An Efficient and Revocable Decentralizing Attribute-Based Encryption for Mobile Cloud Computing

Lei Sun, Yumeng Fu(✉), and Zuohui Li

Zhengzhou Information Science and Technology Institute,
Zhengzhou 450001, China
fym25jessica@qq.com

Abstract. Mobile cloud computing (MCC) is becoming an important way of data sharing. It is of great value for people to store and retrieve personal data at anytime and anywhere. Attribute-based encryption (ABE) can solve the problem of flexible sharing among multiple users in traditional encryption, but it cannot apply to mobile clients directly because of its low efficiency. How to meet the people's practical application needs and to control and manage the personal data safely and flexibly is a concrete embodiment of the security problem after the cloud computing is mobile. In this paper, an efficient and revocable decentralizing attribute-based encryption scheme for mobile cloud environment is proposed. In the scheme, it does not have the upper limit of the total attribute, and without the central authority (CA), each attribute authority generates private key independently with the users. In addition, the linear secret-sharing scheme (LSSS) is used to construct the access structure with a high flexibility. The method of precomputing and outsourcing can reduce the computation cost of the user side. Besides, the scheme is proved to be static secure and support revocation under the random oracle model. Compared to the existing related schemes, the proposed scheme is more practical and effective in mobile cloud environment.

Keywords: Mobile cloud computing · Attribute-based encryption
Outsourcing · Revocation · Decentralizing

1 Introduction

For some sensitive data sharing issues (including their personal files, health records, e-mails, etc.) that users store in the cloud, since cloud server loyalty cannot be ensured, access control is generally stored and implemented in encrypted form. Traditional cryptographic algorithms are difficult to implement flexible sharing with multiple users. The attribute-based encryption (ABE) system can solve this problem better. It is an important technical means for data security sharing in cloud computing.

With the popularity of mobile devices and the rapid development of 5G networks, the mobile cloud computing model has been recognized and appreciated by more and more people. Mobile Cloud Computing (MCC) is the combination of cloud computing, mobile computing and wireless networks to bring rich computational resources to

H. Zhang et al. (Eds.): CTCIS 2018, CCIS 960, pp. 167–183, 2019.
https://doi.org/10.1007/978-981-13-5913-2_11

mobile users, net-work operators, as well as cloud computing providers [1]. The combination of mobile network and cloud computing gives people the freedom to the greatest extent. It enables cloud computing to be greatly expanded in application scope. Users can get rid of the limitations of time and space and enjoy cloud-based powerful computing, storage, and software service capabilities more conveniently. At present, well-known domestic and foreign enterprises have successively provided mobile cloud computing services, such as Ali "Apsara Mobile" weighs in 2017, Apple's "MobileMe" service, and Microsoft "Live Mesh".

The typical ABE scheme cannot apply to the actual application of mobile cloud directly, mainly because of efficiency. Most of the proposed schemes are based on the bilinear mapping. The overhead of large bilinear pairings and group power operations is very high, and the amount of computing in the encryption and decryption phases is proportional to the size of the attribute set or the size of the access structure. Secondly, in the actual application scenario, different authorities manage the different attributes of the user, so the ABE scheme expanded from a single authority to a multi-authority has higher computational complexity and management complexity. In addition, in order to improve the security of the scheme, scholars have improved the scheme that can achieve selective security by using the composite order bilinear group and the dual system encryption technology. The improved scheme proves that the adaptive security is satisfied, but the same group operation in the new method is 1–2 orders of magnitude slower than the prime group in the number order group, and it does not meet the actual security requirements. Therefore, research on flexible and practical attribute-based encryption schemes based on the improvement of security performance has important application value for data security sharing in mobile cloud environment.

In this paper, a decentralized multi-authority architecture is adopted. To make use of the semi-trusted features of the cloud server, we divide the encryption and decryption into two phases. The cloud server participates in the partial encryption, decryption, and revocation, and reduces the computational overhead of the user. We use the mapping idea to map attributes and group elements in a one-to-one mapping. Any string can be added as an attribute into the system, which is more in line with the actual application scenario. Finally, it is proved that the proposed scheme is static secure under the random oracle model and is suitable for data security sharing in mobile cloud environment.

2 Related Work

In 2005, Sahai and Waters [5] expanded the concept of IBE, and they put forward the concept of attribute encryption for the first time. Since then, Goyal et al. [6] defines two forms of ABE based on the location of the access strategy in the scheme: key policy attribute-based encryption (KP-ABE) and ciphertext policy attribute-based encryption (CP-ABE). We put forward a CP-ABE scheme that is more suitable for solving the problem of data security sharing in cloud environment, in which access policies can be defined simultaneously when data owners encrypt plaintext.

Cheung et al. [7] proposed a CP-ABE scheme based on the AND access structure. Then, Goyal et al. [8] proposed a CP-ABE scheme based on the access tree structure. Waters [9] has published a CP-ABE scheme based on the linear secret sharing schemes (LSSS) access structure to achieve stronger expressive and higher efficiency. But the essence of the ABE efficiency problem is that when using bilinear mapping technology to build a scheme, the complex function is realized by some modular exponentiation or bilinear pairing, so it is difficult to achieve a breakthrough in the calculation efficiency in the construction of the scheme. Therefore, scholars naturally think of outsourcing complex computing and use powerful computing power to solve the problem. Early research on outsourcing computing is mainly around a single type of computing, until 2011, Green et al. [10] introduced the outsourcing idea into the ABE, greatly alleviated the decryption burden of the end users, and also provided a new possibility for the development of the attribute-based encryption in the mobile cloud computing.

Since the mobile cloud computing is proposed in 2010, the problem of attribute-based data security sharing has been paid much attention by scholars, but the attribute-based encryption schemes for mobile cloud environment are still very few. Until 2014, Hohenberger et al. [11] introduced a method of reducing the computational complexity of encryption operations in ABE schemes by precomputing and proposed an online/offline ABE scheme. It can be used to solve the problem of data sharing in the cloud computing environment due to the above method that reduce the computing overhead of users online effectively, so it is especially suitable for the setting of user low end equipment in mobile cloud.

In order to solve the problem that the property can not only be managed by a single authority in the actual environment, the scholars have deeply studied the multi-authority attribute-based encryption (MA-ABE) scheme, which mainly focuses on how to reduce the security threats brought by the corruption of the central authority and how to ensure that the authorities do not affect the independent operation of each other. In 2011, Lewko and Waters et al. [12] proposed a MA-ABE scheme that did not need a CA and each authority even not knowing each other and implemented a decentralizing attribute-based encryption (DABE). In 2015, Vijay et al. [13] proposed a CP-ABE scheme in the mobile cloud environment with attribute revocation. This scheme takes into account that the ABE scheme of a single authority cannot solve the different attribute issues of the multi-authority management user in the real application scenario and supports multiple attribute authority to work simultaneously. In 2016, Li et al. [14] proposed a lightweight data sharing scheme in the mobile cloud environment. By changing the access control tree structure, most of the computation was handed over to the external proxy server and the revocation function was realized. In 2017, Lyu et al. [15] adopted an anonymous key release protocol to achieve privacy protection, in addition to online/offline technology and outsourced decryption. Zhao et al. [16] proposed a mobile cloud CP-ABE scheme that verifiable outsourced computing, verified by two kinds of hash functions, and De [17] proposed a scheme for fast encryption and decryption in mobile cloud environment, which could realize the decentralization of attributes. Li et al. [18] uses dual factor identity authentication mechanism to realize anonymous authentication to users and proposes a multi authority CP-ABE scheme without CA in mobile cloud environment.

In terms of security, in 2010, Lewko et al. [19] improved the CP-ABE structure by using multiple order bilinear groups and dual system encryption technology to make the scheme change from the security based on the selection attribute set attack model to the higher security level of adaptive security from the same period, but the performance of the scheme was greatly sacrificed. In 2015, the Rouselakis of the Waters team proposed a static security [20] model to adapt to the setting of multi authority institutions, which would be more consistent with data sharing in the multi authority scenario, and a considerable degree of recognition in the industry.

In summarizing the above work, the paper finds that the design (composite order and prime order) of the adaptive security solution needs to sacrifice the performance of the solution to a certain extent, and it is not suitable for data sharing under the mobile cloud environment. Since the attributes of the user in the CP-ABE are associated with the decryption key, and the attributes are distributed in a fragmented manner, each attribute should correspond to a different permission (in a single organization scenario, the private key request is independent and complete). In this paper, the static security model proposed by Rouselakis and Waters is applied to the setting of multi-authority. At the same time, for the computational complexity caused by the multi-authority setting, this paper uses the pre-computing of the encryption phase and partial outsourcing in the decryption phase, making the scheme more suitable for mobile devices, so that users can get better application experience.

3 Preliminaries

This section will introduce the relevant basic knowledge and data sharing model of the proposed mobile cloud environment attribute-based encryption scheme and give the complexity assumption and the security model based on the scheme.

A. Linear Secret Sharing Schemes (LSSS)
Let p be a prime and U the attribute universe.

A secret-sharing scheme Π with domain of secrets $\mathbb{Z}_p$ realizing access structures on U is linear over $\mathbb{Z}_p$ if

(1) The shares for each party form a vector over $\mathbb{Z}_p$
(2) There exists a matrix A with rows ℓ and n columns called the share-generating matrix for Π. For all $i = 1, \ldots, \ell$, the row of A is labeled by a party $\rho(i)$ ($\rho(i)$ represents the participants marked by the A line i). When we consider the column vector $\boldsymbol{v} = (s, y_2, \ldots, y_n) \in \mathbb{Z}_p$, where $s \in \mathbb{Z}_p$ is the secret to be shared, and $y_2, \ldots, y_n \in \mathbb{Z}_p$ are randomly chosen, then $A\boldsymbol{v}$ is the vector of ℓ shares of the secret s according to Π. The i^{th} share $(A\boldsymbol{v})_i$ belongs to party $\rho(i)$.

It is shown in [21] that every linear secret-sharing scheme according to the above definition also enjoys the linear reconstruction property, defined as follows. Suppose that Π is an LSSS for the access structure. Let $S \in A$ be any authorized set and let $I \subseteq \{1, 2, \ldots, l\}$ be defined as $I = \{i : \rho(i) \in S\}$. Then there exist constants $\{\omega_i \in \mathbb{Z}_p\}_{i \in I}$ such that, if $\{\lambda_i\}$ are valid shares of any secret s according to Π, then $\sum_{i \in I} \omega_i \lambda_i = s$. There is no such constant for unauthorized sets.

B. Complexity Assumption

For our security proof we will use a q-type assumption on prime order bilinear groups. It is a slightly modified version of the q-Decisional Parallel Bilinear Diffie-Hellman Exponent Assumption [20]. We will be referring to our assumption as q-DPBDHE2 for short. The assumption is defined as follows:

Choose a bilinear group G of order p according to the security parameter κ, which admits a non-degenerate bilinear mapping $e : G \times G \to G_T$. Pick $a, s, b_1, b_2, \ldots, b_q \in \mathbb{Z}_p^*$. Let

$$D = \left(\begin{array}{l} p, g, G, e, g^s, \{g^{a^i}\}_{i\in[2q], i\neq q+1}, \{g^{b_j a^i}\}_{(i,j)\in[2q,q], i\neq q+1}, \\ \{g^{s/b_i}\}_{i\in[q]}, \{g^{sa^i b_j}/b_{j'}\}_{(i,j,j')\in[q+1,q,q], j\neq j'} \end{array} \right) \tag{1}$$

The assumption states that no polynomial-time distinguisher can distinguish the distribution $e(g,g)^{sa^{q+1}}$ from the distribution R (R is randomly chosen from G)with more than negligible advantage.

C. Security Model

(1) The static security model we define is a security game between the challenger and the attacker, and the difference between the adaptive model is that the attacker must specify the attack object and the query content immediately after receiving the public parameters, then send it to the challenger and change it after the game is over. In the same way as the adaptive model, the static security model allows an attacker to ask the user's private key and some decryption ciphertext stored in the cloud many times, that is to say, the attacker can decrypt the ciphertext by asking the outsourced decryption key to get some decryption ciphertext. In addition, we allow an attacker to participate in encryption by generating some authoritative authority to generate authoritative public key. On the basis of resisting the cloud server attack, the model increases the conspiracy attack against multiple legitimate users through the inquiry of the private key and can be described by the Games in the following stages. The symbolic correspondence is shown in Table 1.

Global Setup: The global initialization algorithm in the challenger's operation plan and send the public parameters GP to the attacker.

Attacker's Queries: The attacker first came from the authority V select a part of the authority $C(C\subseteq V)$, then generate and send $\{\mathrm{PK}_\beta\}_{\beta\in C}$ to the challenger, Then the attacker responds with:

- Select authorized m users $\{\mathrm{GID}_i\}_{i=1}^m$ to inquire about their public and private key.
- Select some non-corrupt authorities $N(N\subseteq V)$ to ask their public key.
- Select n users $\{S_i, \mathrm{GID}_i\}_{i=1}^n$ to ask its outsourcing decryption key. $S_i\subseteq U$ is the attribute set of user i. Required $T(S_i)\cap C = \varnothing$ means the attributes owned by the users are authorized by the uncorrupted authority. In addition, requiring $n > m$ is that the attacker can not only query the user's outsourcing decryption key in the m authorized users, but also ask other users for the corresponding outsourced decryption key.

Table 1. Symbol corresponding list

Symbol	Meaning
GP	The global parameter
U/V	Space of attribute/authority
S	Attribute set of user
GID_i	The i^{th} user global identities
$\mathrm{PK}_\beta/\mathrm{SK}_\beta$	AA_β's public/private key
$\mathrm{userPK}_i/\mathrm{key}$	User's public/private key
$\mathrm{SK}_{\mathrm{Out}}$	CSP's private key
TK	The temporary key
IC	Intermediate ciphertext
Klist	CSP's key list
CT	Ciphertext

– Two messages m_0, m_1 of equal length, and a challenge access structure (A, ρ) encoded in a suitable form. We require that for every $i(1 \leq i \leq n)$ the set $S_i \cup S_C$ is an unauthorized set of the access structure (A, ρ).

Challenger's Replies: The challenger flips a random coin $b \in \{0, 1\}$ and replies with:

– The secret keys of users $\{\mathrm{GID}_i\}_{i=1}^m$: $\{\mathrm{userPK}_{\mathrm{GID}_i}, \mathrm{key}\}_{i=1}^m$
– The public keys of authorities $N \subseteq V$: $\{\mathrm{PK}_\beta\}_{\beta \in N}$
– The secret keys of cloud serve provider: $\{\mathrm{SK}_{\mathrm{Out}}\}_{i=1}^n$
– The challenge ciphertext CT*

Guess: The attacker outputs a guess $b' \in \{0, 1\}$.

Definition 1. We say that an attacker statically breaks the scheme if it has a non-negligible advantage in correctly guessing the bit b in the above security game.

When the game does not contain a class queries, the security model is converted to attack only against cloud servers.

(2) We propose a revocable security model for collusion attacks of multiple revocation users, regardless of whether their attributes satisfy the access policy. Therefore, an opponent can ask multiple private keys to revoke the user. In this article, it is assumed that the revocation user cannot conspire with the cloud server and the authority, so the enemy cannot ask for the private key of the cloud server corresponding to the cancellation of the user, nor the public key of the authority. At the same time, in order to ensure that the enemy can obtain partial decryption ciphertext from the unrevoked user, the model allows the enemy to access the private key of the cloud server that corresponds to the unrevoked user. The game description between the adversary and the challenger is basically the same as the static security model. During the enquiry phase, the opponent inquired the challenger as follows:

- Select a part of the revocation of the users $\{\mathrm{GID}_i\}_{i=1}^{m}$ to inquire about their public and private key.
- Select some non-corrupt authorities $N(N \subseteq V)$ to ask their public key.
- Select n users $\{S_i, \mathrm{GID}_i\}_{i=m}^{n}$ to ask its outsourcing decryption key.
- Two messages m_0, m_1 of equal length, and a challenge access structure (A, ρ) encoded in a suitable form. Ask them the challenge ciphertext.

Definition 2. We say that the scheme is revocable if the attacker has a non-negligible advantage in correctly guessing the bit b in the above security game.

D. Data Sharing Model

The data sharing model in the mobile cloud environment contains four entities: the data owner (DO), the cloud service provider (CSP), the data user i.e. mobile user (DU), and attribute authority (AA). Among them, the data owner DO draws up the access structure according to the security policy, then encrypts the data according to the access structure, and then uploads the encrypted result, that is, the ciphertext associated with the access policy to the cloud end; any user can freely access and obtain the ciphertext files on the cloud service, when and only as the user DU The attributes that are available can only be decrypted when they satisfy the access policy of ciphertext. The attribute authority AA generates a private key for the cloud service provider according to the user's permissions, and the user DU generates the private key for himself. Only when the cloud server decrypts the original ciphertext with its own private key, can the user decrypt the ciphertext. When the user is revoked, the cloud server simply removes the user's corresponding private key of the cloud server. The shared framework is shown in Fig. 1.

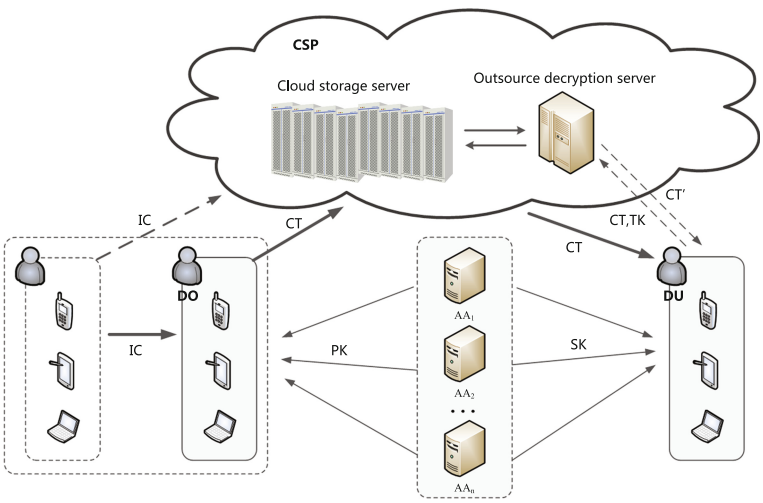

Fig. 1. Data sharing model in MCC

Cloud service provider CSP is generally considered honest and curious, with powerful storage and computing power, but only used for storing ciphertext, partially deciphering, and assisting revocation, and cannot get any information about the data or key from it; DU does not need to over consider its own hardware and software conditions and can be configured by different configurations. The device can access the CSP to obtain its authorized resources.

The DO and DU here mainly refer to users of low-end devices, such as mobile phones, vehicle systems, and high-end devices such as PC.

4 Our Scheme

Taking into account the actual application requirements and cloud data sharing mechanism research, we use functions T mapping attributes $i \in U$ to i's authority $\beta \in V$. That is, the existence of a full shot δ can correspond to the row x of a matrix and an authority β $(\delta(x) = T(\rho(x)) \rightarrow \beta)$. In addition, we introduce pre-computing outsourcing operation before the scheme encryption, and divide the mobile cloud attribute-based security sharing into four aspects, namely, initialization, user registration, data encryption and data access. Our scheme is constructed as follows:

A. Initialization

GlobalSetup$(\lambda) \rightarrow$ GP. The global setup algorithm takes as input the security parameter λ, chooses a suitable bilinear group G of prime order p with generator g. It also chooses a function H mapping GID, GID $\in G$. Another function F mapping strings, interpreted as attributes, to elements of G. Both of these functions will be modeled as random oracles in the security proof. Finally, it defines U, V, and T. The global parameters are GP $= \{p, G, g, H, F, U, V, T\}$. GP as the common input parameter of the remaining seven algorithms, for the sake of conciseness, the following algorithms will no longer be mentioned.

AuthoritySetup$(\mathrm{GID}_\beta) \rightarrow \{\mathrm{PK}_\beta, \mathrm{SK}_\beta\}$ The authority run setup algorithm independently, input $\mathrm{GID}_\beta(\beta \in V)$ chooses two random exponents $\alpha_\beta, y_\beta \in \mathbb{Z}_p^*$ and publishes $\mathrm{PK}_\beta = \{e(g,g)^{\alpha_\beta}, g^{y_\beta}\}$ as its public key. It keeps $\mathrm{SK}_\beta = \{\alpha_\beta, y_\beta\}$ as its secret key.

B. User Registration

When new users access the system, users need to request private keys to the attribute authority. The private key is generated by the execution $\mathrm{KeyGen}_{\mathrm{Out}}$ algorithm by the authority corresponding to each attribute of the user attribute set S.

$\mathrm{KeyGen}_{\mathrm{user}}(\mathrm{GID}_i) \rightarrow \{\mathrm{userPK}_i, \mathrm{key}\}$ First, this algorithm is run on the mobile device, that is, the user part of the key generation algorithm in the classic scheme, which is completed by the user, input $\mathrm{GID}_i(i \in U)$, and output the public and private key pairs of the user $\{\mathrm{userPK}_i, \mathrm{key}\}$. Chooses a random $z \in \mathbb{Z}_p^*$, using its own GID_i

computing the user's public key $\text{userPK}_{\text{GID}_i} = \{g^z, H(\text{GID}_i)^z\}$ and publish it. The relevant authority operates the outsourcing decryption server key generation algorithm for cloud service provider CSP:

$$\text{KeyGen}_{\text{Out}}(\text{GID}_i, \text{userPK}_i, S, \{\text{SK}_j\}) \rightarrow \text{SK}_{\text{Out}}$$

For example, user GID_i input userPK_i, S and the private key set $\{\text{SK}_\beta\}$ of the relevant authority (i.e. $\forall i \in S, T(i) = \beta$), output the decryption key for the user GID_i, and add the attribute based data sharing system.

For $\forall i \in S$, if $T(i) = \beta$, Then the authorized β mechanism chooses the random element $t_i \in \mathbb{Z}_p^*$ and calculation: $K_{i,1} = g^{z\alpha_i} H(\text{GID})^{zy_i} F(i)^{t_i}$, $K_{i,2} = g^{t_i}$, outputs the secret key: $\text{SK}_S = \{S, K_{i,1}, K_{i,2}\}_{i \in S}$. The user secrecy the private key key $= 1/z$ and calculate: $K'_{i,1} = K_{i,1}^{1/z}$, $K'_{i,2} = K_{i,2}^{1/z}$, output cloud server decryption key: $\text{SK}_{\text{Out}} = \{S, K'_{i,1}, K'_{i,2}\}_{i \in S}$, added $\{\text{GID}, \text{SK}_{\text{Out}}\}$ to the cloud server key list Klist.

C. Data Encryption

When the mobile device is idle, run the algorithm $\text{Pre}_{\text{Enc}}(\{\text{PK}_\beta\}) \rightarrow \text{IC}$, enter a public key set $\{\text{PK}_\beta\}$ of authoritative authorities based on user-defined access policies. Output the intermediate ciphertext IC, it can be uploaded to the cloud storage server. Mainly in the formal encryption before, for each attribute i in U, complete pre-calculation at first, for the encryption provides the calculation results. The attribute i, random selection $\lambda'_i, \omega'_i, r_i \in \mathbb{Z}_p^*$, and calculation:$\text{IC}_{i,1} = \text{e}(g,g)^{\lambda'_i} e(g,g)^{\alpha_i r_i}$, $\text{IC}_{i,2} = g^{-r_i}$, $\text{IC}_{i,3} = g^{y_i r_i} g^{\omega'_i}$, $\text{IC}_{i,4} = F(i)^{r_i}$. The encrypted person can choose to upload the middle ciphertext $\text{IC} = \{\text{IC}_{i,1}, \text{IC}_{i,2}, \text{IC}_{i,3}, \text{IC}_{i,4}\}_{i \in U}$ to the CSP outsourced storage server to save the storage resources of the device.

The temporary key $\text{TK} = \{\lambda'_i, \omega'_i\}_{i \in U}$ is stored locally.

When mobile users need to share secret data, run $\text{Encrypt}(\text{IC}, M, (A, \rho)) \rightarrow \text{CT}$. This algorithm can also skip precomputation and encrypt plaintext directly. Input messages, access policies, intermediate ciphertext and temporary keys in turn. Then random selection $s, y_2, \ldots, y_n, z_2, \ldots, z_n \in \mathbb{Z}_p^*$, order vector $\boldsymbol{v} = (s, y_2, \ldots, y_n)^T$, $\boldsymbol{w} = (0, z_2, \ldots, z_n)^T$, for all $x \in [\ell]$ calculations $\lambda_x = (A\boldsymbol{v})_x$, $\omega_x = (A\boldsymbol{w})_x$. Due to $\delta(x) = T(\rho(x)) \rightarrow \beta$, it can be mapped $x \in [\ell]$ to authority β.

Computing ciphertext: $C_0 = Me(g,g)^s$, among them $C_{x,j} = \text{IC}_{\rho(x),j} \Big| j \in \{\mathbb{Z}_p^* | 1 \le j \le 4\}$, $C_{x,5} = \lambda_x - \lambda'_{\rho(x)}$, $C_{x,6} = \omega_x - \omega'_{\rho(x)}$, Output ciphertext: $\text{CT} = ((A, \rho), C_0, \{C_{x,j}\}_{x \in [\ell], j \in \{\mathbb{Z}_p^* | 1 \le j \le 6\}})$. The above operations can also be precomputed by the data owner when the official data is encrypted, and then the encryption is completed. This design draws on the idea of online/offline, making full use of the idle time and cloud storage capacity of the user side, providing some calculation results for the formal encryption phase and alleviating the encryption pressure to a certain extent.

D. Data Access

DU downloads ciphertext from CSP. If the ciphertext is legal, the mobile terminal uses the private key to complete the decryption.

When the cloud server receives the access request, it first depends $\text{userPK}_{\text{GID}}$ on the terminal. Find the corresponding cloud server decryption key $\text{SK}_{\text{Out}} = \{S, K'_{i,1}, K'_{i,2}\}_{i \in S}$ in the cloud server key list Klist.Then run $\text{Out}_{\text{Dec}}(\text{SK}_{\text{Out}}, \text{userPK}_i, \text{CT}) \rightarrow \text{CT}'$ to partial decryption. When the end user's associated attribute set S in the key of the outsourcing decryption server does not satisfy the access policy (A, ρ) in the ciphertext, decryption fails. Otherwise, for $I = \{x : \rho(x) \in S\} \subseteq \{1, 2, \ldots, \ell\}$, the decryption server compute $\{c_x \in \mathbb{Z}_p\}$ to satisfied $\sum_{x \in I} c_x A_x = (1, 0, \ldots, 0)$, and finally sent the partially decrypted ciphertext $\text{CT}' = (C_0, C_{\text{part1}}, C_{\text{part2}})$ to DU. Among them:

$$C_{part1} = \prod_{x \in I} \{C_{x,1} \cdot e(g,g)^{C_{x,5}} e(K'_{\delta(x),1}, C_{x,2}) e(K'_{\delta(x),2}, C_{x,4})\}^{c_x} \tag{2}$$

$$C_{\text{part2}} = \prod_{x \in I} \{e(H(\text{GID})^z, C_{x,3} \cdot g^{C_{x,6}})\}^{c_x} \tag{3}$$

After receiving the encrypted ciphertext from the cloud server part, the end user runs the algorithm Decrypt(), uses the reserved user's private key $\text{key} = 1/z$ to complete the remaining decryption operation, calculates $C_{\text{part1}} \cdot C_{\text{part2}}^{1/z} = e(g,g)^s$ and finally returns: $M = \frac{C_0}{C_{\text{part1}} \cdot C_{\text{part2}}^{1/z}}$.

Revoke(GID,KT) The user revoked. Enter the user identity and key list, find and delete the array in the list $\{\text{GID},\text{SK}_{\text{Out}}\}$, update the list $\text{KT} = \text{KT} \backslash \{\text{GID},\text{SK}_{\text{Out}}\}$.

5 Analysis

A. Correctness

(1) Outsourcing decryption process: when the attribute set S satisfies the access policy (A, ρ), $I = \{x : \rho(x) \in S\} \subseteq \{1, 2, \ldots, \ell\}$, there is the constant $\{c_x \in \mathbb{Z}_p\}$ satisfies $\sum_{x \in I} \lambda_x c_x = s$ and $\sum_{x \in I} \omega_x c_x = 0$. The following results are as follows:

$$
\begin{aligned}
&C_{\text{part1}} \\
&= \prod_{x\in I}\{C_{x,1}\cdot e(g,g)^{C_{x,5}} e(K'_{\delta(x),1}, C_{x,2}) e(K'_{\delta(x),2}, C_{x,4})\}^{c_x} \\
&= \prod_{x\in I}\{\mathrm{e}(g,g)^{\lambda'_{\rho(x)}}\mathrm{e}(g,g)^{\alpha_{\rho(x)}r_{\rho(x)}} e(g,g)^{(\lambda_x-\lambda'_{\rho(x)})} e((g^{z\alpha_{\rho(x)}} H(\mathrm{GID})^{zy_{\rho(x)}} F(\rho(x))^{t_{\rho(x)}})^{1/z}, \\
&\quad g^{-r_{\rho(x)}}) e((g^{t_{\rho(x)}})^{1/z}, F(\rho(x))^{r_{\rho(x)}})\}^{c_x} \\
&= \prod_{x\in I}\{e(g,g)^{\alpha_{\rho(x)}r_{\rho(x)}} e(g,g)^{\lambda_x} e((g,g)^{-\alpha_{\rho(x)}r_{\rho(x)}}\mathrm{e}(H(\mathrm{GID}), \\
&\quad g)^{-y_{\rho(x)}r_{\rho(x)}} e(F(\rho(x)), g)^{-r_{\rho(x)}t_{\rho(x)}/z} e((g, F(\rho(x)))^{r_{\rho(x)}t_{\rho(x)}/z}\}^{c_x} \\
&= \prod_{x\in I}\{e(g,g)^{\lambda_x}\mathrm{e}(H(\mathrm{GID}), g)^{-y_{\rho(x)}r_{\rho(x)}}\}^{c_x} \\
&= e(g,g)^{\sum_{x\in I}\lambda_x c_x} e(H(\mathrm{GID}), g)^{-\sum_{x\in I} y_{\rho(x)} r_{\rho(x)} c_x} \\
&= e(g,g)^{s} e(H(\mathrm{GID}), g)^{-\sum_{x\in I} y_{\rho(x)} r_{\rho(x)} c_x}
\end{aligned}
\tag{4}
$$

$$
\begin{aligned}
&C_{part2} \\
&= \prod_{x\in I}\{e(H(\mathrm{GID})^{z}, C_{x,3}\cdot g^{C_{x,6}})\}^{c_x} \\
&= \prod_{x\in I}\{e(H(\mathrm{GID})^{z}, g^{y_{\rho(x)}r_{\rho(x)}} g^{\omega'_{\rho(x)}}\cdot g^{(\omega_x-\omega'_{\rho(x)})})\}^{c_x} \\
&= \prod_{x\in I}\{e(H(\mathrm{GID})^{z}, g^{y_{\rho(x)}r_{\rho(x)}} g^{\omega_x})\}^{c_x} \\
&= \prod_{x\in I}\{e(H(\mathrm{GID}), g)^{z\cdot y_{\rho(x)}r_{\rho(x)}} e(H(\mathrm{GID}), g)^{z\cdot\omega_x}\}^{c_x} \\
&= e(H(\mathrm{GID}), g)^{z\sum_{x\in I} y_{\rho(x)}r_{\rho(x)}c_x} e(H(\mathrm{GID}), g)^{z\sum_{x\in I}\omega_x c_x} \\
&= e(H(\mathrm{GID}), g)^{z\sum_{x\in I} y_{\rho(x)}r_{\rho(x)}c_x}
\end{aligned}
\tag{5}
$$

(2) The mobile device completes the final decryption:

$$
\begin{aligned}
&C_{part1}\cdot C_{part2}^{1/z} \\
&= \{e(g,g)^{s} e(H(\mathrm{GID}), g)^{-\sum_{x\in I} y_{\rho(x)}r_{\rho(x)}c_x}\}(e(H(\mathrm{GID}), g)^{z\sum_{x\in I} y_{\rho(x)}r_{\rho(x)}c_x})^{1/z} \\
&= e(g,g)^{s}\frac{C_0}{C_{part1}\cdot C_{part2}^{1/z}} = \frac{Me(g,g)^{s}}{e(g,g)^{s}} = M
\end{aligned}
\tag{6}
$$

B. Security

(1) Static Security

Lemma 1. assumes that the Rouselakis-Waters (RW) scheme [20] is static security, then the decentralizing multi authority CP-ABE scheme for the mobile cloud computing is also static security.

Proof. Assuming that the attacker cannot ignore the advantage of the probability polynomial time breaking this scheme, it is proved that we can construct a probability polynomial time algorithm Φ to break the RW scheme,

Φ runs algorithm of Global Setup: output and send the public parameters GP to the attacker.

Attacker's Queries: The attacker first came from the authority V select a part of the agency $C(C \subseteq V)$, then generate and send its public key $\{\mathrm{PK}_\beta\}_{\beta \in C}$ to the Φ, Then ask Φ as follows:

- Select m authorized users $\{\mathrm{GID}_i\}_{i=1}^m$ to inquire about their public and private key.
- Select some non-corrupt authorities $N(N \subseteq V)$ to ask their public key.
- Select n users $\{S_i, \mathrm{GID}_i\}_{i=1}^n$ to ask its outsourcing decryption key. $S_i \subseteq U$ is the attribute set of the user i. Required $T(S_i) \cap C = \varnothing$ means the attributes owned by the users are authorized by the uncorrupted authority. In addition, the requirement $n > m$ is that the attacker can not only query the user's outsourcing decryption key in the m authorized users, but also ask other users for the corresponding outsourced decryption key.
- Two messages m_0, m_1 of equal length, and a challenge access structure (A, ρ) encoded in a suitable form. We require that for every $i(1 \le i \le n)$ the set $S_i \cup S_C$ is an unauthorized set of the access structure (A, ρ).

Challenger's Replies: Φ send $\{\mathrm{PK}_\beta\}_{\beta \in C}$ to the challenger and inquire about the corresponding public key of $N \subseteq V$ in the RW scheme, also do the corresponding private key and the challenge ciphertext of $\{S_i, \mathrm{GID}_i\}_{i=1}^m$. The challenger turns back $\{\mathrm{SK}_{S_i,\mathrm{GID}_i} = (g^{\alpha_\beta} H(\mathrm{GID}_i)^{y_\beta} F(j)^{t_j}, g^{t_j})_{j \in S_i}\}_{i=1}^m$, The public keys $\{\mathrm{PK}_\beta\}_{\beta \in N}$ of authorities $N \subseteq V$ and the challenge ciphertext CT*. First, Φ calculate the user's private key in this scheme: for $1 \le i \le m$, chooses the random element $z \in \mathbb{Z}_p^*$, calculate $\mathrm{userPK}_{\mathrm{GID}_i} = \{g^z, H(\mathrm{GID}_i)^z\}$ and $\mathrm{key}_{\mathrm{GID}_i} = \{1/z\}_i$, then discuss the corresponding outsourced decryption key for $\{S_i, \mathrm{GID}_i\}_{i=1}^n$, as shown below:

- for $1 \le i \le m$, $j \in S_i$,

$$K_{j,1,\mathrm{GID}_i} = (g^{\alpha_\beta} H(\mathrm{GID}_i)^{y_\beta} F(j)^{t_j})^{z_i} = g^{\alpha_\beta z_i} H(\mathrm{GID}_i)^{y_\beta z_i} F(j)^{t_j z_i} \tag{7}$$

$$K_{j,2,\mathrm{GID}_i} = (F(j)^{t_j})^{z_i} = F(j)^{t_j z_i} \tag{8}$$

Order

$$\mathrm{SK}_{\mathrm{Out,GID}_i} = \{S_i, K_{i,1,\mathrm{GID}_i}^{z_i}, K_{i,2,\mathrm{GID}_i}^{z_i}\}_{j \in S_i} \tag{9}$$

- for $m \le i \le n$, chooses the random element $g_j \in G$, $k_j \in \mathbb{Z}_p^*$, calculate $K_{j,1,\mathrm{GID}_i} = g_j F(j)^{k_j} g_j$, $K_{j,2,\mathrm{GID}_i} = F(j)^{k_j}$, order $\mathrm{SK}_{\mathrm{Out,GID}_i} = \{S_i, K_{i,1,\mathrm{GID}_i}^{z_i}, K_{i,2,\mathrm{GID}_i}^{z_i}\}_{j \in S_i}$. Notice the $g^{\alpha_\beta} H(\mathrm{GID}_i)^{y_\beta}$ is an element of group G, and G is a cyclic group, and there is an unknown element $z_i \in \mathbb{Z}_p^*$, s. t. $g_j = (g^{\alpha_\beta} H(\mathrm{GID}_i)^{y_\beta})^{z_i} = g^{\alpha_\beta z_i} H(\mathrm{GID}_i)^{y_\beta z_i}$. So, $K_{j,1,\mathrm{GID}_i} = g_j F(j)^{k_j} = g^{\alpha_\beta z_i} H(\mathrm{GID}_i)^{y_\beta z_i} F(j)^{k_j}$, $K_{j,2,\mathrm{GID}_i} = F(j)^{k_j}$, is a uniform distribution of the outsourced decryption key.
- Φ send the above results to attacker.

Guess: The attacker outputs a guess $b' \in \{0, 1\}$ with Φ at the same time.

The above distribution is truly indistinguishable from the attacker. Therefore, if the attacker can break this scheme with non-negligible advantage, he can also break the RW scheme with non-negligible advantage.

Lemma 2. Given that the q-DPBDHE2 assumption holds, the RW scheme is statically safe under the random oracle model.

Proof: The document [20] has given a detailed proof. For reasons of space, it will not be repeated here.

Theorem 1. Assuming that the q-DPBDHE2 assumption holds, the scheme in this paper is statically secure under the random oracle model.

Proof: Lemmas 1 and 2 can be obtained directly.

(2) Revocable Security Certificate

The idea of this section is similar to that of Lemma 1. First of all, the following lemmas are proved.

Lemma 3. Assuming that the Rouselakis-Waters (RW) scheme [20] is static and secure, the no-centric multi-agency CP-ABE scheme proposed in this paper supports user revocation.

Proof: Assumptions For the proposed scheme of this paper, there is a polynomial-time opponent A who can win the revocable game in Sect. 3 with the advantage ε, then a simulator B can be constructed to defeat the RW scheme with the advantage ε. Let C be the challenger to interact with B in the RW scheme.

Same as static security certificate, Challenger C sends the public parameter GP in the RW scheme to simulator B. B sends the GP as an open parameter of the scheme to adversary A.

Adversary A asks B about the uncorrupted authority's public key, part of the user's public/private key pair, and the corresponding cloud server private key and challenge cipher text.

Challenger's Replies: Simulator B queries the public key and challenge ciphertext of the noncorrupted institution in the RW solution. C returns to B, then B performs the following operations.

– for $1 \le i \le m$, chooses the random element $z \in S_i$,

$$K_{j,1,\mathrm{GID}_i} = (g^{\alpha_\beta} H(\mathrm{GID}_i)^{y_\beta} F(j)^{t_j})^{z_i} = g^{\alpha_\beta z_i} H(\mathrm{GID}_i)^{y_\beta z_i} F(j)^{t_j z_i} \tag{10}$$

$$K_{j,2,\mathrm{GID}_i} = (F(j)^{t_j})^{z_i} = F(j)^{t_j z_i} \tag{11}$$

order

$$\mathrm{SK}_{\mathrm{Out},\mathrm{GID}_i} = \{S_i, K_{i,1,\mathrm{GID}_i}^{z_i}, K_{i,2,\mathrm{GID}_i}^{z_i}\}_{j \in S_i} \tag{12}$$

– for $1 \le i \le m$, chooses the random element $z \in \mathbb{Z}_p^*$, calculate $\mathrm{userPK}_{\mathrm{GID}_i} = \{g^z, H(\mathrm{GID}_i)^z\}$ and $\mathrm{key}_{\mathrm{GID}_i} = \{1/z\}_i$.
– for $m \le i \le n$, $j \in S_i$ chooses the random element $g_j \in G$, $k_j \in \mathbb{Z}_p^*$, calculate $K_{j,1,\mathrm{GID}_i} = g_j F(j)^{k_j} g_j$, $K_{j,2,\mathrm{GID}_i} = F(j)^{k_j}$, order

$$\mathrm{SK}_{\mathrm{Out},\mathrm{GID}_i} = \{S_i, K_{i,1,\mathrm{GID}_i}^{z_i}, K_{i,2,\mathrm{GID}_i}^{z_i}\}_{j \in S_i}. \tag{13}$$

Notice the $g^{\alpha_\beta} H(\mathrm{GID}_i)^{y_\beta}$ is an element of group G, and G is a cyclic group, there is an unknown element $z_i \in \mathbb{Z}_p^*$, s. t.

$$g_j = (g^{\alpha_\beta} H(\mathrm{GID}_i)^{y_\beta})^{z_i} = g^{\alpha_\beta z_i} H(\mathrm{GID}_i)^{y_\beta z_i}. \tag{14}$$

So, $K_{j,1,\mathrm{GID}_i} = g_j F(j)^{k_j} = g^{\alpha_\beta z_i} H(\mathrm{GID}_i)^{y_\beta z_i} F(j)^{k_j}$, $K_{j,2,\mathrm{GID}_i} = F(j)^{k_j}$, is a uniform distribution of the outsourced decryption key.
– B send the above results to A.

Guess: A outputs a guess $b' \in \{0, 1\}$ with B at the same time.

Theorem 2. Assuming that the q-DPBDHE2 hypothesis holds, the scheme in this paper supports user revocation under the random oracle model.

Proof can be obtained directly from Lemmas 2 and 3.

C. Performance Comparison Analysis

This section mainly compares and analyzes the performance of the proposed scheme and related schemes, Table 2 shows the performance comparison results for each scheme. We mainly performed functional analysis and comparison with the current attribute-based encryption schemes proposed for mobile cloud environments.

Except that the scheme [16] is a single authority establishment, other programs are multi-authority institutions. By comparison, it can be seen that scholars choose the bilinear group of prime number order to construct the scheme because the same group operation is 1–2 orders of magnitude faster than the order group in the prime order group, which is more suitable for the mobile cloud environment.

Table 2. Performance comparison of attribute-based encryption schemes proposed in this paper and mobile cloud environment

	AA	CA	Prime order	Pre-calculation	Outsource decryption	Access structure	Safety	Large universe
Vijay [13]	Multi-authority	×	√	×	×	Not mentioned		×
Li [14]	Multi-authority	×	√	×	√	Access Tree	CPA	×
Lyu [15]	Multi-authority	√	√	√	√	LSSS	CPA	×
Zhao [16]	Single	–	√	√	√	Access Tree	CPA	×
De [17]	Multi-authority	√	√	√	√	LSSS	CPA	×
Li [18]	Multi-authority	√	√	√	√	LSSS	CPA	×
Our scheme	Multi-authority	√	√	√	√	LSSS	Static Security	√

Since the biggest difference between a mobile cloud environment and an ordinary cloud environment is the device performance of a user terminal, most schemes consider adopting technical means at the encryption and decryption stages to migrate the amount of computation that should have been completed by the user terminal to a third party and encrypt the information. Files are securely stored to the cloud. Schemes [13, 14] all require a central authority to certify their identities. This does not prevent the central organization from corrupting the entire encryption process. The central authority must interact with each authority to exchange information. The communication costs brought about by this cannot be ignored either. The [15, 17, 18] are similar to the schemes proposed in this paper, but our scheme is static secure and is superior to the plaintext security of the above scheme.

Our scheme draws on the mapping idea in [22] and uses functions $F : U \rightarrow G$ to map the attribute space to group G. [22] was proposed in cloud computing environment and did not consider precomputing, Table 3 gives the performance comparison results of our scheme and the scheme proposed by [22]. The advantage of this is that the number of attributes in the system is not limited, and any string in group G can be used as a new one in the later period. That is, our scheme is to support large attribute universe. There is no need to specify the number of attributes to be used when the system is established. In addition, this scheme maps the user identifier GID to G through the function H, so that the user and organization that have the unique identifier can achieve complete decentralization, thereby resisting the collusion attack between the user and the organization.

In addition, the solution of this paper is also superior to the multi-authority scheme proposed in the current cloud environment in terms of user costs and is more suitable for the requirements of the user's low-end and configuration equipment.

Table 3. Performance comparison of our scheme and [22]

	Environment	Pre-calculation	Outsource	Encryption of users	Decryption of users
Ref [22]	Cloud computing	×	√	$(6\ell+1)E$	E
Our scheme	MCC	√	√	E	E

6 Conclusion

At present, the research of mobile cloud data security sharing mechanism is still at the initial stage, and there is no secure, effective and thorough trusted deletion scheme.

The scheme proposed in this paper adopts attribute-based encryption technology, which utilizes pre-computation and outsourcing decryption, effectively reduces the computational overhead of the client, satisfies the large attribute domain, supports revocation, and has no central setting, which is more in line with practical application requirements. Finally, the security of the scheme is proved under the random oracle model. Compared with related schemes, our method can effectively reduce the overhead of mobile devices and is suitable for data security sharing in mobile cloud environments.

Acknowledgments. This work was supported by the National Key Research Program of China "Collaborative Precision Positioning Project" (Grant No.2016YFB0501900)

References

1. Wikipedia: The definition of Mobile Cloud Computing. https://en.wikipedia.org/wiki/Mobile_cloud_computing
2. White Paper: Mobile Cloud Computing Solution Brief. Aepona (2010)
3. Cui, Y., Song, J., Miao, C., et al.: Mobile cloud computing research progress and trends. Chin. J. Comput. **40**(2), 273–295 (2017)
4. Gartner: Five trends in cybersecurity for 2017 and 2018. https://www.gartner.com/smarterwithgartner/5-trends-in-cybersecurity-for-2017-and-2018/
5. Sahai, A., Waters, B.: Fuzzy identity-based encryption. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 457–473. Springer, Heidelberg (2005). https://doi.org/10.1007/11426639_27
6. Goyal, V., Pandey, O., Sahai, A., et al.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the ACM Conference on Computer and Communications Security, pp. 89–98 (2006)
7. Cheung, L., Newport, C.: Provably secure ciphertext policy ABE. In: Proceedings of the ACM Conference on Computer and Communications Security, pp. 456–465. ACM (2007)
8. Goyal, V., Jain, A., Pandey, O., Sahai, A.: Bounded ciphertext policy attribute based encryption. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008. LNCS, vol. 5126, pp. 579–591. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70583-3_47

9. Waters, B.: Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization. In: Catalano, D., Fazio, N., Gennaro, R., Nicolosi, A. (eds.) PKC 2011. LNCS, vol. 6571, pp. 53–70. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19379-8_4
10. Green, M., Hohenberger, S., Waters, B.: Outsourcing the decryption of ABE ciphertexts. In: Usenix Conference on Security, p. 34. USENIX Association (2011)
11. Hohenberger, S., Waters, B.: Online/offline attribute-based encryption. In: Krawczyk, H. (ed.) PKC 2014. LNCS, vol. 8383, pp. 293–310. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-54631-0_17
12. Lewko, A., Waters, B.: Decentralizing attribute-based encryption. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 568–588. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20465-4_31
13. Vijay, H., Goyal, D., Singla, S.: An efficient and secure solution for attribute revocation problem utilizing CP-ABE scheme in mobile cloud computing. Int. J. Comput. Appl. **129**(1), 16–21 (2015)
14. Li, R., Shen, C., He, H., et al.: A lightweight secure data sharing scheme for mobile cloud computing. IEEE Trans. Cloud Comput. **6**(2), 344–357 (2017)
15. Lyu, M., Li, X., Li, H.: Efficient, verifiable and privacy preserving decentralized attribute-based encryption for mobile cloud computing. In: IEEE Second International Conference on Data Science in Cyberspace, pp. 195–204. IEEE Computer Society (2017)
16. Zhao, Z., Wang, J.: Verifiable outsourced ciphertext-policy attribute-based encryption for mobile cloud computing. Ksii Trans. Internet Inf. Syst. **11**(6), 3254–3272 (2017)
17. De, S.J., Ruj, S.: Efficient decentralized attribute based access control for mobile clouds. IEEE Trans. Cloud Comput. (2017)
18. Li, X., Lyu, M.: Multi-authority attribute-based encryption scheme in mobile cloud environment. Appl. Res. Comput. **35**(05), 1–9 (2018)
19. http://www.arocmag.com/article/02-2018-05-006.html
20. Lewko, A., Okamoto, T., Sahai, A., Takashima, K., Waters, B.: Fully secure functional encryption: attribute-based encryption and (hierarchical) inner product encryption. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 62–91. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_4
21. Rouselakis, Y., Waters, B.: Efficient statically-secure large-universe multi-authority attribute-based encryption. In: Böhme, R., Okamoto, T. (eds.) FC 2015. LNCS, vol. 8975, pp. 315–332. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-47854-7_19
22. Beimel, A.: Secure schemes for secret sharing and key distribution. Int. J. Pure Appl. Math. (1996)
23. Zhang, K., Ma, J., Li, H., et al.: Multi-authority attribute-based encryption with efficient revocation. J. Commun. **38**(3), 83–91 (2017)