme adopts *TrapGen*-1 algorithm to generate trapdoor from the secret key, and only cost one paring operation to generate trapdoor by using *TrapGen*-2 algorithm, which is less than other schemes. With regard to computation complexity of test algorithm, Wu *et al*. [29] and Ma *et al*. [27] cannot achieve flexible authentication, while our scheme and Ma *et al*. [30]

TABLE 3. Computation efficiency in profile matching.

Schemes	*TrapGen*-1	*TrapGen*-2	*Test*-1	*Test*-2	*Test*-3
Wu et al. [29]	0	-	$2T_{pair}+2T_{exp}+2T_{hash}$	-	-
Ma et al. [27]	0	-	$4T_{pair}+2T_{hash}$	-	-
Ma et al. [30]	0	$T_{exp}+T_{hash}$	$2T_{pair}+2T_{exp}+2T_{hash}$	$2T_{pair}+T_{hash}$	$2T_{pair}$
Our scheme	0	T_{pair}	$4T_{pair}+2T_{hash}$	$3T_{pair}+2T_{hash}$	$2T_{pair}+2T_{hash}$

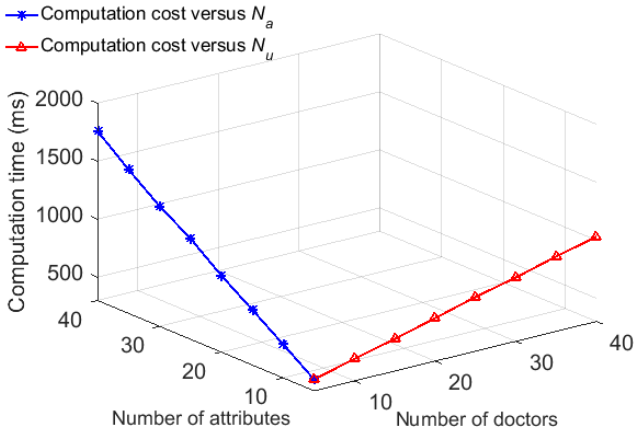

FIGURE 2. Computation cost of encryption.

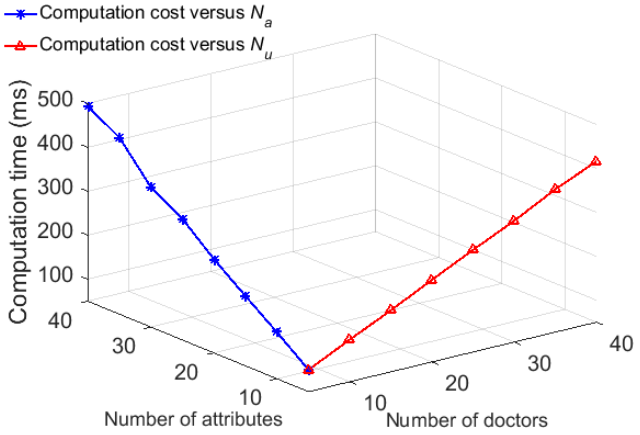

FIGURE 3. Computation cost of re-encryption.

which protects data confidentiality using PKE support several types of authorization, that are *Test*-1, *Test*-2 and *Test*-3 algorithm respectively. The computation cost of *Test*-1 in our scheme is slightly smaller than that in Ma *et al*. [30], while the*Test*-2 and *Test*-3 algorithm in our scheme have higher computational costs.

C. EXPERIMENTAL RESULTS

We implement the proposed system with java pairing-based cryptography library [34] to evaluate its performance. The experiments are conducted on a Windows platform with Intel Core CPU @ 2.70 GHz, 8 GB memory.

Since the encryption computation time is mainly related to N_a and N_u, we evaluate the impact of these two factors on the computation cost respectively by setting one of the factors as a fixed value. The results are shown in Fig. 2, and corroborate the fact that the *Enc* algorithm of our scheme performs linear computations with the N_a and N_u. The computation cost with 5 attributes and 20 doctors is about 670 ms, while the computation cost with 5 doctors and 20 attributes is about 960 ms, which is realistic and should suffice to meet the complex requirement of data access control in MHSN.

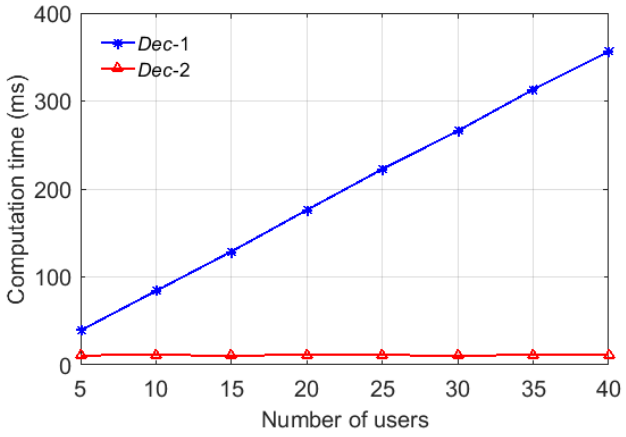

FIGURE 4. Computation cost of decryption.

TABLE 4. Computation time (ms).

Schemes	*TrapGen*-1	*TrapGen*-2	*Test*-1	*Test*-2	*Test*-3
Wu et al. [29]	0.0172	-	53.7969	-	-
Ma et al. [27]	0.0103	-	63.3425	-	-
Ma et al. [30]	0.0213	32.0129	75.7607	45.2521	12.6613
Our scheme	0.0166	5.4253	62.0125	57.1098	51.4711

The experimental result of re-encryption phase is depicted in Fig. 3, which shows the computational time of *ReEnc* varies linearly with N_a and N_u in the *CT*. Obviously, the computation time grows at a faster pace with the increasing of N_a than with the increasing of N_u. It is reasonable because more pairing operations will be required in the re-encryption phase, as N_a increases.

Fig. 4 reveals the computation time on the user side by decrypting the initial ciphertext and re-encrypted ciphertext. It is obvious that the computation time to decrypt the initial ciphertext is increasing with N_u, and the computation time to decrypt the re-encrypted ciphertext is almost constant, which takes about 11 ms.

A comparative summary of profile matching is presented in Table 4. It is clear from our evaluation that the computation cost in the *TrapGen*-1 of our scheme is close to the other three schemes, and the computation cost in the *Test*-1 of our scheme is little lower than Ma *et al*. [27], but slightly higher than Wu *et al*. [29], that is because that our scheme supports three types of authorization. The computation cost in the *TrapGen*-2 of our scheme is much lower than Ma *et al*. [30], since the heavy hash operation is performed by the CSP which has rich computing resources. Hence, the computation costs in the *Test*-2 and *Test*-3 of our scheme are higher than Ma *et al*. [30].

VIII. CONCLUSION

The MHSN has improved the healthcare through its convenient data sharing. For the purpose of guaranteeing

data confidentiality and availability in MHSN, we propose a secure identity-based data sharing and profile matching scheme in cloud computing. We first realize secure data sharing in MHSN with IBBE cryptographic technique, which allows the patients to store EHRs to cloud securely and share them with a group of doctors efficiently. Then we present an attribute-based CPRE mechanism in MHSN, which allows doctors who satisfy the pre-defined conditions to authorize the cloud to convert a stored ciphertext into a new ciphertext under IBE for the specialist, without leaking any sensitive information. Further, we provide a profile matching mechanism based on IBEET, which can achieve flexible authorization on encrypted EHRs and help patients to find friends in a privacy-preserving and efficient way. The analysis and results show that the computation cost on patient side is reduced.

REFERENCES

[1] L. Guo, C. Zhang, J. Sun, and Y. Fang, "PAAS: A privacy-preserving attribute-based authentication system for eHealth networks," in *Proc. 32nd Int. Conf. Distrib. Comput. Syst.*, Macau, China, Jun. 2012, pp. 224–233.

[2] A. Abbas and S. U. Khan, "A review on the state-of-the-art privacy-preserving approaches in the e-health clouds," *IEEE J. Biomed. Health Informat.*, vol. 18, no. 4, pp. 1431–1441, Apr. 2014.

[3] C. Delerablée, "Identity-based broadcast encryption with constant size ciphertexts and private keys," in *Proc. 13th Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, Kuching, Malaysia, 2007, pp. 200–215.

[4] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy*, Berkeley, CA, USA, May 2007, pp. 321–334.

[5] M. Green and G. Ateniese, "Identity-based proxy re-encryption," in *Proc. 5th Int. Conf. Appl. Cryptogr. Netw. Secur.*, Zhuhai, China, 2007, pp. 288–306.

[6] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 1, pp. 131–143, Jan. 2013.

[7] M. Barua, X. Liang, R. Lu, and X. Shen, "ESPAC: Enabling security and patient-centric access control for eHealth in cloud computing," *Int. J. Secur. Netw.*, vol. 6, nos. 2–3, pp. 67–76, 2011.

[8] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *Proc. IEEE INFOCOM*, San Diego, CA, USA, Mar. 2010, pp. 1–9.

[9] Y. Liu, Y. Zhang, J. Ling, and Z. Liu, "Secure and fine-grained access control on e-healthcare records in mobile cloud computing," *Future Gener. Comput. Syst.*, vol. 78, pp. 1020–1026, Jan. 2017.

[10] Y. Yang, X. Liu, R. H. Deng, and Y. Li, "Lightweight sharable and traceable secure mobile health system," *IEEE Trans. Dependable Secure Comput.*, to be published, doi: 10.1109/TDSC.2017.2729556.

[11] Y. Yang, X. Liu, and R. H. Deng, "Lightweight break-glass access control system for healthcare Internet-of-Things," *IEEE Trans. Ind. Inform.*, to be published, doi: 10. 1109/TII.2017.2751640.

[12] G.-C. Li, C.-L. Chen, H. C. Chen, F. Lin, and C. Gu, "Design of a secure and effective medical cyber-physical system for ubiquitous telemonitoring pregnancy," *Concurrency Comput. Pract. Exper.*, vol. 30, no. 2, p. e4236, 2018.

[13] C. C. Tan, H. Wang, S. Zhong, and Q. Li, "IBE-lite: A lightweight identity-based cryptography for body sensor networks," *Trans. Inf. Technol. Biomed.*, vol. 13, no. 6, pp. 926–932, Jun. 2009, doi: 10.1109/titb.2033055.

[14] X. An Wang, J. Ma, F. Xhafa, M. Zhang, and X. Luo, "Cost-effective secure E-health cloud system using identity based cryptographic techniques," *Future Gener. Comput. Syst.*, vol. 67, pp. 242–254, Feb. 2017.

[15] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.*, Espoo, Finland, 1998, pp. 127–144.

[16] T. Matsuo, "Proxy re-encryption systems for identity-based encryption," in *Proc. Int. Conf. Pairing-Based Cryptogr.*, Tokyo, Japan, 2007, pp. 247–267.

[17] Y. Zhou, H. Deng, Q. Wu, B. Qin, J. Liu, and Y. Ding, "Identity-based proxy re-encryption version 2: Making mobile access easy in cloud," *Future Gener. Comput. Syst.*, vol. 62, pp. 128–139, Sep. 2016.

[18] J. Weng, R. Deng, X. Ding, C.-K. Chu, and J. Lai, "Conditional proxy re-encryption secure against chosen-ciphertext attack," in *Proc. 4th Int. Symp. Inf., Comput., Commun. Secur.*, Sydney, NSW, Australia, 2009, pp. 322–332.

[19] P. Xu, T. Jiao, Q. Wu, W. Wang, and H. Jin, "Conditional identity-based broadcast proxy re-encryption and its application to cloud email," *IEEE Trans. Comput.*, vol. 65, no. 1, pp. 66–79, Jan. 2016.

[20] K. Liang *et al.*, "A secure and efficient ciphertext-policy attribute-based proxy re-encryption for cloud data sharing," *Future Generat. Comput. Syst.*, vol. 52, pp. 95–108, Nov. 2015.

[21] Y. Yang, H. Zhu, H. Lu, J. Weng, Y. Zhang, and K.-K. R. Choo, "Cloud based data sharing with fine-grained proxy re-encryption," *Pervasive Mobile Comput.*, vol. 28, pp. 122–134, Jun. 2016.

[22] Q. Huang, Y. Yang, and J. Fu, "PRECISE: Identity-based private data sharing with conditional proxy re-encryption in online social networks," *Future Gener. Comput. Syst.*, vol. 86, pp. 1523–1533, Sep. 2018.

[23] M. Li, N. Cao, S. Yu, and W. Lou, "FindU: Privacy-preserving personal profile matching in mobile social networks," in *Proc. IEEE INFOCOM*, Shanghai, China, Apr. 2011, pp. 2435–2443.

[24] R. Zhang, J. Zhang, Y. Zhang, J. Sun, and G. Yan, "Privacy-preserving profile matching for proximity-based mobile social networking," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 656–668, Sep. 2013.

[25] L. Zhang, X.-Y. Li, K. Liu, T. Jung, and Y. Liu, "Message in a sealed bottle: Privacy preserving friending in mobile social networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 9, pp. 1888–1902, Sep. 2015.

[26] S. Qiu, J. Liu, Y. Shi, M. Li, and W. Wang, "Identity-based private matching over outsourced encrypted datasets," *IEEE Trans. Cloud Comput.*, to be published, doi: 10.1109/TCC.20 15.2511723.

[27] S. Ma, "Identity-based encryption with outsourced equality test in cloud computing," *Inf. Sci.*, vol. 328, pp. 389–402, Jan. 2016.

[28] L. Wu, Y. Zhang, K.-K. R. Choo, and D. He, "Efficient identity-based encryption scheme with equality test in smart city," *IEEE Trans. Sustain. Comput.*, vol. 3, no. 1, pp. 44–55, Jan./Mar. 2018.

[29] L. Wu, Y. Zhang, K.-K. R. Choo, and D. He, "Efficient and secure identity-based encryption scheme with equality test in cloud computing," *Future Generat. Comput. Syst.*, vol. 73, pp. 22–31, Aug. 2017.

[30] S. Ma, Q. Huang, M. Zhang, and B. Yang, "Efficient public key encryption with equality test supporting flexible authorization," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 3, pp. 458–470, Mar. 2015.

[31] Q. Huang, Y. Yang, and Y. Shi, "SmartVeh: Secure and efficient message access control and authentication for vehicular cloud computing," *Sensors*, vol. 18, no. 2, p. E666, 2018.

[32] Q. Huang, Y. Yang, and J. Fu, "Secure data group sharing and dissemination with attribute and time conditions in public cloud," *IEEE Trans. Services Comput.*, to be published, doi: 10.1109/TSC.2018.2850344.

[33] G. Yang, C. H. Tan, Q. Huang, and D. S. Wong, "Probabilistic public key encryption with equality test," in *Proc. Cryptographers' Track RSA Conf.*, San Francisco, CA, USA, 2010, pp. 119–131.

[34] *The Java Pairing-Based Cryptography Library*. Accessed: 2018. [Online]. Available: http://gas.dia.unisa.it/projects/jpbc/

QINLONG HUANG (M'17) received the Ph.D. degree from the School of Computer, Beijing University of Posts and Telecommunications, China, in 2014. He is currently a Lecturer with the School of Cyberspace Security, Beijing University of Posts and Telecommunications, and the Associate Director of the National Engineering Laboratory for Disaster Backup and Recovery, China. He is also a Principal Investigator of the project funded by the National Natural Science Foundation of China. His research interests include cloud computing security, social network security, and IoT security. He was serving as a Reviewer for the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, *ACM Computing Surveys*, the *ACM Transactions on Multimedia Computing, Communications, and Applications*, the IEEE ACCESS, and *IET Information Security*.

WEI YUE is currently pursuing the master's degree with the School of Cyberspace Security, Beijing University of Posts and Telecommunications. Her research interests include cloud computing security.

YUE HE is currently pursuing the master's degree with the School of Cyberspace Security, Beijing University of Posts and Telecommunications. His research interests include cloud computing security.

YIXIAN YANG received the Ph.D. degree from the Beijing University of Posts and Telecommunications, China, in 1988. He was a Changjiang Distinguished Professor of China in 1993, and was selected in the National Science Fund for Distinguished Young Scholars of China in 1994. He is currently a Professor with the School of Cyberspace Security, Beijing University of Posts and Telecommunications. He is also the Director of the National Engineering Laboratory for Disaster Backup and Recovery, China. He has published over 300 journals and conference papers. His research interests include cryptography, information, and network security. He is a fellow at the China Institute of Communications and the Chinese Association for Cryptologic Research. He was the Editor-in-Chief of the *Journal on Communications*.