IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 26, 2018, accepted May 11, 2018, date of publication May 22, 2018, date of current version June 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2838095

Anonymous Data Sharing Scheme in Public Cloud and Its Application in E-Health Record

HUAQUN WANG

Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

e-mail: wanghuaqun@aliyun.com

This work was supported in part by the Qing Lan Project of Jiangsu Province, in part by the 1311 Talent Plan Foundation of NUPT, NUPTSF, under Grant NY216001, and in part by the Open Foundation of the State Key Laboratory of Information Security of China under Grant 2017-MS-15.

ABSTRACT In the past few years, cloud computing develops very quickly. A large amount of data are uploaded and stored in remote public cloud servers which cannot fully be trusted by users. Especially, more and more enterprises would like to manage their data by the aid of the cloud servers. However, when the data outsourced in the cloud are sensitive, the challenges of security and privacy becomes urgent for wide deployment of the cloud systems. This paper proposes a secure data sharing scheme to ensure the privacy of data owner and the security of the outsourced cloud data. The proposed scheme provides flexible utility of data while solving the privacy and security challenges for data sharing. The security and efficiency analysis demonstrate that the designed scheme is feasible and efficient. At last, we discuss its application in electronic health record.

INDEX TERMS Attribute-based encryption, cloud computing, data sharing, searchable encryption.

I. INTRODUCTION

With the fast development and application of cloud computing, more and more users are moving their data to cloud servers. The technique of cloud computing relieves the consumes of data management, data processing, and capital expenditure on hardware, software, and personnel maintenances, *etc*. Although the advantages of cloud computing, some barriers affect and make the enterprises reluctant to migrate the data to the cloud server. Public cloud is owned and controlled by public cloud servers (PCS), which cannot be trusted. PCS might steal or get the data information stored by the users. Thus, many different security notions are proposed to ensure the security in cloud such as remote data integrity, remote data sharing, *etc*.

Data sharing is one of important applications in cloud computing, especially for enterprise. Usually, an enterprise may authorize some entities to share its remote data under the its defined policy. However, the data have to satisfy the following security in most applications: 1) the privacy information of the data should be preserved, 2) non-authorized entities are unable to get the information of the outsourced data and share their remote data with other users. Thus, how to design a data sharing scheme while achieving privacy-preserving and data confidentiality in public cloud is an urgent challenge. For example, it is common that a user has his own medical/health data which includes electronic medical records, biomedical image, audio or video media, *etc*. These medical/health data needs strict security protection since it involves the patients' privacy. In order to further study medicine and improve the level of medical care, medical researchers need to share the patients' data and mine the valuable information. In order to find the general data rule, these medical researchers will deal with huge number of patients' data which targets at particular individuals. Since the medical/health data is privacy, the patients' identity information must be protected while their data are shared. At the same time, the medical/health data only can be shared by the authorized entities. The non-authorized entities cannot get any information of the medical/health data, *i.e.*, data confidentiality must be ensured.

A. RELATED WORK

When more and more data are uploaded and stored in public clouds, some new data management issues are proposed. Data sharing is an indispensable service from the cloud computing. In order to share data with others in cloud storage, Chu *et al.* [1] described new public-key cryptosystems. The new systems can generate constant-size ciphertexts which can realize the delegation of decryption rights for any set of ciphertexts [1]. By using the private cloud, Tong *et al.* [2] studies the privacy problem of mobile healthcare systems. Pervez *et al.* [3] proposed self-healing attribute-based privacy aware data sharing in cloud. In order to realize dynamic membership management with arbitrary states, Fan *et al.* [4] presented an attribute-based encryption scheme. Boneh *et al.* [5] defined and constructed public key encryption with keyword search. Cao *et al.* [6] proposed a basic idea for the multi-keyword ranked search over encrypted cloud data, then they give two significantly improved multi-keyword ranked search schemes which satisfy many kinds of stringent privacy requirements. Seo *et al.* proposed a mediated certificateless encryption scheme without pairing operations. They applied their mediated certificateless encryption scheme to construct an efficient sharing sensitive information scheme in public clouds [7]. Some other works [11]–[17] drew more attention on adding functionalities of sharing, such as authentication and matching.

Along with the rapid increasing of medical/health data, more and more hospitals upload their data to public clouds and delegate the public cloud providers to manage their data. Medical/health data security has attracted many researchers. Until now, many research results have appeared. Li *et al.* [19] proposed a novel patient-centric framework and a suite of mechanisms for data access control to personal health records stored in semi-trusted servers. Benaloh *et al.* [20] build an efficient system that allows patients both to share partial access rights with others, and to perform searches over their records. Sun *et al.* propose a secure electronic health record system, based on cryptographic constructions, to enable secure sharing of sensitive patient data during cooperation and preserve patient data privacy. Their system further incorporates advanced mechanisms for fine-grained access control, and on-demand revocation, as enhancements to the basic access control offered by the delegation mechanism, and the basic revocation mechanism, respectively [21]. Bahga and Madisetti [22] described the high-level design of cloud health information systems technology scheme and the approaches for semantic interoperability, data integration, and security. In 2014, Anthony *et al.* [23] studied the access control and security audit for medical/health data in order to data security. Canim *et al.* introduced a framework that removes the need for multiple third parties by collocating services to store and to process sensitive biomedical data through the integration of cryptographic hardware. They define a secure protocol to process genomic data and perform a series of experiments [24]. Lest [25] studied electronic records secrecy, anonymity and privacy-preservation. Hass *et al.* [26] proposed the electronic health system which can protect the patients' privacy . Based on the group signature, Zhang and Liu [27] proposed anonymous digital certification which can be used to electronic health network in cloud computing. In 2013, Fernandez-Aleman *et al.* [28] gave the systematic literature review on security and privacy in electronic health records. Ahmed *et al.* argued that the eHealth Exchange needs to be augmented to provide greater patient awareness and control. They take an approach that informs the patient when her health data is accessed by a healthcare enterprise that is not already trusted by the patient. Such awareness is ensured even when some systems in the health information sharing environment become compromised [29]. Aiming at allowing for efficient storing and sharing personal health records and also eliminating patients' worries about personal health records privacy, Xhafa *et al.* [30] designed a secure cloud-based electronic health record system, which guaranteed security and privacy of medical data stored in the cloud, relying on cryptographic primitive but not the full trust over cloud servers. Wang *et al.* [31] designed and developed a patient-centric, cloud-based personal health record system based on open-source Indivo project. In addition, there were some literatures [9], [10], [32]–[36] that presented solutions for problems in cloud services, such as data identity management, secure data analysis including privacy machine learning and classification and deduplication etc.

B. CONTRIBUTIONS

This paper focuses on data sharing scheme which achieves anonymity and data confidentiality. With the outsourced data, it is difficult to design an efficient way to share the data while keeping the data owners identity privacy. In order to solve the above problem, we investigate an anonymous data sharing scheme. Our contribution is two-fold:

- First, we give the formal model of data sharing achieving anonymity and data confidentiality in public clouds. Through analyzing the real system and security requirements, we gave the formal system model and security model.
- Second, we realize the data sharing scheme which can achieve privacy-preservation and data confidentiality in public clouds. By using symmetric encryption, searchable encryption and attribute-based encryption techniques, we design an efficient scheme which satisfies the security properties.

C. PAPER ORGANIZATION

The rest of the paper is organized below. Section II formalizes the data sharing system model and security model. Section III presents our data sharing scheme which satisfies the security properties. Section IV evaluates the security and performance of our scheme. Section V discusses the proposed scheme's application in E-health record. Finally, Section VI concludes the paper.

II. MODELING DATA SHARING SCHEME

The data sharing scheme system model and its security model are given in this section. The data sharing scheme comprises of three different entities, Cloud Server, Data owner and Data sharer, as illustrated in Figure 1. They can be identified as

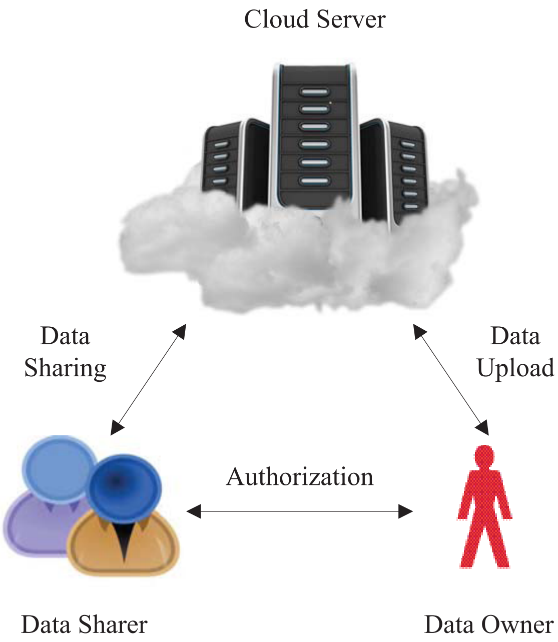

FIGURE 1. The system model of data sharing.

follows:

1) *Data Owner*: Data owner is an entity whose massive data will be uploaded to the cloud servers for storage and processing. It is either the patients or the hospital.
2) *Data Sharer*: Data sharer is an entity who will share the data owners' remote data. It maybe the medical/health researcher, the medical/health research organization or the relatives of the data owner.
3) *Cloud Server* : Cloud server is an entity who is managed by cloud service provider. It has enormous storage space and computation resource which are used to process the data owners' data.

The technique of cloud computing relieves the consumes of data management, data processing, and capital expenditure on hardware, software, and personnel maintenances, etc. Since the data owners no longer possess their data locally, it is important to ensure their remote data is integer. When the data owners authorize some entities to share their remote data, it is important to efficiently share the authorized remote data by the data sharers.

In the data sharing protocol, the access structure [37]–[39] and access tree [40], [41] are necessary. We give their definition or description below:

Definition 1 (Access Structure): Denote a set of parties as $\{P_1, P_2, \cdots, P_n\}$. We define a collection $\mathbb{A} \subseteq 2^{\{P_1,P_2,\cdots,P_n\}}$ which consists of all the subsets of $\{P_1, P_2, \cdots, P_n\}$. If the following condition is satisfied, i.e., $\forall B, C$: if $B \in \mathbb{A}$ and $B \subseteq C$ then $C \in \mathbb{A}$, we call $\mathbb{A}$ is monotone. When $\mathbb{A} \subseteq 2^{\{P_1,P_2,\cdots,P_n\}} \backslash \{\Phi\}$ and $\mathbb{A}$ is monotone, $\mathbb{A}$ is called the monotone access structure, where Φ is the empty set. If the set $D \in \mathbb{A}$, we called D is the authorized set; otherwise, it is called the unauthorized set.

Access tree $\mathbb{T}$. We denote an access structure as a tree $\mathbb{T}$. In the tree $\mathbb{T}$, every non-leaf node represents a threshold gate which is described by its children and a threshold value. Denote the number of children of a node x as num_x and the threshold value as k_x which satisfy $0 < k_x \leq num_x$. When $k_x = 1$, the threshold gate is an OR gate and when $k_x = num_x$, it is an AND gate. Each leaf node x of the tree is described by an attribute and a threshold value $k_x = 1$. Denote the parent of the node x as $parent(x)$. If x is a leaf node, we denote the attribute associated with the leaf node x as the function $att(x)$. In $\mathbb{T}$, the children of a node are numbered from 1 to num. The function $index(x)$ returns such a number associated with the node x.

Satisfying an access tree. If the access tree $\mathbb{T}$ has the root r, it is denoted as $\mathbb{T}_r$. Let $\mathbb{T}_x$ denote the subtree of $\mathbb{T}$ with the root x. When the set of attributes γ satisfies $\mathbb{T}_x$, it is denoted as $\mathbb{T}_x(\gamma) = 1$. The value $\mathbb{T}_x(\gamma)$ can be calculated recursively below. If x is a non-leaf node, evaluate $\mathbb{T}_{x'}(\gamma)$ for all children x' of node x. $\mathbb{T}_x(\gamma)$ returns 1 if and only if at least k_x children return 1. If x is a leaf node, then $\mathbb{T}_x(\gamma)$ returns 1 if and only if $att(x) \in \gamma$.

Next, we model a data sharing scheme which can achieve anonymity and data confidentiality. Then, we present the formal security definitions according to the security requirements.

Definition 2 (Data Sharing Scheme): A data sharing scheme comprises seven polynomial time algorithms: **Setup**, **Sym-Enc**, **AB-Enc**, **S-Enc**, **GenList**, **GenRetr**, and **Retr**. These seven algorithms are detailed below:

1) $(params, mpk, msk) \leftarrow$ **Setup**(1^k) is the parameter generation algorithm. Let k denote the security parameter. Upon receiving k, the algorithm can output the system public parameters $params$. At the same time, it also outputs the master public/secret key pair (mpk, msk).
2) $E_1 \leftarrow$ **Sym-Enc(F)** is a symmetric encryption algorithm that is run by the data owner. Firstly, the data owner classifies these data F into n categories $(F_1, \cdots, F_n)$ based on the data properties. Secondly, the data owner selects the different symmetric encryption keys $(k_1, k_2, \cdots, k_n)$ to encrypt the different classified data $(F_1, \cdots, F_n)$.
3) $E_2 \leftarrow$ **AB-Enc**$(k_i, 1 \leq i \leq n)$ is a probabilistic polynomial time algorithm. By using the algorithm, the data owner encrypts $\{k_i, 1 \leq i \leq n\}$.
4) $E_3 \leftarrow$ **S-Enc**$(F_i$'s keywords) is a probabilistic polynomial time algorithm that is run by the data owner to encrypt the keywords of the data F_i by using the searchable encryption algorithm. Based on the classified data $(F_1, \cdots, F_n)$, the data owner selects the keywords of the different classified data. Then, the data owner encrypts the keywords.
5) $L \leftarrow$ **GenList**$(F_{name}, F_{key}, Owner_{name}, Owner_{alias})$ is an algorithm that is run by the data owner. The data

owner's name is denoted as $Owner_{name}$ and the data owner's alias is denoted as $Owner_{alias}$. The stored file's name and keywords are denoted as F_{name} and F_{key}, respectively. At last, they are organized into the list L.

6) $V \leftarrow \mathsf{GenRetr}(params, E_1, E_2, E_3, L, keywords)$ is run by the PCS in order to share the remote data. It takes as inputs the public parameter $params$, the ciphers E_1, E_2, E_3, the list L and the queried keywords $keywords$. It returns V as the response.
7) $\{\hat{F}\} \leftarrow \mathsf{Retr}(mpk, sk_{ID}, V)$ is run by the data sharer in order to share the remote data. It takes as inputs mpk, the sharer's secret key sk_{ID}, and the PCS's response V. The sharer can retrieve the data $\hat{F}$.

To guarantee the proposed scheme's security, a remote data sharing scheme should satisfy the security requirements below:

1) The unauthorized entity cannot retrieve the remote data.
2) The public key searchable encryption algorithm is semantically secure against an adaptive chosen keyword attack.
3) The sharer cannot get the data owner's real name.

To capture the above security requirements, we define the security of a data sharing scheme as follows.

In order to define the security of the phase $\mathsf{AB-Enc}$, we give the game below. The game is run between the challenger $\mathcal{C}$ and the adversary $\mathcal{A}$.

1) Setup: $\mathcal{C}$ generates the public parameters A_{pub} and the secret parameters A_{priv}. $\mathcal{C}$ sends A_{pub} to $\mathcal{A}$ and keeps A_{priv} secret for $\mathcal{A}$.
2) Phase 1: For the different attribute sets $s_1, s_2, \cdots, s_{q_1}$, $\mathcal{A}$ adaptively queries their secret keys to $\mathcal{C}$.
3) Challenge: $\mathcal{A}$ submits two messages M_0 and M_1 who have the same length, *i.e.*, $|M_0| = |M_1|$. On the other hand, $\mathcal{A}$ also gives a challenge access structure $\mathbb{A}^*$ such that none of the sets $s_1, \cdots, s_{q_1}$ satisfies the access structure $\mathbb{A}^*$. $\mathcal{C}$ flips a random coin b and encrypts M_b under the access structure $\mathbb{A}^*$. $\mathcal{C}$ sends the ciphertext CT^* to $\mathcal{A}$.
4) Phase 2: Phase 1 is replayed with the restriction that none of set of attributes $s_{q_1+1}, \cdots, s_{q_2}$ satisfies the access structure $\mathbb{A}^*$.
5) Guess: The adversary outputs a guess b' of b.

In this game, $\mathcal{A}$'s advantage is defined as

$$Adv_{\mathcal{A}} = \left|\Pr[b = b'] - \frac{1}{2}\right|.$$

Definition 3 (AB-Enc Security): A ciphertext-policy attribute-based encryption scheme is secure if all probabilistic polynomial time adversaries have at most a negligible advantage in the above game.

In order to define the security of the phase $\mathsf{S-Enc}$, we give the game below. In the game, we denote the challenger as $\mathcal{C}$ and the adversary $\mathcal{A}$.

1) $\mathcal{C}$ generates the public parameters A_{pub} and the secret parameters A_{priv}. $\mathcal{C}$ sends A_{pub} to $\mathcal{A}$ and keeps A_{priv} secret for $\mathcal{A}$.
2) Phase 1: For the different keywords $W_1, W_2, \cdots, W_{q_1}$, $\mathcal{A}$ adaptively asks $\mathcal{C}$ for the trapdoor T_{W_i} which corresponds to the keyword W_i where $i = 1, 2, \cdots, q_1$.
3) $\mathcal{A}$ picks two challenged keywords $\hat{W}_0, \hat{W}_1$ and sends them to $\mathcal{C}$ where $\hat{W}_i \notin \{W_1, W_2, \cdots, W_{q_1}\}, i = 0, 1$. $\mathcal{C}$ picks a random $b \in \{0, 1\}$ and sends $\mathsf{S\text{-}Enc}(A_{pub}, \hat{W}_b)$ to $\mathcal{A}$.
4) Phase 2: Phase 1 is replayed with the restriction $\hat{W}_i \notin \{W_{q_1+1}, \cdots, W_{q_2}\}$ and $i = 0, 1$.
5) Finally, $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$ and wins the game if $b = b'$.

In the process of breaking $\mathsf{S\text{-}Enc}$, $\mathcal{A}$'s advantage is defined below

$$Adv_{\mathcal{A}} = \left|\Pr[b = b'] - \frac{1}{2}\right|$$

Definition 4 (S-Enc Security): For any probabilistic polynomial time adversary $\mathcal{A}$, if $Adv_{\mathcal{A}}$ is a negligible function, we call the phase $\mathsf{S\text{-}Enc}$ is semantically secure.

Definition 5 (Anonymity): For the data sharer, it is difficult to identify the data owner.

In this paper, the privacy denotes the data owner identification. Privacy-preserving denotes to realize the data owner anonymity.

III. OUR PROPOSED DATA SHARING SCHEME

This section gives an efficient data sharing scheme which satisfies data owner anonymity and data confidentiality. Our proposed scheme is built from bilinear pairings. Bilinear pairings come from the Weil pairings or Tate pairings of the elliptic curve on the finite field. We also briefly review them below.

A. BILINEAR PAIRINGS

In order to simplify the expressions, we denote $\mathcal{G}_1$ and $\mathcal{G}_2$ as two cyclic multiplicative groups. $\mathcal{G}_1$ and $\mathcal{G}_2$'s orders are the same prime order q. We define the bilinear map as $e : \mathcal{G}_1 \times \mathcal{G}_1 \rightarrow \mathcal{G}_2$ [44]. By using the modified Weil [8] or Tate pairings [42] on elliptic curves, we can construct such a bilinear map e. Our scheme is constructed on the gap Diffie-Hellman group, *i.e.*, the computational Diffie-Hellman (CDH) problem is hard while the decisional Diffie-Hellman (DDH) problem is easy [43]. CDH and DDH problems are given below.

Definition 6 (Gap Diffie-Hellman (GDH) Group): Let g be a generator of $\in \mathcal{G}_1$. Given $g, g^a, g^b, g^c \in \mathcal{G}_1$ where $a, b, c \in \mathcal{Z}_q^*$ are unknown, there exists an efficient algorithm to determine whether $ab = c \bmod q$ holds by verifying $e(g^a, g^b) = e(g, g)^c$ in polynomial time (DDH problem), while there does no exist efficient algorithm to compute $g^{ab} \in \mathcal{G}_1$ with non-negligible probability within polynomial time (CDH problem). A group $\mathcal{G}_1$ is a (t, ϵ)-GDH group if DDH problem can be efficiently solved while no algorithm (t, ϵ)-breaks CDH problem.

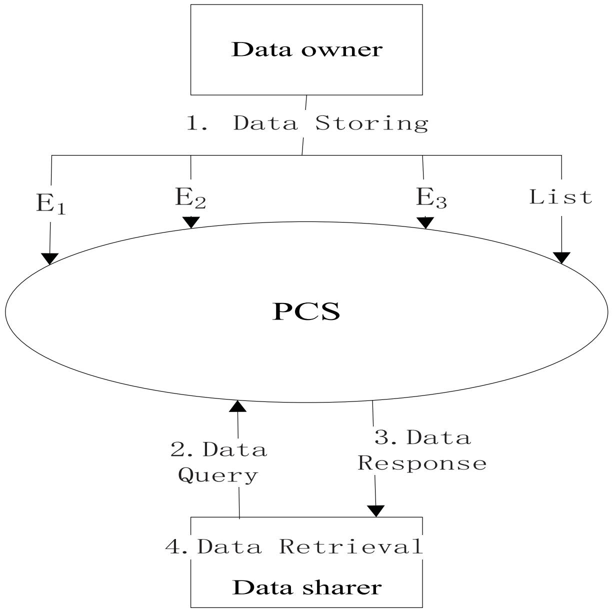

FIGURE 2. Data sharing scheme architecture.

Definition 7 (Bilinear Diffie-Hellman Assumption (BDH)): Given (g, g^a, g^b, g^c) for unknown $a, b, c \in \mathcal{Z}_q^*$, it is difficult to compute $W = e(g, g)^{abc} \in \mathcal{G}_2$.

B. DATA SHARING SCHEME CONSTRUCTION

This scheme comprises seven procedures, *i.e.*, Setup, Sym-Enc, AB-Enc, S-Enc, GenList, GenRetr and Retr. They can be given as Figure 2. We express the figure below: 1. Data owner classifies its data and gets the keywords for the different type data. The ciphertext and list $(E_1, E_2, E_3, List)$ are uploaded to PCS. 2. The data sharers send data sharing query to PCS. 3. PCS sends the queried data to the data sharers. 4. Data sharers decrypt the received data and get the plaintext.

Suppose that the data owner O_i will upload the data F_i to PCS. Firstly, O_i picks its alias Al_i. In the data F_i, O_i's real name will be replaced by the alias Al_i. Secondly, based on the data property, O_i classifies F_i into $\{F_{i1}, F_{i2}, \cdots, F_{in_i}\}$. For the classified data F_{ij}, O_i extracts its keywords W_{ij}. At the same time, we define two cryptographic hash functions below:

$$H_1 : \{0, 1\}^* \rightarrow \mathcal{G}_1, H_2 : \mathcal{G}_2 \rightarrow \{0, 1\}^{\lceil \log_2 q \rceil}$$

- Setup: O_i picks a secure symmetric encryption algorithm E. It also picks two random numbers $\alpha, \beta \in \mathcal{Z}_q^*$ and computes $g_1 = g^\beta, g_2 = e(g, g)^\alpha$. The public key is $pk = (\mathcal{G}_1, g, g_1, g_2)$. The master secret key is $mk = (\beta, g^\alpha)$.
- Sym-Enc: O_i picks $k_{ij} \in \mathcal{Z}_q$ for $1 \leq j \leq n_i$. For every F_{ij}, by using symmetric encryption algorithm E, O_i computes $C_{ij} = E_{k_{ij}}(F_{ij})$ and gets the ciphertext $SC_i = (C_{ij})_{1 \leq j \leq n_i}$. At the same time, for every F_{ij}, it extracts the keyword W_{ij}, $1 \leq j \leq n_i$.
- AB-Enc: O_i performs the procedures below:

TABLE 1. List of the file, keywords and owner.

F_{name}	T_{key}	$Owner_{name}$	$Owner_{alias}$
FN_{i1}	T_{i1}	O_i	Al_i
$\cdots$	$\cdots$	$\cdots$	$\cdots$
FN_{in_i}	T_{in_i}	O_i	Al_i

1) **AB-KeyGen**: For a user who has the set of attributes $\mathbb{A}$, O_i first randomly picks $r \in \mathcal{Z}_q$, and $r_j \in \mathcal{Z}_q$ for every attribute $a_j \in \mathbb{A}$. Then, O_i gives the user $sk = (D = g^{\frac{\alpha+r}{\beta}}, \{D_j = g^r H_1(a_j)^{r_j}, D'_j = g^{r_j}\}_{\forall a_j \in \mathbb{A}})$, where an attribute a_j is represented as a string.
2) **AB-Encrypt**: Denote the access tree as $\mathbb{T}$. For each node N_j in tree $\mathbb{T}$, O_i selects a polynomial f_j whose degree is $d_j = n_j - 1$, where n_j is a threshold such as node N_j is TRUE if it has n_j child nodes whose Boolean values are TRUE. For the root node N_1, selects a random $s \in \mathcal{Z}_q$ and let $f_1(0) = s$. For any non-root node N_j in the tree, choose a polynomial $f_j(\cdot)$ which satisfies $f_j(0) = f_{parent(N_j)}(j)$. The ciphertext AC_i is computed below:

$$B_{ij} = k_{ij} g_2^s = k_{ij} e(g, g)^{\alpha s}, \quad 1 \leq j \leq n_i$$
$$\bar{C} = g_1^s = g^{s\beta}$$
$$\{E_l = g^{f_l(0)}, \ E'_l = H_1(a_i)^{f_l(0)}\}_{l \in \mathcal{L}}$$

where attribute $a_i \in \mathcal{S}$ is associated with a leaf node $N_j \in \mathcal{L}$, $\mathcal{S}$ is the set of attributes, and $\mathcal{L}$ is the set of leaf nodes.

- S-Enc: For the keyword W_{ij}, O_i computes its trapdoor $T_{ij} = H_1(W_{ij})^\beta$. Denote $T_i = \{T_{ij}, 1 \leq j \leq n_i\}$.
- GenList: O_i creates the following list $List_i$ below:
  1) The first column gives the different classified file name;
  2) The second column gives the file keyword trapdoor;
  3) The third column gives the file owner's real name;
  4) The fourth column gives the file owner's alias.

  When the above procedures are finished, O_i uploads $(SC_i, AC_i, List_i)$ to PCS. Upon receiving $(SC_i, AC_i, List_i)$, PCS combines all the $list_i$ into one list $list$. At the same time, PCS stores (SC_i, AC_i) by itself.
- GenRetr: Upon receiving the challenge (A, B) which contains the queried file keywords, PCS performs the following procedures:
  1) In the list $list$ of file, keywords and owner, for $1 \leq i \leq n, 1 \leq j \leq n_i$, PCS checks whether $H_2(e(A, T_{ij})) = B$ holds. In the phase, PCS gets the keyword trapdoor set $(T_{ij}, (i, j) \in (I, J))$, *i.e.*, (I, J) denotes all the subscripts of the valid keyword trapdoor.
  2) Based on the searched keywords trapdoor $(T_{ij}, (i, j) \in (I, J))$ and the list $list$, PCS determines the corresponding file information $(FN_{ij}, (i, j) \in (I, J))$.

3) PCS sends the following data V to the data sharer

$$V = (C_{ij}, B_{ij}, \bar{C}, \{E_l,\ E'_l\}_{l\in\mathcal{L}}, (i,j) \in (I, J)))$$

- **Retr**: Let the retrieved file keyword be W. The data sharer picks a random $r \in \mathcal{Z}_q$ and computes $H_2((e(H_1(W), g_1^r))$. Denote $A = g^r, B = H_2((e(H_1(W), g_1^r))$. Then, the data sharer sends (A, B) to PCS as the query. Upon receiving the response V, the following procedures are performed by the data sharer who has a set $\mathcal{S}$ of attributes $\mathbb{A}$. Specifically, for any node N_j in $\mathbb{T}$:

1) If N_j is a leaf node which associates with attribute $a_i \in \mathbb{A}\bigcap\mathcal{S}$, let the Boolean value of node N_j be TRUE, and compute

$$\begin{aligned} V_j &= \frac{e(D_i, E_j)}{e(D'_i, E'_j)} \\ &= \frac{e(g^r H_1(a_i)^{r_i}, g^{f_j(0)})}{e(g^{r_i}, H_1(a_i)^{f_j(0)})} \\ &= e(g,g)^{rf_j(0)} \end{aligned}$$

2) If N_j is a non-leaf node, let $\mathcal{S}_j$ be its arbitrary n_j-sized set of child nodes N_k whose $V_k \neq\perp$. If the satisfied set does not exist, set $V_j =\perp$; otherwise, we set N_j's Boolean function value as TRUE, and compute

$$\begin{aligned} V_j &= \prod_{k\in\mathcal{S}_j} V_k^{\Delta_{k,\mathcal{S}_j}(0)} \\ &= \prod_{k\in\mathcal{S}_j} e(g,g)^{rf_k(0)\Delta_{k,\mathcal{S}_j}(0)} \\ &= e(g,g)^{rf_j(0)} \end{aligned}$$

where

$$\Delta_{k,\mathcal{S}_j}(x) = \prod_{u\in\mathcal{S}_j, u\neq k} \frac{x-u}{k-u}$$

3) N_1's output is

$$V_1 = e(g,g)^{rf_1(0)} = e(g,g)^{rs}$$

4) The data sharer computes

$$\frac{B_{ij}V_1}{e(\bar{C}, D)} = \frac{k_{ij}e(g,g)^{\alpha s}e(g,g)^{rs}}{e(g^{\beta s}, g^{\frac{\alpha+r}{\beta}})} = k_{ij}$$

By using the symmetric decryption algorithm and the symmetric key k_{ij}, the ciphertext C_{ij} is decrypted into the plaintext F_{ij} for $(i,j) \in (I, J)$. Finally, the data sharer retrieves the plaintext $\{F_{ij}, (i,j) \in (I, J)\}$.

IV. SECURITY ANALYSIS AND PERFORMANCE ANALYSIS

This section analyzes our proposed scheme's security and performance. We give the provable security analysis. Besides, we give the theoretical analysis and the prototype implementation for the performance analysis.

A. SECURITY ANALYSIS

The security of our proposed data sharing scheme is guaranteed by the following security results.

Theorem 1: Denote the adversary as $\mathcal{A}$. $\mathcal{A}$ interacts with the oracles of hash functions and the phase AB-E. The total number of the interaction can be bounded by the integer $\hat{q}$. Then, in the AB-E security game, $\mathcal{A}$'s advantage is $O(\frac{\hat{q}^2}{q})$.

Proof: The intuition of the proof process is given below. In order to design a secure AB-E phase, we have to solve the main challenge of defying against attacks from colluding users. In the access control tree $\mathbb{T}$, we have randomized users private keys. At the same time, we have embedded the secret sharing into the ciphertext by replacing the private keys. In order to decrypt the symmetric key k_{ij}, the attacker must recover $e(g,g)^{s\alpha}$. In order to do this, the attacker has to pair $\bar{C}$ from the ciphertext with the D component from some user's private key. The desired value $e(g,g)^{s\alpha}$ is gotten although it is blinded by some value $e(g,g)^{rs}$. It can be blinded out if and only if the user has the correct key components which satisfy the secret sharing scheme. Based on the randomization of the blinding value, collusion attacks can be resisted.

Due to the page limits, the detailed proof process can refer to [45]. ■

Theorem 2: Suppose that bilinear Diffie-Hellman (BDH) problem is difficult, our proposed data sharing scheme satisfies the S-Enc security against a chosen keyword attack in the random oracle model.

Proof: Let g be a generator of $\mathcal{G}_1$. If the attacker $\mathcal{A}$ wins in the S-Enc game, we can construct an algorithm $\mathcal{C}$ which can break the BDH problem. $\mathcal{C}$ simulates the challenger and interacts with $\mathcal{A}$. Given $(g, u_1 = g^a, u_2 = g^b, u_3 = g^c)$, $\mathcal{C}$'s goal is to calculate $v = e(g,g)^{abc}$.

KeyGen. $\mathcal{C}$ picks a random element $\bar{g}_2 \in \mathcal{G}_2$ and sends $(g, u_1, \bar{g}_2)$ to $\mathcal{A}$.

H_1-queries. To respond to H_1 queries, $\mathcal{C}$ maintains a list of tuples (W_j, h_j, a_j, c_j) called the H_1 list. H_1 list is initially empty. Upon the query $W_i \in \{0,1\}^*$, $\mathcal{C}$ responds below:

1) If $(W_i, h_i, a_i, c_i) \in H_1$-list, $\mathcal{C}$ responds with $H_1(W_i) = h_i$.
2) Otherwise, $\mathcal{C}$ generates a random coin $c_i \in \{0,1\}$ according to the bivariate distribution $\Pr[c_i = 0] = \frac{1}{q_T+1}$. $\mathcal{C}$ picks a random $a_i \in \mathcal{Z}_q$ and calculates $h_i = u_2 g^{a_i}$ if $c_i = 0$; $h_i = g^{a_i}$ if $c_i = 1$.
3) $\mathcal{C}$ adds the tuple (W_i, h_i, a_i, c_i) to the H_1-list and responds h_i to $\mathcal{A}$.

H_2-queries. It is a real hash function query and response. Let the query-response pair be (t, V). They are added into the H_2-list.

Trapdoor queries. Upon receiving the trapdoor query for the keyword W_i, $\mathcal{C}$ responds below:

1) Through running the H_1 oracle, $\mathcal{C}$ gets h_i which satisfies $H_1(W_i) = h_i$, where $(W_i, h_i, a_i, c_i) \in H_1$-list. If $c_i = 0$, then $\mathcal{C}$ reports failure and terminates.

2) Otherwise, define $h_i = g^{a_i}$ and $T_i = u_1^{a_i}$. Observe that $T_i = H_1(W_i)^a$ and therefore T_i is the correct trapdoor for W_i. $\mathcal{C}$ sends T_i to $\mathcal{A}$.

Challenge. $\mathcal{A}$ picks a pair of keywords (W_0, W_1) and submits them to $\mathcal{C}$, $\mathcal{C}$ performs the procedures below:

1) From the H_1 oracle, $\mathcal{C}$ obtains $h_0, h_1 \in \mathcal{Z}_q$ which satisfy $H_1(W_0) = h_0$ and $H_1(W_1) = h_1$. If both $c_0 = 1$ and $c_1 = 1$ then $\mathcal{C}$ reports failure and terminates.
2) Otherwise, at least there exists one $c_b = 0$ where $b \in \{0, 1\}$. $\mathcal{C}$ picks a random $J \in \{0, 1\}^{\lceil \log_2 q \rceil}$ and responds $C = (u_3, J)$.

More trapdoor queries. $\mathcal{A}$ continues to issue trapdoor queries for keyword W_i where $W_i \neq W_0, W_1$. $\mathcal{C}$ responds to these queries as in the phase *Trapdoor queries*.

Output. Finally, $\mathcal{A}$ outputs its guess $b' \in \{0, 1\}$ indicating the response C is the result of S-Enc$(W_{b'})$. $\mathcal{C}$ picks a random pair $(t, V) \in H_2$-list and calculates $\frac{t}{e(u_1,u_3)^{ab}}$ as its guess for $e(g, g)^{abc}$.

Thus, if the attacker $\mathcal{A}$ can break the S-Enc security, then an algorithm $\mathcal{C}$ can be constructed to break the BDH problem. Based on the difficulty of BDH problem, our proposed data sharing scheme satisfies the security of S-Enc. ■

Theorem 3: According to the trust on the PCS, our proposed data sharing scheme satisfies the data owner anonymity, *i.e.*, it is difficult to identify the data owner's real identity for the data sharer.

Proof: When the data sharer sends the queried keywords to PCS, PCS responds

$$V = (C_{ij}, B_{ij}, \bar{C}, \{E_l,\ E'_l\}_{l\in\mathcal{L}}, (i,j) \in (I,J)))$$

In the above response, $(B_{ij}, \bar{C}, \{E_l,\ E'_l\}_{l\in\mathcal{L}}, (i,j) \in (I,J)))$ have nothing to do with the data owner's real identity. $(C_{ij}, (i,j) \in (I,J)))$ is the ciphertext of the files by using the data owner's alias. Thus, they also have nothing to do with the data owner's real identity. Based on the trust on the PCS, the data owner's real identity also cannot be leaked from the PCS. Thus, our proposed data sharing scheme satisfies the data owner anonymity. ■

data confidentiality. Since our scheme satisfies the AB-E security, the unauthorized client cannot decrypt the $(B_{ij}, \bar{C}, \{E_l,\ E'_l\}_{l\in\mathcal{L}}, (i,j) \in (I,J)))$. Thus, the symmetric key k_{ij} cannot be gotten. The corresponding plaintext F_{ij} cannot also be gotten from C_{ij} since the symmetric key k_{ij} is unknown. Our proposed scheme satisfies the data confidentiality.

B. PERFORMANCE ANALYSIS

Computation and communication are two important elements for the performance analysis of our scheme. They are analyzed below:

Computation: Before the data has been uploaded, the data owner has to perform the phases Sym-Enc, AB-Enc and S-Enc. Compared to AB-Enc and S-Enc, Sym-Enc is more efficient. In AB-Enc, the encryption algorithm will require 2 exponentiations on $\mathcal{G}_1$ for each leaf in the ciphertext's access tree. Foe each tree leaf, the ciphertext size will include two elements of $\mathcal{G}_1$. The key generation algorithm requires 2 exponentiations for every attribute given to the user, and the private key consists of two group elements for every attribute. In S-Enc, in order to create the trapdoor for the keyword, data owner will perform one pairings and one exponentiation. In Retr, the data sharer will cost the computation. The decryption algorithm requires two pairings for every leaf of the access tree and (at most) one exponentiation for each node along a path from such a leaf to the root. On the other hand, in order to send (A, B) to PCS, the data sharer will compute one pairings and two exponentiations. In GenRetr, PCS will perform $\sum_{i=1}^{n} n_i$ pairings. We implement our scheme and show its computation performance below. Based on the modern computation technology, our proposed data sharing scheme is practical.

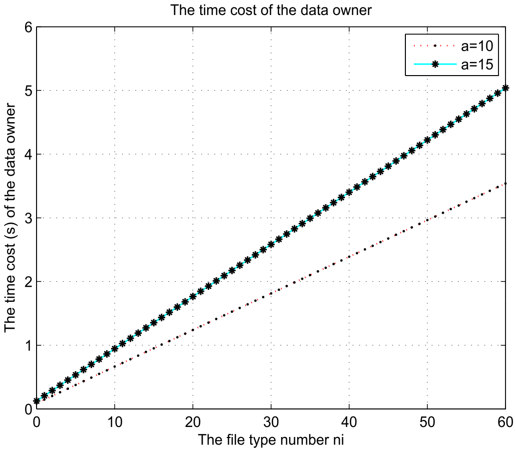

FIGURE 3. Time cost of data owner.

Implememtation: In order to implement our scheme and evaluate its computation cost, we simulated our concrete scheme by using C programming language with GMP Library (GMP-5.1.1) [46], Miracl Library [47] and PBC Library (pbc-0.5.13) [48]. Our scheme is implemented in the environment: PCS works on DELL PowerEdge R420 Server whose settings are listed below:

- CPU: Intel® Xeon® processor E5-2400 and E5-2400 v2 product families
- Physical Memory: 8GB DDR3 1600MHz
- OS: Ubuntu 13.04 Linux 3.8.0-19-generic SMP i686

The client works on an PC Laptop which has the following settings:

- CPU: CPU I PDC E6700 3.2GHz
- Physical Memory: DDR3 2G 1600MHz
- OS: Windows 7

In the implementation, we pick an elliptic curve with 160-bit group order whose security level is almost the same with 1024-bit RSA. Figure 3 depicts the time cost of data owner in the phase of Sym-Enc, AB-Enc, S-Enc, GenList. Suppose that the file F_i is classified into $\{F_{i1}, F_{i2}, \cdots, F_{in_i}\}$. X-label denotes the file type number n_i and Y-label denotes

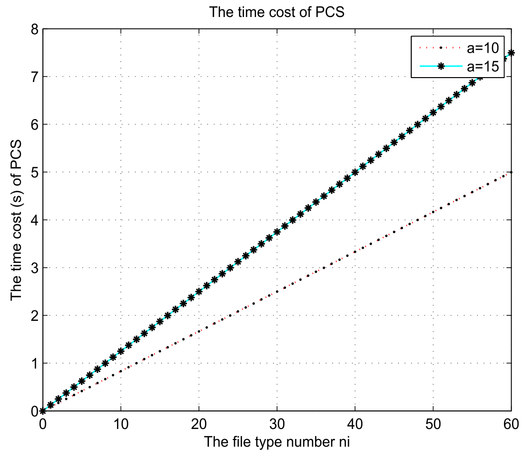

FIGURE 4. Time cost of PCS.

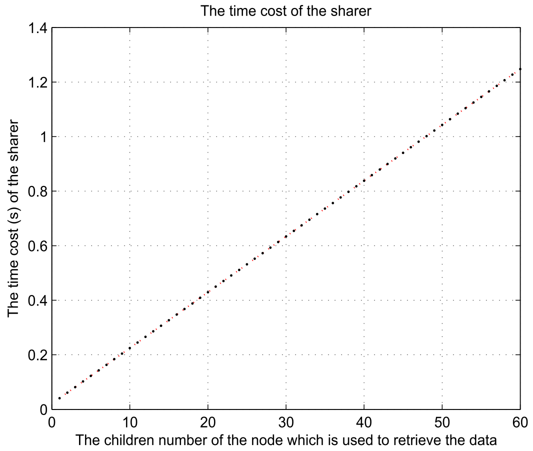

FIGURE 5. Time cost of sharer.

the time cost (second). The two curves (almost two lines) denote two different attribute number: a=10 and a=15. Figure 4 depicts the time cost of PCS. X-label denotes the file type number n_i and Y-label denotes the time cost (second) of PCS in the phase GenRetr. Figure 5 depicts the time cost of sharer in Retr. Suppose that the sharer will retrieve the remote data on the node N_j which has $\hat{n}$ leaf nodes. X-label denotes the number $\hat{n}$ and Y-label denotes the time cost (second) of the sharer in order to retrieve the remote data. These implementation figures show that our concrete data sharing scheme is fast and efficient.

Communication: National Bureau of Standards and ANSI X9 have determined the shortest key length requirements: RSA and DSA is 1024 bits, ECC is 160 bits [49]. Suppose that the data owner plans to upload 10T bits to PCS. These data is classified into 10^6 files where every file size is 10M bits. Let Sym-Enc expansion rate is α. In AB-Enc, the created ciphertext size is $160 * 10^6 + 160 * 2 + 160 * 4 * |\mathcal{L}|$. In S-Enc, the created ciphertext size is $160 * 2 * 10^6$. Thus, the uploaded data size approximately is $10 * 10^{12}(1 + \alpha) + 160 * 10^6 + 160 * 2 + 160 * 4 * |\mathcal{L}| + 160 * 2 * 10^6$ bits. On the whole, the expansion rate approximately is $\frac{10*10^{12}\alpha+160*10^6+160*2+160*4*|\mathcal{L}|+160*2*10^6}{10*10^{12}} \approx \alpha$. By using the similar analysis, in GenRetr, the responded data expansion rate is also approximately α. From the above analysis, our proposed data sharing scheme has low expansion rate. It is practical.

V. APPLICATION IN E-HEALTH RECORD

By using E-health, patient data are shared with different healthcare professionals. For E-health, many factors block the use of e-Health tools from widespread acceptance. Especially, patient records' privacy is the most important security issue. Most specifically, the E-health records need more strong privacy preservation. This main concern has to handle the confidentiality of the data and the anonymity of the patient. The same security problems also exist when the E-health records are uploaded to the public clouds. By using the phases *Sym-Enc, AB-Enc, S-Enc, GenList* of our scheme, the E-health records are encrypted and stored in the public clouds. When the authorized entity wants to access the remote E-health records which satisfy the specified conditions, it sends the corresponding challenge to PCS. By using the phase *GenRetr*, PCS sends the computed data V to the authorised entity. Upon receiving V, the authorized entity can retrieve the authorized data by using the phase *Retr* of our scheme. Thus, by using our proposed scheme, E-health records can be securely shared in the public clouds.

VI. CONCLUSION

In this paper, we proposed a data sharing scheme which can achieve the anonymity and data confidentiality in public clouds. We formalize the definition and the security model. Then, we designed a concrete data sharing scheme and gave the security proof. Security analysis showed our scheme is provably secure in the proposed security model. Performance analysis showed that our scheme is applicable.

REFERENCES

[1] C.-K. Chu, S. S. M. Chow, W.-G. Tzeng, J. Zhou, and R. H. Deng, "Key-aggregate cryptosystem for scalable data sharing in cloud storage,," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 2, pp. 468–477, Feb. 2014.

[2] Y. Tong, J. Sun, S. S. M. Chow, and P. Li, "Cloud-assisted mobile-access of health data with privacy and auditability," *IEEE J. Biomed. Health Inform.*, vol. 18, no. 2, pp. 419–429, Mar. 2014.

[3] Z. Pervez, A. M. Khattak, S. Lee, and Y.-K. Lee, "SAPDS: Self-healing attribute-based privacy aware data sharing in cloud," *J. Supercomput.*, vol. 62, no. 1, pp. 431–460, Oct. 2012.

[4] C. Fan, V. S.-M. Huang, and H.-M. Ruan, "Arbitrary-state attribute-based encryption with dynamic membership," *IEEE Trans. Comput.*, vol. 63, no. 8, pp. 1951–1961, Apr. 2014.

[5] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Advances in Cryptology—EUROCRYPT*. Interlaken, Switzerland: Springer-Verlag, May 2004, pp. 506–522.

[6] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 222–233, Jan. 2014.

[7] S.-H. Seo, M. Nabeel, X. Ding, and E. Bertino, "An efficient certificateless encryption for secure data sharing in public clouds," *IEEE Trans. Knowl. Eng.*, vol. 26, no. 9, pp. 2107–2119, Sep. 2014.

[8] L. A. Dunning and R. Kresman, "Privacy preserving data sharing with anonymous ID assignment," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 2, pp. 402–413, Feb. 2013.

[9] X. Chen, X. Huang, J. Li, J. Ma, W. Lou, and D. S. Wong, "New algorithms for secure outsourcing of large-scale systems of linear equations," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 1, pp. 69–78, Jan. 2015.

[10] X. Chen, J. Li, J. Weng, J. Ma, and W. Lou, "Verifiable computation over large database with incremental updates," *IEEE Trans. Comput.*, vol. 65, no. 10, pp. 3184–3195, Oct. 2016.

[11] C.-Z. Gao, Q. Cheng, X. Li, and S.-B. Xia, "Cloud-assisted privacy-preserving profile-matching scheme under multiple keys in mobile social network," *Cluster Comput.*, to be published, doi: 10.1007/s10586-017-1649-y.

[12] J. Shen, Z. Gui, S. Ji, J. Shen, H. Tan, and Y. Tang, "Cloud-aided lightweight certificateless authentication protocol with anonymity for wireless body area networks," *J. Netw. Comput. Appl.*, vol. 106, no. 15, pp. 117–123, Mar. 2018.

[13] J. Li *et al.*, "Secure distributed deduplication systems with improved reliability," *IEEE Trans. Comput.*, vol. 64, no. 12, pp. 3569–3579, Dec. 2015.

[14] J. Li, Y. Zhang, X. Chen, and Y. Xiang, "Secure attribute-based data sharing for resource-limited users in cloud computing," *Comput. Secur.*, vol. 72, pp. 1–12, Jan. 2018.

[15] J. Li, X. Huang, J. Li, X. Chen, and Y. Xiang, "Securely outsourcing attribute-based encryption with checkability," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 8, pp. 2201–2210, Aug. 2014.

[16] Y. Zhang, X. Chen, J. Li, D. S. Wong, H. Li, and I. You, "Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing," *Inf. Sci.*, vol. 379, pp. 42–61, Feb. 2017.

[17] W. Sun, S. Yu, W. Lou, Y. T. Hou, and H. Li, "Protecting your right: Attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud," in *Proc. INFOCOM*, Toronto, ON, Canada, Apr./May 2014, pp. 226–234.

[18] A. Rosenthal, P. Mork, M. H. Li, J. Stanford, D. Koester, and P. Reynolds, "Cloud computing: A new business paradigm for biomedical information sharing," *J. Biomed. Informat.*, vol. 43, no. 2, pp. 342–353, 2010.

[19] 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.

[20] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, "Patient controlled encryption: Ensuring privacy of electronic medical records," in *Proc. ACM Workshop Cloud Comput. Secur.*, Chicago, IL, USA, Nov. 2009, pp. 103–114.

[21] J. Sun and Y. Fang, "Cross-domain data sharing in distributed electronic health record systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 6, pp. 754–764, Jun. 2010.

[22] A. Bahga and V. K. Madisetti, "A cloud-based approach for interoperable electronic health records (EHRs)," *IEEE J. Biomed. Health Inform.*, vol. 17, no. 5, pp. 894–906, Sep. 2013.

[23] D. Anthony *et al.*, "Securing information technology in healthcare," *IEEE Security Privacy*, vol. 11, no. 6, pp. 25–33, Nov./Dec. 2013.

[24] M. Canim, M. Kantarcioglu, and B. Malin, "Secure management of biomedical data with cryptographic hardware," *IEEE Trans. Inf. Technol. Biomed.*, vol. 16, no. 1, pp. 166–175, Jan. 2012.

[25] M. Lesk, "Electronic medical records: Confidentiality, care, and epidemiology," *IEEE Security Privacy*, vol. 11, no. 6, pp. 19–24, Nov. 2013.

[26] S. Haas, S. Wohlgemuth, I. Echizen, N. Sonehara, and G. Müller, "Aspects of privacy for electronic health records," *Int. J. Med. Inform.*, vol. 80, no. 2, pp. e26–e31, Feb. 2011.

[27] R. Zhang and L. Liu, "Security models and requirements for healthcare application clouds," in *Proc. IEEE 3rd Int. Conf. Cloud Comput. (CLOUD)*, Miami, FL, USA, Jul. 2010, pp. 268–275.

[28] J. L. Fernández-Alemán, I. C. Señor, P. A. O. Lozoya, and A. Toval, "Security and privacy in electronic health records: A systematic literature review," *J. Biomed. Inform.*, vol. 46, no. 3, pp. 541–562, 2013.

[29] M. Ahmed, M. Ahamad, and T. Jaiswal, "Augmenting security and accountability within the ehealth Exchange," *IBM J. Res. Develop.*, vol. 58, no. 1, pp. 8:1–8:11, 2014.

[30] F. Xhafa, J. Li, G. Zhao, J. Li, X. Chen, and D. S. Wong, "Designing cloud-based electronic health record system with attribute-based encryption," *Multimedia Tools Appl.*, vol. 74, no. 10, pp. 3441–3458, May 2015.

[31] C. Wang, X. Liu, and W. Li, "Design and implementation of a secure cloud-based personal health record system using ciphertext-policy attribute-based encryption," *Int. J. Intell. Inf. Database Syst.*, vol. 7, no. 5, pp. 389–399, Sep. 2013.

[32] J. Li, Y. K. Li, X. Chen, P. P. C. Lee, and W. Lou, "A hybrid cloud approach for secure authorized deduplication," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 5, pp. 1206–1216, May 2015.

[33] P. Li *et al.*, "Multi-key privacy-preserving deep learning in cloud computing," *Future Generat. Comput. Syst.* vol. 74, pp. 76–85, Sep. 2017.

[34] J. Li, J. Li, D. Xie, and Z. Cai, "Secure auditing and deduplicating data in cloud," *IEEE Trans. Comput.*, vol. 65, no. 8, pp. 2386–2396, Aug. 2016.

[35] T. Li, Z. Liu, J. Li, C. Jia, and K.-C. Li, "CDPS: A cryptographic data publishing system," *J. Comput. Syst. Sci.*, vol. 89, pp. 80–91, Nov. 2017.

[36] J. Li, J. Li, X. Chen, C. Jia, and W. Lou, "Identity-based encryption with outsourced revocation in cloud computing," *IEEE Trans. Comput.*, vol. 64, no. 2, pp. 425–437, Feb. 2015.

[37] A. Beimel, "Secure schemes for secret sharing and key distribution," Ph.D. dissertation, Dept. Comput. Sci., Technion–Israel Inst. Technol., Haifa, Israel, 1996.

[38] R. Ostrovsky, A. Sahai, and B. Waters, "Attribute-based encryption with non-monotonic access structures," in *Proc. CCS*, Alexandria, VA, USA, Oct./Nov. 2007, pp. 195–203.

[39] R. Ostrovsky, A. Sahai, and B. Waters, *Attribute-Based Encryption With Non-Monotonic Access Structures*. Accessed: May 23, 2018. [Online]. Available: http://eprint.iacr.org/2007/323.pdf

[40] R. D'Souza, D. Jao, I. Mironov, and O. Pandey, "Publicly verifiable secret sharing for cloud-based key management," in *Progress in Cryptology–INDOCRYPT*. Chennai, India: Springer-Verlag, Dec. 2011, pp. 290–309.

[41] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute based encryption," in *Proc. ICALP*, Reykjavik, Iceland, Jul. 2008, pp. 579–591.

[42] A. Miyaji, M. Nakabayashi, and S. Takano, "New explicit conditions of elliptic curve traces for FR-reduction," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. 84, pp. 1234–1243, May 2001.

[43] H. Wang, Q. Wu, B. Qin, and J. Domingo-Ferrer, "Identity-based remote data possession checking in public clouds," *IET Inf. Secur.*, vol. 8, no. 2, pp. 114–121, Mar. 2014.

[44] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," in *Advances in Cryptology–ASIACRYPT*, Gold Coast, QLD, Australia: Springer-Verlag, Dec. 2001, pp. 514–532.

[45] 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.

[46] *The GNU Multiple Precision Arithmetic Library (GMP)*. Accessed: May 23, 2018. [Online]. Available: http://gmplib.org

[47] *Multiprecision Integer and Rational Arithmetic C/C++ Library (MIRACL)*. [Online]. Available: http://certivox.com

[48] *The Pairing-Based Cryptography Library (PBC)*. Accessed: May 23, 2018. [Online]. Available: http://crypto.stanford.edu/pbc/howto.html

[49] Research C. *SEC 2: Recommended Elliptic Curve Domain Parameters*. Accessed: May 23, 2018. [Online]. Available: http://www.secg.org/collateral/sec_final.pdf

HUAQUN WANG was born in Jining, Shandong, China, in 1974. He received the B.S. degree in mathematics education from Shandong Normal University, China, in 1997, and the M.S. degree in applied mathematics from East China Normal University, China, in 2000, and the Ph.D. degree in cryptography from the Nanjing University of Posts and Telecommunications in 2006. He is currently a Full Professor with the Nanjing University of Posts and Telecommunications. His research interests include applied cryptography, blockchain, network security, and cloud computing security.