A Searchable CP-ABE Privacy Preserving Scheme

Tao Feng[1], Xiaoyu Yin[1], Ye Lu[2], Junli Fang[1], and Fenghua Li[3]
(Corresponding author: Tao Feng)
School of Computer and Communication, Lanzhou University of Technology, China[1]
College of Electrical and Information Engineering, LanZhou University of Technology, China[2]
The State Key Laboratory of Information Security, Institute of Information Engineering,
Chinese Academy of Sciences, Beijing[3]
(Email: fengt@lut.cn)
(Received Jan. 9, 2018; Revised and Accepted May 22, 2018; First Online Mar. 12, 2019)

Abstract

The separation of users and data in the cloud storage system brings new security threats. The cloud storage scheme with single encryption mechanism has been unable to meet new demands. Aiming at the above problems, the PEKS was introduced into the CP-ABE scheme with multiple authorities to propose a searchable CP-ABE privacy-preserving scheme which supports the direct revocation of users. The access control of users is achieved by the central authority, which avoids the security risks caused by submitting the private keys and access structure to cloud server. The collusion problems are solved by using 1-out-of-n Obvious Transfer algorithm and associating the private key components with random identity token of users. The privacy of keywords is guaranteed though the new improved algorithm. Based on the DBDH assumption, the security of proposed scheme is proved in the random oracle model.

Keywords: Attribute-based Encryption; Cloud Storage; Privacy Preserving; Searchable Encryption

1 Introduction

1.1 Background

Cloud storage services as an emerging data management technology develops rapidly, which achieves data storage, searching and sharing by providing the dynamic network resources for users. However, the separation of data owner and data in the cloud storage system brings new security threats, and the frequent information leakage has triggered the trust crisis of cloud storage service. The privacy preserving gradually becomes the core issue in cloud storage research. In order to apply the cloud storage system to the core data management and realize the secure data sharing mechanism, the privacy preserving mechanism in the attribute-based encryption scheme must be more perfect and the security of the system needs to be further improved. The research about attribute-based encryption cloud storage systems has been relatively mature, but the cloud storage system with single encryption mechanism is unable to meet the users' new demands. The cloud storage system with various encryption mechanisms becomes the research hotspot [5, 8, 12, 13, 20, 24]. Because of the complexity of network environment, there still are some privacy disclosure problems in the cloud storage scheme with hybrid encryption mechanisms including the leak of attribute information, identity information disclosure, and data breaches in the cloud.

1.2 Related Works

Researchers have proposed a number of ciphertext-policy attribute-based encryption (CP-ABE) cloud storage schemes and some privacy protection measures. For the problem of content privacy disclosure, the measures usually adopted are data segmentation and encryption [2]. Besides, the secure revocation mechanism [7] can also ensure that data will not be stolen by illegal users. Attribute privacy preserving is mainly carried out in two aspects: access structure and users' attribute set. By hiding the access strategy [28] and solving the collusion problem the attribute privacy can be well protected. With the deepening of research, the function of cloud storage scheme with attribute-based encryption (ABE) is more perfected, but the privacy disclosure problems still exist in the current ABE cloud storage schemes. The single key generator may lead to users' attribute set leakage and bring vulnerability to the system. For instance, the scheme in literature [29] outsources the calculations of encryption, decryption and key generation to the cloud server, which greatly reduces the computing and communication overhead. Feng *et al.* proposed a decentralized ciphertext-policy attribute based encryption (CP-ABE) scheme in literature [4] to avoid the system vulnerabil-

ity caused by central authority. In literature [19], Li *et al.* also proposed a CP-ABE cloud storage scheme with multiple attribute authorities which outsourcing the bilinear pairing operation to the cloud server. Meanwhile the group keys were introduced into the attribute authority to realize the efficient and fine-grained revocation mechanism. In addition, the multiple attribute authorities (AAs) may recover users' attribute set though collusion, resulting in the disclosure of users' attribute information. Aiming at the collusion problems, Jung *et al.* proposed an attribute-based encryption scheme in [15] by improving the anonymous scheme in [14] and introduced the 1-out-of-n Oblivious Transfer into the multi-authority attribute-based encryption scheme preventing the collusion between the attribute authorities. In addition, users may collude to obtain the private key and decrypt the data without permission, causing the leakage of content privacy in practical applications. Guan *et al.* [6] introduced the attribute management server (AMS) into the scheme to assign the attribute authority for users according to users' attribute set. Attribute name was used to interact and the attribute value was hid in the scheme which still has the privacy disclosure problems caused by collusion. The above schemes have realized the access control to the data, but the privacy protection mechanism is not complete. Besides, the cloud system with single attribute-based encryption mechanism can't realize the search operation of ciphertext.

Searchable encryption is very suitable for ciphertext search environment of the cloud storage and its application prospect is very broad [9,11,16,18]. Users can search and update the data files stored in the cloud though cloud storage system based on searchable encrypted. Wang *et al.* proposed a mixed index structure in [23]. In the scheme, the static index (SI) and dynamic index (DI) were used in the first-time searches and repeated searches respectively, which reduced the complexity of the search operation. Moreover, the scheme also achieved the function of ciphertext updating by means of the dynamic index. The third-party is permitted to obtain the keyword search trapdoor to perform the ciphertext search operation in the public-key encryption with keyword search (PEKS) mechanism. However, in this data sharing mechanism, there still are the privacy disclosure problems brought by the keyword guessing attack that can't be ignored. Xu *et al.* [25] presented a public-key encryption with fuzzy keyword search scheme which can against the keyword guessing attack. The scheme transformed from the anonymous identity encryption scheme. And the mechanism that many keywords sharing one fuzzy search trapdoor solves the problem of privacy leakage caused by third-party stealing keywords. Fuzzy search trapdoor was sent to the untrusted server for ciphertext matching and filtering, and the exact trapdoor was used for local secondary filtering to get the matched ciphertext. With the method of authenticating the keywords, Huang *et al.* prevented the untrusted server from recovering the keywords by keyword guessing attack in literature [10]. The literature [25] and literature [10] have both realized the public-key encryption with keyword search scheme which can resist keyword guessing attack. The researchers established the users' privacy protection mechanism and solved the problem of users' privacy disclosure problem caused by keywords leakage. Researches on cloud storage system based on searchable encryption have been relatively mature, but there still are some security risks in the searchable encryption system as lacking of fine-grained access control.

Secure data sharing in complex network environments requires not only a complete privacy protection mechanism but also efficient and robust system functionalities. In order to achieve both access control and ciphertext search operations, the searchable encryption technology is introduced into the current attribute-based encryption cloud storage system. The researchers have proposed the cloud storage sharing mechanism with multiple encryption technologies. In [26], Yang *et al.* achieved fine-grained access control over searchable encryption schemes through ciphertext-policy attribute-based encryptio. They also achieved the concealment of keywords and the direct revocation of users. But the single key generator may lead to the leakage of users' attribute set. Once the generator is breached, it will bring the inevitable damage to the system. In addition, the access structure is uploaded to the semi-trusted cloud server in the scheme, which may cause the problems of sensitive attribute information disclosure. Wang *et al.* also used the multiple encryption mechanism in [21] and proposed a multi-user, fine-grained searchable encryption scheme, which adopted the hybrid cloud structure. In the structure, public cloud was used to achieve access control and ciphertext search operations. The security of cloud storage services was guaranteed by the re-encryption calculation of private cloud. But the scheme requires users to submit the private key to the cloud server for access control, which inevitably increased the risk of privacy disclosure. The other problem of the scheme is that the trapdoor generation process lacks privacy protection mechanism for keywords. In [17], Li *et al.* proposed a searchable ciphertext-policy attribute-based encryption scheme, in which fine-grained attribute revocation was realized via the version number and the access structure was hid. Meanwhile, the ciphertext search operation was achieved in the scheme and the computation of ciphertext updating was decreased by using homomorphic encryption. But the functions including key updating and re-encryption calculating were performed with a single authority, which brought the system inevitable vulnerability.

1.3 Our Contribution

In order to improve the security and practicability of the existing cloud storage schemes, this paper proposes a searchable ciphertext-policy attribute-based encryption privacy preserving scheme. The capability of cloud storage system is extended with the PEKS and CP-ABE. And the privacy disclosure problems in the current hy-

brid cloud storage systems are solved by optimizing the algorithm and improving the system structure.

1) This scheme solves the problems that the ciphertext can't be searched in attribute-based encryption schemes by introducing the PEKS into the multi-authority CP-ABE cloud storage schemes. Moreover, the scheme adopts the direct revocation to realize the revocation mechanism of users' searching rights.

2) The authority in the proposed scheme is composed of two parts, central authority and attribute authorities. The access control of the users' searching permissions is accomplished by the central authority. To prevent malicious user colluding with each other, the random identity token (RID) of users is introduced into the calculation of privacy key. What's more, the 1-out-of-n Obvious Transfer algorithm is used in the process of request and distribution so that to avoid the collusion caused by the attribute authorities.

3) The security of algorithms generating ciphertext of keywords and trapdoors is improved though the random numbers and user key (UK), which protects the privacy of keyword in the process of searching. Finally, we prove the security of proposed scheme based on the decisional bilinear Diffie–Hellman assumption in the random oracle model and analysis the performance of the cloud storage scheme.

The rest of this paper is arranged as follows. In Section 2 are some preliminaries related to the proposed scheme. The system model and threats model are presented in Section 3. The specific algorithm of searchable CP-ABE privacy preserving scheme are all given in Section 4. The security of proposed scheme is proved in Section 5. The analysis of privacy preserving and performance are described in Sections 6. Finally, conclusion and prospect are in Section 7.

2 Preliminaries

In this section, we introduce some definitions related to our schemes.

2.1 Bilinear Map

Definition 1. *Let G_1 and G_2 be two groups of prime order p and the generator of G_1 is g. The finite field of prime order p is defined as Z_p, the set of integers $\{0, 1, \ldots, , p-1\}$. A bilinear mapping $e : G_1 \times G_1 \to G_2$ satisfies the following properties:*

- *Bilinearity: For any $u, v \in G_1, a, b \in Z_p$, it has $e(u^a, v^b) = e(u, v)^{ab}$.*
- *Non-degeneracy: there exists $u, v \in G_1$ such that $e(g, g) \neq 1$.*
- *Computability: For any $u, v \in G_1$, there is an efficient bilinear mapping computation $e(u, v)$.*

2.2 Decision Bilinear Diffie–Hellman (DBDH) Assumption

Definition 2. *DBDH problem in group G of prime order p with generator g is defined as follows: let $g^a, g^b, g^c \in G$ and $e(g, g)^{abc} = e(g, g)^z$, and then decide whether $z = abc$ or z is a random number where $a, b, c, z \in Z_p$.*

Definition 3. *The DBDH assumption is that no probabilistic polynomial-time algorithm has a non-negligible advantage in solving the DBDH problem [22].*

2.3 CP-ABE and PEKS

Ciphertext-policy attribute-based encryption (CP-ABE) is a public-key encryption mechanism proposed by Bethencourt *et al.*, which implements fine-grained access control by encrypting data with access structure. In CP-ABE, the ciphertext is related to the access structure, and the users' private keys are associated with their attribute set. Basic algorithms usually include initialization, encryption, key generation and decryption. Ciphertext-policy attribute-based encryption cloud storage model is illustrated in Figure 1.

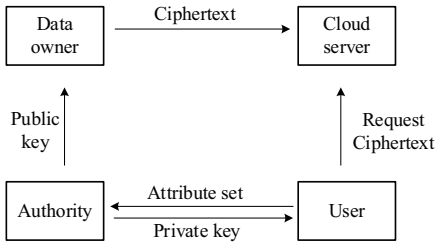

Figure 1: CP-ABE cloud storage mode

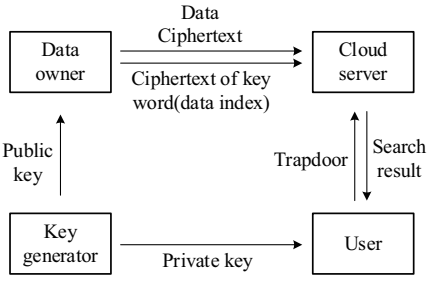

Figure 2: PEKS cloud storage model

Figure 2 shows the cloud storage model based on public key encryption with keyword search (PEKS) which is proposed by Boneh *et al.* in [1]. The keywords extracted from shared data are encrypted with the public key by the data owner, which generates the keyword ciphertext, the data index. The private key is used to encrypt the search keyword to generate the trapdoor by users. The ciphertext search operation is completed by matching the index and the trapdoor. The basic algorithm of PEKS is composed of key generation, encryption, trapdoor generation and test.

3 System Model

In this section, we will describe the basic structure as well as threats model of the proposed scheme.

3.1 Basic Structure

We propose a searchable CP-ABE privacy preserving scheme shown in Fig. 3. There are five participating entities in our scheme including data owner, users, cloud server, N attribute authorities and the central authority.

1) Data owner: In the stage of setup, data owner generates a key pair including index key and trapdoor key to encrypt the keyword. Then, data owner extracts the keywords from the shared data and encrypts the keywords as data index with the index key. The access structure is formulated for encrypting the trapdoor key and shared data. The calculated verification (VR) and ciphertext of trapdoor key are sent to central authority, while the data index and ciphertext are transmitted to the cloud server.

2) User: Users need to register themselves to get the user key (UK) and the random identity token (RID). By asking attribute authorities for private key, users can decrypt ciphertext of trapdoor key and calculate verification of user (VR') submitted to the central authority for permissions validation. And then, users encrypt the search keywords with the trapdoor key and UK, which generates trapdoor sent to the cloud server for ciphertext searching. After receiving the matched ciphertext, users can recover the shared data.

3) Cloud server: The storage and searching of data ciphertext are executed by the cloud server. Taking the UK, data index, and trapdoor as input, cloud server matches the data ciphtertext and returns the result to users. In the revocation phase, it also needs to achieve the direct revocation of users.

4) Central authority: In the register phase, UK and RID are generated by central authority (CA) which sends the trapdoor key ciphertext to users and verifies users' access permissions. If users pass the validation, UK and RID will be sent to cloud server for ciphertext searching.

5) Attribute authorities: Setup and generate the master key and public key which is sent to data owner for encrypting. The attribute authorities (AAs) respond the request of private key and generate the corresponding private key components after receiving users' attribute set.

3.2 Threats Model

In the proposed scheme, only the central authority is fully trusted. The attribute authorities will honestly generate and distribute the private key for users, but they may collude with each other to steal users' attribute information. The cloud server is honest but curious. It will abide by the protocol returning the searched ciphertext to users and revoke users' search permissions. However the semi-trusted cloud server may steal the shared data and information of users internally. Users may collude to decrypt the data that they don't have permissions.

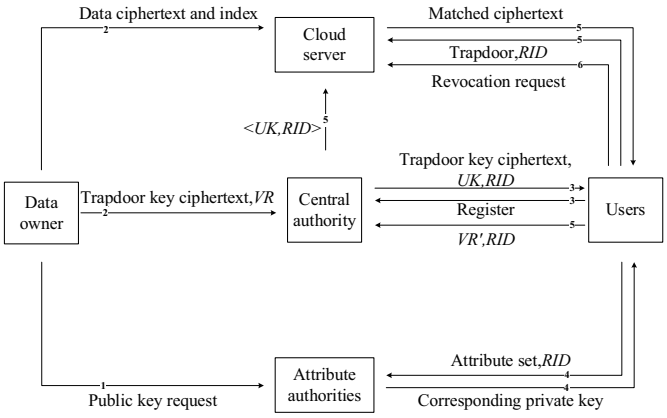

Figure 3: A searchable CP-ABE privacy preserving scheme

4 Concrete Algorithm

In this section, we will give the specific algorithm construction of searchable CP-ABE privacy protection scheme.

4.1 Setup

1) AAs setup $\alpha_i \rightarrow (PK, MK)$. Any one of attribute authorities chooses a bilinear group G of prime order p with generator g. AA_i chooses $\alpha_i \in Z_p$ and calculates $Y_i = e(g,g)^{\alpha_i}$ sent to the rest of AAs which all need to calculate $Y = \prod_{i \in AA} Y_i = e(g,g)^{\sum \alpha_i}$. The public key of system is $PK = \{G, g, Y = e(g,g)^{\sum \alpha_i}\}$. Each attribute authority randomly chooses $N-1$ integers $S_{ij} \in Z_p, (j \in \{1,2,\ldots N\} \backslash \{i\})$ and calculates $g^{S_{ij}}$ sent to all other authorities. After receiving the $N-1$ $g^{S_{ji}}$, AAs all calculate parameter x_i follows which belong to Z_p and satisfy $\prod_{i \in AA} x_i = 1 \bmod p$.

$$x_i = \left(\prod_{j \in \{1,\ldots,N\} \backslash \{i\}} g^{S_{ij}} \right) / \left(\prod_{j \in \{1,\ldots,N\} \backslash \{i\}} g^{S_{ji}} \right) = g^{\left(\sum_{j \in \{1,\ldots,N\} \backslash \{i\}} g^{S_{ij}} - \sum_{j \in \{1,\ldots,N\} \backslash \{i\}} g^{S_{ji}} \right)} \quad (1)$$

In the initialization phase, the generated master key is $MK = \{\alpha_i, x_i\}$. Each AA_i chooses random number $\gamma_i \in Z_p$ and calculates $x_i \cdot g^{\gamma_i}$ using for generating private key. The $x_i \cdot g^{\alpha_i} \cdot g^{\gamma_i}$ generated by AA_i is shared with the other attribute authorities and any

one of them calculates $Y' = \prod x_i g^{\alpha_i} g^{\gamma_i} = g^{\sum \alpha_i + \sum \gamma_i}$ sent to data owner.

2) Data owner setup. Choose a bilinear group G_1 of prime order p with generator g_1. Let $H_1 : \{0,1\}* \to G_1$ be the hash function. With choosing the random number η, μ, data owner calculates $PK = \{g_1, g_1^{\eta}\}$ and $SK = \eta$, which called index key (IK) and trapdoor key (TK) respectively in the scheme.

4.2 IndexGen $(W, IK) \to I_W$

Data owner extracts the keyword from shared data and encrypts the key words with index key and random numbers τ, μ. The index of shared data is calculated as follows:

$$I_W = (I_1, I_2) = (g_1^{\mu\tau}, e(H_1(W)^{\mu}, g_1^{\eta\tau})) \tag{2}$$

4.3 Encrypt $(M, TK, T_p, PK) \to C_M, C_{TK}, VR$

First, the algorithm choose a polynomial q_x for each node x in $\{T_p\}_{p\in\{0,\ldots,r-1\}}$. The degree d_x of polynomial q_x should less than the threshold value k_x. Starting from the root node R_p, the algorithm randomly picks $S_0 \in Z_p$ and sets $q_{R_p}(0) = S_0$ and the other coefficients of q_{R_p} are picked randomly. The attribute set in access tree is defined as A_{T_p}. With picking a random element $h \in Z_p$, the ciphertext is created as:

$$\begin{aligned} C_{TK} = \big\langle &\{T_p\}_{p\in\{0,\ldots,r-1\}}, E_0 = TK \cdot Y^{S_0}, C = g^{hS_0}, \\ &\hat{C} = (Y')^{h^{-1}}, \{C_i = g^{q_i(0)}, C_i' = H(att(i))^{q_i(0)}\} \\ &i \in A_{T_p}, p \in \{0,\ldots,r-1\}\big\rangle \\ C_M = \langle &E_1 = M \cdot Y^{S_0}, C = g^{hS_0}, \hat{C} = (Y')^{h^{-1}}\rangle \end{aligned} \tag{3}$$

And the verification used to verify the privilege of users is computed as $VR = \{Y^{S_0}\}_{p\in\{1,\ldots,r-1\}}$.

4.4 Enroll $\zeta_i \to UK, RID$

This algorithm enrolls the users who want to join the system and picks the user key (UK) randomly and generates a random sequence as the random identity of users (RID).

4.5 TrapdoorGen $(W', TK, UK, \lambda) \to T_{W'}$

In this algorithm, the random number λ is picked and the calculation of trapdoor is

$$T_{W'} = (T_1, T_2) = (\lambda \cdot UK, H_1(W')^{\lambda \cdot TK}) \tag{4}$$

4.6 Test $(RID, I_W, T_{W'}) \to \{C_M\}$

According to the RID submitted by users and the corresponding UK this algorithm performs the matching calculation $e(T_2, I_1^{UK}) = I_2^{T_1}$ like follows:

$$e(H_1(W)^{\lambda \cdot TK}, g_1^{\mu \cdot \tau \cdot UK}) = e(H_1(W')^{\mu}, g_1^{\eta \cdot \tau})^{UK \cdot \lambda} \tag{5}$$

If users' search keywords are same to those included in the index, the equation will be established. The cloud server sets $result = \{C_M\}$ and returns result to users. If not, the cloud server sets $result = \varnothing$ returned to users.

4.7 KeyGen $\{RID, PK, MK, A_u\} \to SK_{RID}$

For any attribute $k \in A_u$ every AA_i picks a random number $\beta_{RID,k} \in Z_p$ and calculates the private key components $H(att(k))^{\beta_{RID,k}}, D_k' = g^{\beta_{RID,k}}$ sent to user with $x_i \cdot g^{\gamma_i}$ where A_u is the attribute set of user. User calculates as:

$$\begin{aligned} D_k &= H(att(k))^{\beta_{RID,k}} \cdot \prod (x_i \cdot g^{\gamma_i}) \\ &= H(att(k))^{\beta_{RID,k}} \cdot g^{\sum \gamma_i} \end{aligned} \tag{6}$$

By combining the private key components, users can get the private key as $SK_{RID} = \{D_k, D_k' = g^{\beta_{RID,k}}\}$.

4.8 Decrypt $(C_M, C_{TK}, SK) \to (M, TK, VR')$

By calling this algorithm recursively, the TK and validation of user (VR') can be calculated.

1) If the node x is a leaf node and its attribute is i, the algorithm defined as follows.

If $i \in A_u$:

$$\begin{aligned} &DecryptNode(CT, SK, x) \\ &= \frac{e(D_k, C_x)}{e(D_k', C_x')} \\ &= \frac{e(H(att(i))^{\beta_{RID,k}} \cdot g^{\sum \gamma_i}, g^{q_x(0)})}{e(g^{\beta_{RID,k}}, H(att(i))^{q_x(0)})} \\ &= e(g,g)^{(\sum \gamma_i) \cdot q_x(0)} \end{aligned} \tag{7}$$

If $i \notin A_u$, the algorithm return $\varnothing$.

$$\begin{aligned} F_x &= \prod F_z^{\Delta_{index(z), S_x'}(0)} \\ &= \prod (e(g,g)^{(\sum r_i) \cdot q_z(0)})^{\Delta_{index(z), S_x'}(0)} \\ &= \prod (e(g,g)^{(\sum r_i) \cdot q_{parrent(z)}(index(z))})^{\Delta_{index(z), S_x'}(0)} \\ &= \prod (e(g,g)^{(\sum r_i) \cdot q_x(index(z))})^{\Delta_{index(z), S_x'}(0)} \\ &= e(g,g)^{(\sum \gamma_i) \cdot q_x(0)} \end{aligned} \tag{8}$$

2) If x is not a leaf node, the nodes z, children nodes of x, call $DecryptNode(CT, SK, z)$ and write the outputs as F_z. Let S_x be an arbitrary k_x -sized set of child nodes z with the index S_x'. By using polynomial interpolation the calculation is as follows.

After getting the ciphertext of trapdoor key, users call the decryption algorithm recursively starting from the root node R_p and calculate the verification of user (VR') as follows.

$$DecryptNode(C_{TK}, SK, R_p) = e(g,g)^{S_0 \sum \gamma_i} \tag{9}$$

If users' attribute set meet the access tree, they can decrypt the ciphertext of trapdoor key as:

$$\frac{E_0}{\frac{e(C,\hat{C})}{e(g,g)^{S_0\sum\gamma_i}}} = \frac{TK\cdot Y^{S_0}}{\frac{e(g^{hS_0},(g^{\sum\alpha_i+\sum\gamma_i})^{h^{-1}})}{e(g,g)^{S_0\sum\gamma_i}}} = \frac{TK\cdot e(g,g)^{(\sum\alpha_i)\cdot S_0}}{e(g,g)^{S_0\sum\alpha_i}} = TK \quad (10)$$

The shared data can be recovered as

$$M{=}E_1/\frac{e(C,\hat{C})}{VR'} \quad (11)$$

4.9 Revoke

By generating the list of *UK* and *RID*, the direct revocation to users' search permission can be achieved. In the phase of revocation, users submit their *RID* and then the cloud server remove the corresponding item of *UK* and *RID* from the list. If cloud server can't find the corresponding *UK* in the process of ciphertext matching, the ciphertext search operation is terminated and then the cloud server returns the information that authentication fails.

5 Security Proof

In this section, the security of proposed scheme is proved in the random oracle model.

Lemma 5.1. *Based on DBDH assumption, if the scheme in [15] is secure against chosen plaintext attacks (CPA) in the random oracle model, our scheme is secure against CPA.*

Proof. Suppose there exists a probabilistic polynomial time adversary A can attack our scheme with advantage ε. We prove that the following DBDH game can be solved by the challenger B with advantage $\frac{\varepsilon}{2}$.

Let $e: G\times G\to G_0$ be a bilinear map where G is a cyclic group of prime order p with generator g. First, the challenger B randomly picks $a,b,c,z\in Z_p, \theta\in\{0,1\}$ and sets tuple $(g,A,B,C,Z)=(g,g^a,g^b,g^c,e(g,g)^{abc})$ if θ=0. Otherwise if θ=1, the tuple is set to $\left(g,g^a,g^b,g^c,e(g,g)^z\right)$.

Initialization: The adversary controls part of attribute authorities where at least two authorities are not controlled by the adversary and remaining authorities are controlled by challenger B. The adversary A declares the challenged access tree T_0' of which some attributes are managed by the simulator's authorities.

Setup: The challenger sets $a=\sum\gamma_i, b=\frac{\sum\alpha_i}{\sum\gamma_i}, c=s_0$ where $\gamma_i,\alpha_i,s_0\in Z_p$ are randomly picked and gives Y and Y' to the adversary.

Query Phase 1: The adversary queries for the private keys according to attribute set and none of the attribute set satisfy the access tree. After receiving the private key queries from A with *RID*, the challenger randomly picks $\beta_{RID,k}\in Z_p$ and calculates private key components for every attribute $k\in A_u$ as follows: $D_k=H(att(k))^{\beta_{RID,k}}\cdot g^{\sum\gamma_i}, D_k'=g^{\beta_{RID,k}}$.

Query Phase 2: Repeat Phase 1 adaptively.

Guess: The adversary A submits the guess θ' of θ. When $\theta=\theta'$, the simulator represented challenger B outputs $\left(g,g^a,g^b,g^c,e(g,g)^{abc}\right)$ if θ=0, otherwise it outputs a DBDH tuple $\left(g,g^a,g^b,g^c,e(g,g)^z\right)$ composed by five random elements.

When θ= 1, the adversary A can't get any useful information and the advantage is $\Pr=\frac{1}{2}$. And the advantage is $\Pr=\frac{1}{2}+\varepsilon$ when θ=0. Therefore, the advantage of probabilistic polynomial time adversary in the DBDH game is $\Pr(\theta'=\theta)-\frac{1}{2}=\frac{1}{2}\left(\frac{1}{2}+\varepsilon\right)+\frac{1}{2}\cdot\frac{1}{2}-\frac{1}{2}=\frac{\varepsilon}{2}$. To conclude, if the adversary has non-negligible advantage in the constructed game, he can solve the DBDH problem with the non-negligible advantage $\frac{\varepsilon}{2}$. Based on the DBDH assumption, there is no adversary has significant advantage in our security game and our scheme is secure. □

Lemma 5.2. *If DBDH assumption is tenable and the scheme in [10] is semantically secure in the random oracle model, our scheme is semantically secure in the random oracle model.*

Proof. Assume the probabilistic polynomial time (PPT) adversary A can break our scheme with non-negligible advantage and then we construct a PPT algorithm B simulating the oracle to solve the DBDH problem. The adversary issues at most q_H, q_T, q_C to the hash oracle, trapdoor oracle, ciphertext oracle respectively.

Hash Oracle: The adversary queries the hash H_1 and gives a keyword W_i. The simulator B randomly picks $a_i\in Z_p$ and flip a random coin c_i such that $\Pr[c_i=0]=\delta$. If $c_i=0$, the simulator B calculates $g_1^{\frac{z}{\mu}}\cdot g_1^{a_i}=h_i$, and sets $g_1^{a_i}=h_i$ if $c_i=1$. Then, B adds tuple $[W_i,h_i,a_i,c_i]$ to the list L_{H_1} and sets $H_1(W_i)=h_i$ as the hash value of the keyword W_i. The hash value h_i is returned to the adversary A.

Trapdoor Oracle: Given a keyword W_i, the simulator retrieves tuple $[W_i,h_i,a_i,c_i]$ in the list L_{H_1}. if $c_i=0$, the simulator B aborts and out puts the guess b' of b. If $c_i=1$, B randomly chooses $\beta_i,\rho\in Z_p$ and calculates the true trapdoor as $T_i=(T_1,T_2)=(\beta_i\cdot\rho, H_1(W_i)^{\beta_i\cdot\rho})$. And then B return the trapdoor to adversary A.

Ciphertext Oracle: Given a keyword W_i, the simulator retrieves tuple $[W_i,h_i,a_i,c_i]$ in the list L_{H_1}. If $c_i=0$, the simulator B aborts and outputs the guess b' of b. If $c_i=1$, B randomly chooses $\eta,\tau,\mu\in Z_p$

and calculates the true ciphertext $C_i = (C_1, C_2) = (g_1^{\mu\tau}, e(H_1(W_i)^{\mu}, g_1^{\eta\tau}))$ returned to the adversary A.

Challenge: The adversary chooses the keyword W_0W_1 that he wants to challenge. B performs the above algorithm and retrieves the tuples $[W_0, h_0, a_0, c_0]$ and $[W_1, h_1, a_1, c_1]$. If $c_0 = 1$ and $c_1 = 1$, the simulator B aborts and outputs the guess b' of b. If $c_0 = 0$ or $c_1 = 0$, let $\hat{b}$ be the bit such that $c_{\hat{b}} = 0$ and we have $h_{\hat{b}} = g_1^{\frac{z}{\mu}} \cdot g_1^{a_{\hat{b}}}$.

The simulator B calculates $C_2 = Z \cdot e(g_1^{\eta}, g_1^{\tau})^{\mu a_{\hat{b}}}$. If $Z = e(g_1, g_1)^{\eta\tau z}$, then $C_2 = e(g_1, g_1)^{\eta\tau(\mu a_{\hat{b}} + z)} = e(h_{\hat{b}'}^{\mu}, g^{\eta\tau}), C_1 = g_1^{\mu\tau}$. B returns $C = (C_1, C_2)$ to A. The adversary continues to query for W_i where the only restriction is $W_i \neq W_0$,W_1. At last, the adversary submits a guess $\hat{b}'$ of $\hat{b}$. If $\hat{b}' = \hat{b}$, the simulator B outputs $b' = 0$. And it outputs $b' = 1$ if $\hat{b}' \neq \hat{b}$.

The probability that the simulator B doesn't aborts is $\Pr[B] = (1-\delta)^{q_T+q_C}(1-(1-\delta)^2)$. It's no-negligible because it approximately equals to $\frac{2}{(q_T+q_C)e}$. If the adversary can break the algorithm of our scheme, the simulator B can succeed in distinguish that Z is equal to $e(g_1, g_1)^{\eta\tau z}$ or a random element. The probability that simulator B succeeds in guessing b' of b is $\Pr[b' = b] = \frac{1}{2} + \varepsilon \cdot \Pr[B]$. If ε is no-negligible, so is $Pr[b' = b] - \frac{1}{2}$, the advantages of solving the DBDH problems by simulator B. based on the DBDH assumption, there is no adversary can break our algorithm with no-negligible advantage and our scheme is safe. □

Theorem 5.3. *If DBDH assumption is tenable, our scheme is safe in the random oracle model.*

Proof. Directly derived from Lemma 5.1 and Lemma 5.2. □

6 Analysis and Comparison

6.1 Privacy Preserving Analysis

6.1.1 Content Privacy

This paper adopts the CP-ABE algorithm, a public-key encryption mechanism, to encrypt the shared data, which is safer than the symmetrical encryption. By encrypting the shared data with access tree, we ensure that the safety of content privacy of data owner. Besides, the direct revocation mechanism solves the privacy disclosure problems caused by private key mismanagement. Furthermore, the random number $\beta_{RID,k}$ is introduced into the process of private key generating. The components of private key are related to *RID* which is a random sequence, an interactive identity of user. Even the different users collude with each other they can't get the private key that they don't have the permissions. Thus, the illegal user can't search and get the shared data though the collusion.

6.1.2 Identity privacy

The central authority is introduced to the multi-authority CP-ABE scheme, but the central authority in this paper doesn't participate in the process related to attributes. On the one hand, the central authority stores the ciphertext of trapdoor key so that the data owner doesn't need to be always online. On the other hand, CA registers users and randomly generates the user key (UK) and the random identity (RID) for each user. The random sequence RID replaces the user's identity in course of the interaction, which protects the identity privacy of users. Therefore, this mechanism realizes the bidirectional anonymous interaction.

6.1.3 Search privacy

The search mechanism of our scheme can against multiple attacks. By encrypting the hash value of keyword with random number μ maintained only by data owner in the process of index generating, the cloud server can't make the keyword guessing attack internally by matching the candidate keyword with trapdoor. In the stage of trapdoor generating, we hide the search keyword with the random number, which prevents the keyword replay attack executed by malicious attacker after intercepting the trapdoor. Hence, the semi-trusted cloud server and attacker can't obtain any useful information of the keyword and our scheme achieves the privacy preserving for the keyword without reducing security of previous algorithm.

6.1.4 Attribute privacy

Data owner: The fine-grained access control is achieved by the central authority. Users' search privilege is verified by the central authority though the validation (VR), which avoiding the risk brought by submitting access structure to the semi-trusted cloud server. This mechanism protects the attribute of access tree created by the data owner.

Users: Our scheme solves the privacy disclosure problems caused by the collusion of attribute authorities in the multi-authority schemes. The anonymous transfer algorithm is adopted in the interactive process of private key generating as is shown in Fig 4. We assign the attribute to AAs by category, so each attribute authority only manages one kind of attribute. Each user has one value of the attributes controlled by each attribute authority. After receiving private key request of each user, all the attribute authorities compute components of private key for every attribute value. With the anonymous transfer algorithm, attribute authority can't know the components that users choose so that the attributes of users won't be leaked, which protects users' attribute information.

Table 1: The comparison with the classic schemes

scheme	access control	ciphertext search	index security	trapdoor security	access structure security	multi-authority	against AAs' collusion	against users' collusion
[19]	√	—	—	—	×	√	×	√
[15]	√	—	—	—	×	√	√	×
[10]	—	√	√	×	—	—	—	—
[26]	√	√	×	√	×	×	—	×
[21]	√	√	×	×	×	×	—	×
our	√	√	√	√	√	√	√	√

Table 2: The comparison of performance

scheme	Setup	Encrypt	IndexGen	TrapdoorGen	Test	KeyGen	Decrypt
[26]	$O(1)$	$O(X)$	$3E+H+P$	$E+H$	$2E+P$	$O(1)$	$O(X)$
[21]	$O(1)$	$O(XI)$	$2E+H+P$	$2E+P+H$	P	$O(K)$	$O(1)$
our	$O(N^2+1)$	$O(2XI)$	$3E+H+P$	$E+H$	$2E+P$	$O(K)$	$O(X)$

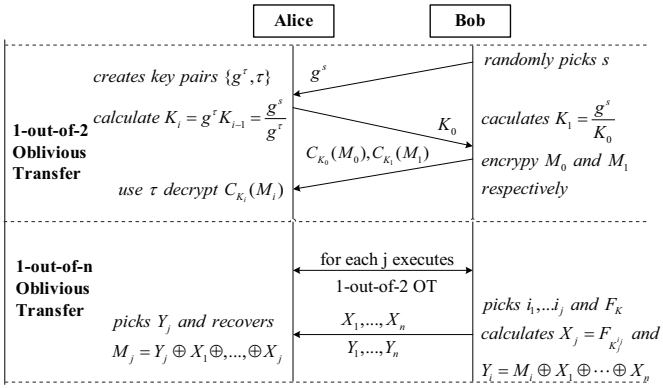

Figure 4: Anonymous transfer algorithm

6.1.5 Scheme Comparison

We compared the proposed scheme with some of classic cloud storage schemes, as is shown in TABLE I. Among the schemes, the literature [15,19] are the single attribute-based encryption cloud storage scheme of which the ciphertext isn't searchable. The shared data is encrypted by symmetric encryption algorithm in [15,21] and the security is insufficient. The scheme in [10] is based on the single public key encryption with keyword search, which lacks of the fine-grained access control. In [21, 26], the multiple encryption algorithms are adopted to achieve access control and ciphertext search function, but the single authority increases the potential vulnerability to the system. Some privacy disclosure problems still exist in the current cloud storage schemes. It can be seen from Table 1 that the scheme proposed in this paper not only realizes the access control and ciphertext search operation of the cloud storage system, but also establishes the relatively perfect privacy preserving mechanism for the hybrid cloud storage system. Our scheme protects the privacy of the keyword in the search process, and solves the collusion problem of the multi-authority mechanism. The mechanism that access control accomplished by the trapdoor key and CA also ensures the security of attribute privacy in the access structure.

6.2 Performance Analysis

In this section, we analyze the performance of the proposed scheme. We denote X as the number of nodes in the access tree and I as the average threshold value. The size of attribute set of users is denoted by K and the number of attribute authority is denoted by N. In the initialization phase of this scheme, the time complexity of the algorithm performed by each attribute authority is $O(1)$. The time complexity of the setup computation is $O(N^2+1)$. There are X nodes in access tree and the average threshold value is I, the complexity of the encryption is $O(2XI)$. In the stage of key generating phase the complexity of N attribute authorities is $O(N^2+N\cdot K)$. Users' private key is composed by K components and the complexity is $O(K)$. Because of the 1-out-of-n transfer algorithm, the communication overhand is increased to $O(K)$. The algorithm of decryption is recursive, which executed at all nodes of access tree, so the complexity is $O(X)$. The computational overhead of search mechanism in this paper is denoted by exponentiation (E), hash function (H) and bilinear pairing (P). In the index generation phase, the overhead is $3E+H+P$ and the cost of trapdoor calculating is $E+H$. Finally, the computational expense of test algorithm is $2E+P$. It can be seen from Table 2 that although we enhance the safety of the system by improving the system structure and algorithm, but the complexity of calculation and communication is increased. In addition, the computing efficiency of search mechanism isn't improved despite the improvement of keyword privacy preserving mechanism in this paper.

7 Conclusion

With the rapid development of Internet technology, cloud storage system centered on data management and sharing has received more and more attention. The cloud storage schemes based on CP-ABE can be used in many files like electronic healthcare [3], Internet of Things [27], and so on. But this model that the data stored by third-party brings the new security risks. The shared content, identity, attributes and other privacy information of users may be disclosed in the use of cloud storage system. Establishing a complete privacy protection mechanism has become an important factor in the development and promotion of cloud storage systems.

Aiming at the privacy disclosure problems caused by submitting the access structure and identities, collusion, and the attacks about keywords, this paper proposes a searchable CP-ABE privacy preserving scheme. The scheme can accomplish the access control and ciphertext search at once and establish a relatively complete privacy protection mechanism for the cloud storage system with hybrid encryption. We introduce the central authority to achieve the access control of users, which protects the attributes in access tree. The problems of collusion and keyword leakage are solved by introducing the anonymous transfer algorithm and improving the original algorithms. The scheme is proved based on the DBDH assumption. Analysis and comparison show that the proposed scheme is more secure and practical.

By analyzing the efficiency of the system, it can be found that that however the privacy protection mechanism adopted in this paper improves the security of the system, but the overhead of computation and communication is still very large. The complexity of the encryption and decryption algorithm increases with the increasing number of attributes and the efficiency of search mechanism needs to be improved. The establishment of safe and efficient Cloud storage system is the key point of our further research. In addition, the revocation in this paper is coarse-grained and it's necessary to achieve the attribute-level user revocation for the hybrid encryption cloud storage scheme in the future.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (No.61462060, No.61562059), Regional Science Foundation Project (No. 61762060), Youth Science and Technology Fund Program of Gansu (No. 1610RJYA008). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 506–522, 2004.

[2] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments", *International Journal of Network Security*, vol. 16, no. 1, pp. 1-13, 2014.

[3] S. Divyashikha, S. Huzur, and G. Daya, "CP-ABE for selective access with scalable revocation: A case study for mobile-based healthfolder," *International Journal of Network Security*, vol. 20, no. 44, pp. 689-701, 2018.

[4] T. Feng and J. Guo, "A new access control system based on CP-ABE in named data networkin," *International Journal of Network Security*, vol. 20, pp. 710–720, 2018.

[5] T. Feng and X. Yin, "Research on privacy preserving mechanism of attribute-based encryption cloud storage," *Chinese Journal of Network and Information Security*, vol. 2, no. 7, pp. 8–17, 2016.

[6] Z. T. Guan, T. T. Yang, R. Z. Xu, and Z. X. Wang, "Multi-authority attribute-based encryption access control model for cloud storage," *Journal on Communications*, vol. 36, no. 6, pp. 116–126, 2015.

[7] H. Hong and Z. Sun, "High efficient key-insulated attribute based encryption scheme without bilinear pairing operations," *SpringerPlus*, vol. 5, no. 1, pp. 1–12, 2016.

[8] W. Hsien, C. C. Yang and M. S. Hwang, "A survey of public auditing for secure data storage in cloud computing," *International Journal of Network Security*, vol. 18, no. 1, pp. 133-142, 2016.

[9] S. T. Hsu, C. C. Yang, and M. S. Hwang, "A study of public key encryption with keyword search", *International Journal of Network Security*, vol. 15, no. 2, pp. 71–79, Mar. 2013.

[10] Q. Huang and H. Li, "An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks," *Information Sciences*, vol. 403, pp. 1–14, 2017.

[11] M. S. Hwang, S. T. Hsu, C. C. Lee, "A new public key encryption with conjunctive field keyword search scheme," *Information Technology and Control*, vol. 43, no. 3, pp. 277–288, Sep. 2014.

[12] M. S. Hwang, C. C. Lee, T. H. Sun, "Data error locations reported by public auditing in cloud storage service," *Automated Software Engineering*, vol. 21, no. 3, pp. 373–390, Sep. 2014.

[13] M. S. Hwang, T. H. Sun, C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits, Systems, and Computers*, vol. 26, no. 5, 2017.

[14] T. Jung, X. Y. Li, Z. Wan, and M. Wan, "Privacy preserving cloud data access with multi-authorities," in *Proceedings IEEE INFOCOM*, pp. 2625–2633, 2013.

[15] T. Jung, X. Y. Li, Z. Wan, and M. Wan, "Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 190–199, 2015.

[16] C. C. Lee, S. T. Hsu, M. S. Hwang, "A study of conjunctive keyword searchable schemes," *International Journal of Network Security*, vol. 15, no. 5, pp. 311–320, 2013.

[17] J. Li, Y. Shi, and Y. Zhang, "Searchable ciphertext-policy attribute-based encryption with revocation in cloud storage," *International Journal of Communication Systems*, vol. 30, no. 1, 2017.

[18] J. W. Li, C. F. Jia, Z. L. Liu, J. Li, and M. Li, "Survey on the searchable encryption," *Journal of Software*, vol. 26, no. 1, pp. 109–128, 2015.

[19] Q. Li, J. Ma, R. Li, X. Liu, J. Xiong, and D. Chen, "Secure, efficient and revocable multi-authority access control system in cloud storage," *Computers & Security*, vol. 59, pp. 45–59, 2016.

[20] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for shared data storage with user revocation in cloud computing", *International Journal of Network Security*, vol. 18, no. 4, pp. 650–666, 2016.

[21] Q. Wang, Y. Zhu, and X. Luo, "Multi-user searchable encryption with coarser-grained access control without key sharing," in *International Conference on Cloud Computing and Big Data (CCBD'14)*, pp. 119–125, 2014.

[22] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, and W. Xie, "An efficient file hierarchy attribute-based encryption scheme in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1265–1277, 2016.

[23] W. Wang, P. Xu, H. Li, and L. T. Yang, "Secure hybrid-indexed search for high efficiency over keyword searchable ciphertexts," *Future Generation Computer Systems*, vol. 55, pp. 353–361, 2016.

[24] Z. Wang, Y. Lu, G. Sun, "A policy-based deduplication mechanism for securing cloud storage," *International Journal of Electronics and Information Engineering*, vol. 2, no. 2, pp. 70–79, 2015.

[25] P. Xu, H. Jin, Q. Wu, and W. Wang, "Public-key encryption with fuzzy keyword search: A provably secure scheme under keyword guessing attack," *IEEE Transactions on Computers*, vol. 62, no. 11, pp. 2266–2277, 2013.

[26] Y. Yang and B. G. Lin, "Secure hidden keyword searchable encryption scheme with fine-grained and flexible access control," *Journal on Communications*, vol. 34, no. Z1, pp. 92–100, 2017.

[27] X. Yao, Z. Chen, and Y. Tian, "A lightweight attribute-based encryption scheme for the internet of things," *Future Generation Computer Systems*, vol. 49, pp. 104–112, 2015.

[28] Z. b. Ying, J. F. Ma, and J. T. Cui, "Partially policy hidden CP-ABE supporting dynamic policy updating," *Journal on Communications*, vol. 36, no. 12, pp. 178–189, 2015.

[29] R. Zhang, H. Ma, and Y. Lu, "Fine-grained access control system based on fully outsourced attribute-based encryption," *Journal of Systems and Software*, vol. 125, pp. 344–353, 2017.

Biography

Tao Feng is researcher and doctoral supervisor, CCF senior member, IEEE and ACM member. He received the Ph.D. degrees in Xidian University and then worked at school of computer and communication in Lanzhou University of Technology. His research interests include Cryptography and information security.

Xiaoyu Yin is a graduate student at School of Computer and Communication, Lanzhou University of Technology. Her research interest is Network and information security.

Ye Lu is a doctoral student at college of Electrical and Information Engineering, LanZhou University of Technology. His research interests include information security and industrial control system.

Junli Fang received her Master's degree in Communication and Information System from Beijing JiaoTong University, Beijing, China in 2009. She is a lecturer in the School of Computer and Communication, Lanzhou University of Technology, China. Her research interests include network and information security.

Fenghua Li is a PhD supervisor worked in The State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences. His current research interests include Network Security, System Security & Evaluation and Trusted Computation.