International Journal of Security and Its Applications
Vol.7, No.6 (2013), pp.385-394
http://dx.doi.org/10.14257/ijsia.2013.7.6.38

Optimized Ciphertext-Policy Attribute-Based Encryption with Efficient Revocation

Yang Li[1], Jianming Zhu[1], Xiuli Wang[1], Yanmei Chai[1] and Shuai Shao[2]

[1] *College of Information, Central University of Finance and Economics, Beijing, 100081, China*
[2] *China Information Technology Security Evaluation Center, Beijing, 100085, China*
liyang@cufe.edu.cn, zjm@cufe.edu.cn, wangcufe@163.com, chai-4@163.com, shaoshuaib@163.com

Abstract

In this work, we design a method for efficient revocation within Ciphertext-Policy Attribute-Based Encryption scheme. Our main technical innovation is based on linear secret sharing and binary tree techniques, every user in system is assigned with both a set of attributes and a unique identifier. Any user can also be efficiently revoked by using this identifier. Furthermore, this technique resulted in two key contributions: the size of the cryptographic key material is smaller and encryption/decryption cannot be affected with an unbounded number of revoked users. Then, the scheme is proved to be secure under the q-MEBDH assumption in the standard model. The efficiency is also optimized that the size of user's private key has only a constant increase. The revocation information is embedded in the ciphertext so that the fine-grained access control is more flexible.

Keywords: *Ciphertext-Policy Attribute-Based Encryption; Efficient Revocation; Linear Secret Sharing Schemes; Security Model*

1 Introduction

In traditional public-key cryptosystem, a pair of keys were used to encrypt. So messages encrypted by sender with access to a public key certificate could only be decrypted by the private key, and vice versa. However, this one could not support expressive access control policies with the development of Internet and distributed computing technology today. Now, the ability to do public key encryption without certificates has many practical applications. For example, a user wanted to send an encrypted urgent mail to recipients, it must be dealt with no time to via either the existence of a Public-Key Infrastructure (PKI) or on-line recipients.

Attribute-Based encryption (ABE) scheme, proposed by Sahai and Waters [1] in 2005, allowed a party to encrypt messages to all users that have a certain set of attributes. It took attributes as the public key and associated the ciphertext and user's private key with attributes. The ciphertext within a certain number of specified attributes could be decrypted by any private keys satisfies the requested attributes that match the access control policies. The advantage to using ABE scheme was that messages can be stored on a simple untrusted storage server instead of relying on trusted server to perform authentication checks before delivering.

However, the main drawback of ABE scheme was that its construction was limited to support threshold access control policy only, without more flexible. In subsequent work, Key-Policy ABE (KP-ABE) was proposed by Goyal, et al. [2]. In this scheme, the attributes were associated with ciphertexts and access structures were associated with user secret keys. "AND" and "OR" were added between attributes in user secret key's access formula and tree structure was used to describe access control policy. It made a major step

ISSN: 1738-9976 IJSIA

forward beyond traditional ABE, but "NOT" operation could not be involved in representing an access structure. Ostrovsky, *et al.*, [3] found a way as broadcast revoke mechanism to solve this defect. Another type of ABE, called Ciphertext-Policy ABE (CP-ABE), was proposed by Bethencourt, *et al.*, [4]. This scheme allowed a sender to disseminate messages according to an access policy which can be expressed as a Boolean function consisting of (AND, OR) gates between attributes. These attributes were associated with user secret keys and access structures were associated with ciphertexts. A receiver whose secret key was associated with those attributes could only decrypt a ciphertext successfully if and only if its attributes satisfy the ciphertext's access policy. However, it could not be proved secure in the standard model. After that, many elegant works [5, 6] proved secure in standard model have been presented later, where tradeoff between expressiveness of access policy and security assumption was made. Waters [7] improved the CP-ABE to make it provably secure and more efficient. A general method was presented to construct a CP-ABE using linear secret sharing technique to achieve more efficient so far. The construction proposed in this paper followed Waters' work.

CP-ABE scheme, similar with role-based access control system, could be widely applied to realize access control in many applications. For example, in medical field, the sensitive medical records, tightly related to patients' privacy, must be accessed only if the users were authorized by patients; in educational field, solutions for exams in the online system also should be only read by specified teachers. The CP-ABE scheme dealt with those situations, by encrypting the target information with expressive access policies, such as {"Medicine" AND "Physician"}, {"Computer Science" AND ("Professor" OR "Assistant professors")}. In fact, CP-ABE provided a perfect solution to an access control system by considering, efficient distributing, expressive access control and data confidentiality.

In the previous ABE schemes, once private keys associated with attributes were obtained by users after the key generation phase, they were treated as activated in ideal condition unless getting compromised or attributes failure caused by dynamic changing of system. From the aspect of users, once users obtained the credentials from a system administrator at the beginning of setup phase, the access ability was always valid for those who might break the confidential rules later by abusing these private information. Upon detecting those malicious adversaries, without any revocation mechanism embedded then, the system administrator had to rebuild up the whole system. Therefore, revocation mechanism should be designed into the system from the beginning and required taking into account where the revocation mechanism should be placed and how to decrease the computational and communication costs.

In this paper, we aimed at proposing an optimized CP-ABE encryption with efficient revocation. However, designing a revocation mechanism for CP-ABE scheme was not a simple task while considering the following aspects:

1. System administrator could only associated user secret keys with different subsets of attributes instead of individual characteristics. The fuzzy identities scheme therefore blocked the system revocation on one specified user;
2. The characteristics of users were taken place by several common attributes, and thus revocation on attributes could not accurately exclude the users with misbehaviors;
3. The system must be secure against collusion attack from revoked users even though they shared some common attributes with non-revoked users.

To consider the revocation solution in a traditional CP-ABE scheme, the choices were limited. One of them was the revocation of a single attribute, which was not related with users' behaviors but more likely update of universal attribute set of the whole system periodically. Another was to revoke one subset of attributes corresponding to a specific set of users, but all the users' access abilities might be revoked at the same time if they

shared the same subset of attributes with the malicious user, which was inappropriate in the real application environment.

Contribution. An optimized CP-ABE scheme with efficient revocation based on Waters [7] was presented in this paper. In our construction, every user private keys were associated with a unique identifier. A set of revocation ID $RS_{ID} = \{ID_1, ID_2, \cdots, ID_r\}$ was embedded in the ciphertext. While user receiving the ciphertext, if the *ID* with private keys owned by receivers matched ID_i in the set of RS_{ID}, the decryption would be failure, otherwise, success.

This scheme had two important features relating to public and private key size respectively.

First, public keys in this scheme were short and enabled a user to create a ciphertext that revoked an unbounded number of users. This was in contrast to other systems [8, 9] where the public parameters bounded the number of users in the system and must be updated to allow more users.

Second, the cryptographic key material was smaller so that it could be stored securely on the receiving devices. Keeping the size of private key storage as low as possible was important as cryptographic keys would often be stored in tamper-resistant memory, which is more costly. This could be especially critical in small devices such as sensor nodes, where maintaining low device cost is particularly crucial. Device keys in this scheme were only a small constant number of group elements from an elliptic-curve group of prime order. In addition, this scheme were public-key stateless broadcast encryption scheme, and we work with stateless receivers.

We achieved this small device key size without compromising on other critical parameters such as ciphertext length. Ciphertexts would consist of just O(*r*+*l*) group elements, where *r* was the number of revoked users and *l was* the number of attributes. This was the same behavior as the previously best-known schemes for revocation. We also didn't compromise on security: we obtained adaptive security in the standard model under the well-established q-MEBDH and decisional linear assumptions.

Related works. The revocation problem in public key encryption scheme had been well studied [10, 11]. Efficient revocation of certificates had been an active topic in the past several years [12, 13]. Gentry [14] also discussed the certificate revocation problem in certificate-based encryption scheme.

Several research works related to revocation in Identity-Based Encryption (IBE) setting were as follows. The schemes in [15, 16] accommodated a special semi-trusted third party called mediator who was able to provide help for the non-revoked users on decryption. Boldyreva, *et al.*, [17] adopted the techniques of fuzzy IBE and binary tree to implement a revocation scheme in the IBE setting which reduced the amounts of update information in comparison with previous works. They also presented an intuitional way to apply the same technique to KP-ABE scheme and Fuzzy IBE scheme. Yu, *et al.*, [18] proposed a tailored KP-ABE with revocation especially for fined-grained distribute data access control in wireless sensor networks. However, the periodical change of system public parameters in [18] introduced extra computational and communication costs.

For the research on solving revocation problem in broadcast encryption, Boneh and Waters [19] introduced a new primitive called augmented broadcast encryption scheme which can be constructed for broadcast encryption with trace-and-revoke function. Liang, *et al.*, [20] proposed a CP-ABE-R scheme by using the binary tree structure and let the system administrator control the revocation list. Lewko, *et al.*, [21] proposed a revocation scheme with very small secret keys but revocation list was controlled by the encryption user itself.

Organization. Section 2 gives a brief review on definition of linear secret sharing scheme, a new definition on algorithms in Revocation Scheme for CP-ABE and the

corresponding security model for it. In Sections 3, we propose a one more efficient Revocation Scheme for CP-ABE and present a complete proof in the standard model. The discussion on efficiency, delegating capability and chosen ciphertext security of our scheme are given in Section 4 and we conclude our paper in Section 5.

2. Preliminaries

2.1 Linear Secret Sharing Schemes (LSSS) [22]

LSSS was a useful technique in constructing ABE scheme. It could be summarized as follows:
A secret-sharing scheme Π over a set of parties P is called linear (over $\mathbb{Z}_p$) if:

1. The shares for each party form a vector over $\mathbb{Z}_p$.
2. There exists a matrix an M with l rows and n columns called the share-generating matrix for Π. For all $i = 1,2,\cdots,l$, the ith row of M we let the function ρ defined the party representing row i as $\rho(i)$. When we consider the column vector $v = (s, r_2, \cdots, r_n)$, where $s\epsilon\mathbb{Z}_p$ is the secret to be shared, and $r_2, \cdots, r_n\epsilon\mathbb{Z}_p$ are randomly chosen, then Mv was the vector of l shares of the secret s according to Π. The share $(Mv)_i$ belongs to party $\rho(i)$.

Suppose [6] that Π is an LSSS for the access structure $\mathbb{A}$. Let $S \in \mathbb{A}$ be any authorized set, and let $I \subset \{1,2,\cdots,l\}$ 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 $\sum_{i\in I} \omega_i\lambda_i = s$.
Access formulas could be represented in terms of binary trees structure. On this basis, any monotonic Boolean formula could be converted into an LSSS representation using standard techniques described in. An access tree of l nodes would result in an LSSS matrix of l rows.

2.2. Revocation Scheme for CP-ABE

A CP-ABE scheme usually included four phrases: Setup, Encrypt, KeyGen, and Decrypt. Our construction also followed in this way.

Setup(U,n_{max}) The authority ran the setup algorithm. The algorithm took the universal attribute set U and the maximum size n_{max} of columns in an access structure as input. It outputted the public parameters *PK* and a master key *MSK*.

KeyGen(*MSK*,*S*,*ID*) The key generation algorithm took the master key *MSK*, an attributes set $S\subseteq U$ and a unique identifier *ID* as input. It outputted a secret key (ID,D_{ID}) to the user.

Encrypt(*PK*,(M,ρ),$\mathbb{M}$,RS_{ID}) The encryption algorithm took the public parameters *PK*, an access structure (M,ρ) over the universe of attributes, a message $\mathbb{M}$ and revocation ID set RS_{ID}. It outputted ciphertext C with (M,ρ). C could be decrypted only by a user who possessed a subset of attributes that satisfied (M,ρ) and whose ID was not in the RS_{ID}.

Decrypt(C,(M,ρ),(ID,D_{ID}),RS_{ID}) The decryption algorithm took the ciphertext C with (M, ρ), the secret key D_{ID} and revocation ID set RS_{ID} for it. If any arbitrary subset of attribute related with D_{ID} satisfied the access structure (M,ρ) and the unique identifier *ID* associated with D_{ID} had not been revoked in revocation ID set RS_{ID}, it decrypted the ciphertext C and returns a message $\mathbb{M}$; else, it returned $\perp$.

2.3 Security Model for Revocation Scheme for CP-ABE

Selective Access Structure Model for revocation scheme of CP-ABE Selective Access Structure (SAS) Model was widely used in analyzing CP-ABE. In this paper, we proposed

the SAS model for a revocation scheme of CP-ABE. This scheme was secure in the SAS model defined between a challenger and an adversary $\mathcal{A}$, if no probabilistic polynomial-time adversary $\mathcal{A}$ had a nonnegligible advantage in winning the following game.

INIT $\mathcal{A}$ chose an access structure (M^*, ρ^*), a set of unique identifier ID^*, where the column of M^* was no larger than n_{max}. The queries of key generation oracle for $S \in (M^*, \rho^*)$ were associated with the unique identifiers in ID^*.

SETUP The challenger ran the setup algorithm and told adversary $\mathcal{A}$ the public parameters *PK* and kept the corresponding master key *MK* for itself.

PHASE 1 $\mathcal{A}$ issued several queries to key generation oracle and revoke oracle.
Key generation oracle: $\mathcal{A}$ issued queries for secret keys related to several tuples (*ID*,*S*) where the unique identifiers $ID \in ID^*$. The challenger gave $\mathcal{A}$ the corresponding private keys D_{ID}(*ID*,*S*).
Revoke oracle: $\mathcal{A}$ inputted several revoked unique identifiers *ID*, the simulator added *ID* to the revocation ID set RS_{ID}.

CHALLENGE Once $\mathcal{A}$ decided that **PHASE 1** is over, it generated two messages $\mathbb{M}_0$, $\mathbb{M}_1$ from the message space of equal length and gave them to challenger with RS_{ID} included all identities queried previously. Next, the challenger flipped a coin $b \in \{0,1\}$ at random, and encrypted $\mathbb{M}_b$ with (M^*, ρ^*) to obtain ciphertext C^*. Then, it was returned back to the $\mathcal{A}$.

PHASE 2 The same as **PHASE 1**.

GUESS $\mathcal{A}$ outputted a guess $b' \in \{0,1\}$ and win the game if $b' = b$.
The following conditions must always hold. In the key generation oracle, once tuples (*ID*,*S*) was queried, the adversary $\mathcal{A}$ would not query any other tuples (*ID*,*S'*), where $S' \neq S$.

A revocation CP-ABE system was secure in this model of security if all polynomial-time adversaries had at most a negligible advantage in the above game as $|Pr[b' = b] - 1/2|$.

3. Our More Efficient Construction

In our construction, the encryption algorithm would take as input a LSSS access matrix *M* and distribute a random exponent $s \in \mathbb{Z}_p$ according to *M*. Our system had the following features: both public and private keys were of size independent of the number of users (*i.e.*, only a constant number of group elements); the private keys were randomized chosen to avoid collusion attack; the ciphertext only contained O(*r*+*l*) group elements.

Intuition. Our construction used a novel application of a secret sharing in the exponent. Suppose an encryption algorithm need to create an encryption with a revocation set RS_{ID} = ID_1,…, ID_r of *r* identities. The algorithm would create an random exponent $s \in \mathbb{Z}_p$ and split it into *r* random shares s_1,…, s_r such that $\sum_{i=1}^{r} s_i = s$. It would create a ciphertext such that any user secret keys with *ID* = ID_i could not be able to incorporate the *i*-th share and thus not decrypt the message.

Our approach presented us with two challenges. First, we need to make sure that a user with revoked identity *ID* = ID_i could not do anything useful with share *i*. Second, we need to worry about collusion attacks between multiple revoked users. Suppose a user with *ID* = ID_i and a user with *ID* = ID_j collude to attack a ciphertext. The attack that we need to worry about was where user *j* processed ciphertext share *i*, while user *i* processed share *j*, and then they combined their results.

The first problem was addressed by the method of decryption. For each share, the ciphertext will have two components. A user with $ID \neq ID_i$ could use these two

components to obtain two linearly independent equations (in the exponent) involving the share s_i (and another variable), which he would use to solve for the share s_i. However, if $ID = ID_i$, user would get two linearly dependent equations and not be able to solve the system. We remark that these techniques are somewhat reminiscent of those used for knowledge extraction in discrete log proof of knowledge settings. In addition, different types of two equation techniques had been applied in e-commerce applications.

To address the second challenge, we randomized each user's private key by an exponent t such that in decryption each user recovered shares $t \cdot s_i$ in the exponent. Thus, we disallowed realizable collusions in a similar manner to some Identity-Based [23, 24] and Attribute-Based [1, 2] encryption systems.

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplicative cyclic groups of the same order p. Let g be a generator of $\mathbb{G}$, and suppose $\mathbb{G}$ and $\mathbb{G}_T$ were equipped with a pairing, *i.e.*, a non-degenerated and efficiently computable bilinear map e: $\mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$. Then the bilinear map e had the following properties:

1. Bilinearity: for all $u, v \in \mathbb{G}$ and $a, b \in \mathbb{Z}_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$.
2. Non-degeneracy: $e(g, g) \neq 1$.

Setup(U,n_{max}) The setup algorithm took as input the universal attribute set U in the system and the maximum size n_{max} of columns in one access structure.
It first chose a group $\mathbb{G}$ of prime order p, two generators $g, h \in \mathbb{G}$ and U random group elements $h_1, h_2, \cdots, h_U \in \mathbb{G}$ that were associated with the U attributes in the system. It then picked random exponents $\alpha, a \in \mathbb{Z}_p$. The system public key was published as:

$PK = g, e(g, g)^{\alpha}, g^a, g^{a^2}, h^a, h_1, h_2, \cdots, h_U$
The system master secret key was published as:
$MSK = g^{\alpha}$

Encrypt(PK, (M, ρ) , $\mathbb{M}$,RS_{ID}) The encryption algorithm took as input the public parameters PK, an LSSS access structure (M, ρ) and message $\mathbb{M}$ to encrypt, while the function ρ associated rows of M to attributes.

M in (M, ρ) was a matrix with size $l \times n$, we expanded M to a $l \times n_{max}$ matrix by filling element 0 into the columns from (n+1)-th to n_{max}-th. Note that such conversion didn't affect the satisfying logic of an access structure. The encryption algorithm first chose a random vector $\vec{v} = (s, y_2, \cdots, y_{n_{max}}) \in \mathbb{Z}_p^{n_{max}}$. These values would be used to share the encryption exponent s. For i = 1 to l, it calculated $\lambda_i = \vec{v} \cdot M_i$, where M_i is the vector corresponding to the ith row of M.

The encryption algorithm also need to create an encryption with a the set of revocation ID. Then, the encryption algorithm set $r = |RS_{ID}|$ and chose random $s_1, \cdots, s_r$ such that $s = s_1 + \cdots + s_r$. We let ID_i denote the ith identity in RS_{ID}.

In addition, the algorithm chose random $r_1, \cdots, r_l \in \mathbb{Z}_p$. It then created the ciphertext C as:
$C' = \mathbb{M}e(g, g)^{\alpha s}, C_0 = g^s$
for $i = 1,2, \cdots, r$, $C_{i,1} = g^{as_i}$, $C_{i,2} = \left(g^{a^2 ID_i} h^a\right)^{s_i}$
for $j = 1,2, \cdots, l$, $\left(C_j = g^{a\lambda_j} h_{\rho(j)}^{-r_j}, D_j = g^{r_j}\right)$

KeyGen(MSK,S,ID) The key generation algorithm took a unique identifier ID, an attribute set S and the master key MSK as input.

Firstly, the algorithm checked the unique identifier ID to see whether it had been queried before. If the answer is yes, S must be the same with the one in the previous query and the algorithm outputted the same secret key; if not, the algorithm chose a vacant leaf node in the tree and bound it with ID.

Then the algorithm chose a random $t \in \mathbb{Z}_p$. It created the private key as:
$K = g^{\alpha} g^{at} g^{a^2 t}$, $L = g^{-t}$, $D_{ID} = (g^{aID} h)^t$

$\forall x \in S, K_x = h_x^t$

Decrypt(C,ID,D_{ID},RS_{ID}) The decryption algorithm took as input a ciphertext C for access structure (M, ρ), a unique identifier ID, a private key D_{ID} for a set S and revocation ID set RS_{ID}.

Suppose that S satisfied the access structure $\mathbb{A}$ and let $I \subset \{1,2,\cdots,l\}$ be defined as $I = \{i: \rho(i) \in S\}$. Then, let $\{\omega_i \in \mathbb{Z}_p\}_{i \in I}$ be a set of constants such that if $\{\lambda_i\}$ were valid shares of any secret s according to M, then $\sum_{i \in I} \omega_i \lambda_i = s$. (Note there could potentially be different ways of choosing the ω_i values to satisfy this.) If there existed $ID^* \in RS_{ID}$ such that $ID = ID^*$ then the algorithm aborted; otherwise, the decryption algorithm computed as:

$$e(C_0, K) / \left[e\left(D_{ID}, \prod_{i=1}^{r} C_{i,1}^{1/(ID-ID_i)}\right) \cdot e\left(L, \prod_{i=1}^{r} C_{i,2}^{1/(ID-ID_i)}\right) \cdot \prod_{i \in I}\left(e(C_i, 1/L) e\left(D_i, K_{\rho(i)}\right)\right)^{\omega_i}\right] \quad (1)$$

Which gave us $e(g,g)^{\alpha s}$; this could immediately be used to recover the message $\mathbb{M}$ from C'. Note that this computation was only defined if $\forall i \ \ ID \neq ID^*$.

We could verify the correctness of the decryption computation, and each component of formula (1) extended as:

$$e(C_0, K) = e\left(g^s, g^{\alpha} g^{at} g^{a^2 t}\right) = e(g,g)^{\alpha s} \cdot e(g,g)^{sat} \cdot e(g,g)^{sa^2 t} \quad (2)$$

It obtained $e(g,g)^{sat}$ as follows:

$$\prod_{i \in I}\left(e(C_i, 1/L) e\left(D_i, K_{\rho(i)}\right)\right)^{\omega_i} = \prod_{i \in I}\left(e\left(g^{a\lambda_i} h_{\rho(i)}^{-r_i}, g^t\right) e\left(g^{r_i}, h_{\rho(i)}^t\right)\right)^{\omega_i} =$$

$$\prod_{i \in I} e(g,g)^{ta\lambda_i\omega_i} = e(g,g)^{ta\sum_{i \in I}\lambda_i\omega_i} = e(g,g)^{sat} \quad (3)$$

It obtained $e(g,g)^{sa^2 t}$ as follows:

$$e\left(D_{ID}, \prod_{i=1}^{r} C_{i,1}^{1/(ID-ID_i)}\right) \cdot e\left(L, \prod_{i=1}^{r} C_{i,2}^{1/(ID-ID_i)}\right) =$$

$$\prod_{i=1}^{r}\left(e\left(D_{ID}, I_{i,1}\right) \cdot e\left(L, I_{i,2}\right)\right)^{1/(ID-ID_i)} =$$

$$\prod_{i=1}^{r}\left(e\left((g^{aID}h)^t, g^{as_i}\right) \cdot e\left(g^{-t}, \left(g^{a^2 ID_i} h^a\right)^{s_i}\right)\right)^{1/(ID-ID_i)} = \prod_{i=1}^{r} e(g,g)^{s_i a^2 t} =$$

$$e(g,g)^{a^2 t \sum_{i=1}^{r} s_i} = e(g,g)^{sa^2 t} \quad (4)$$

We could deduce that the result of formula (1) was $e(g,g)^{\alpha s}$ from (2), (3) and (4). Then for $C' = \mathbb{M} e(g,g)^{\alpha s}$, we can acquire $\mathbb{M}$ from C.

From formula (1) and (4), we noted that if $ID = ID_i$, we wouldn't get the expected result, which determined the premises of (1) was that the user who decrypted CT must not be in the set of revocation ID.

4. Security and Efficiency

The security proof given by Waters [7] was under the decisional Parallel Bilinear Diffie-Hellman Exponent (PBDHE) assumption which could be considered as a generalization of the decisional-Bilinear Diffie-Hellman Exponent (BDHE) assumption. The security proof of our scheme was improved and under the q-decisional Multi-Exponent Bilinear Diffie-Hellman (q-MEBDH) assumption which is proposed by [21].

4.1. q-MEBDH Assumption [21]

Let $\mathbb{G}$ be a bilinear group of prime order p. The q-MEBDH problem in $\mathbb{G}$ was stated as follows:

A challenger picked a generator $g \in \mathbb{G}$ and random exponents $s, \alpha, a_1, \cdots, a_q$.

The attacker was then given $\vec{y} =$

$g, g^s, e(g,g)^\alpha$
$\forall\, 1 \le i,j \le q,\ g^{a_i}, g^{a_i s}, g^{a_i a_j}, g^{\alpha/a_i^2}$
$\forall\, 1 \le i,j,k \le q, i \ne j,$
$g^{a_i a_j s}, g^{\alpha a_j/a_i^2}, g^{\alpha a_i a_j/a_k^2}, g^{\alpha a_i^2/a_j^2}$

It must remain hard to distinguish $e(g,g)^{\alpha s} \in \mathbb{G}_T$ from a random element in $\mathbb{G}_T$.

An algorithm $\mathcal{B}$ outputted $z \in \{0,1\}$ had advantage ε in solving decisional q-parallel BDHE in $\mathbb{G}$ if $|Pr[\mathcal{B}(\vec{y}, T = e(g,g)^{\alpha s}) = 0] - Pr[\mathcal{B}(\vec{y}, T = R) = 0]| \ge \varepsilon$.
We said that the q-MEBDH assumption held if no poly-time algorithm has non-negligible advantage in solving the q-MEBDH problem.

4.2. Security Analysis

The purpose of the colluders was to obtain $e(g,g)^{\alpha s}$, when the proposed scheme was compromised. The only way to get it was to compute $(g,g)^{sat}$ and $e(g,g)^{sa^2t}$ first. For this scheme under the PBDHE and q-MEBDH assumption, the adversary couldn't distinguish between $e(g,g)^{\alpha s}$ and $e(g,g)^{sat}$, $e(g,g)^{sa^2t}$ from the given information.
Because the private keys of users were different, even though all users colluded they would not obtain information about the master secret key. Next, we evaluated the security features of our scheme.

1. Key delegation: Hinek, Jiang *et al.*, [25] proposed the first ABE scheme to consider key delegation as a security threat and a solution for it. It introduced a token server that issued tokens which were necessary for decryption. For example, if a user had the private key for the set of attributes {"information", "student"}, he may delegate a key for the attribute "information". But our scheme derived the key delegation without a token server. Each user had a unique identifier, and the system could trace him by it when a user delegated his key.

2. Fine-grained access control: This system facilitated granting access rights to a set of users, even if necessary, also specified the access rights to individual user. Goyal, Pandey *et al.*, [2] proposed a CP-ABE scheme that constructed an access tree to realize fine-grained access control. For the revocation set changing, our scheme could specify the access rights to individual user for every ciphertext.

4.3. Efficiency

The size of user secret key always increased with the number of users in most of revocation schemes proposed before, now it is $n + |S| + 3$ group elements in our scheme, where n is the size of columns in the access structure (M, ρ), S is the set of attributes corresponding to the user secret key. Our scheme also had no key update phrase compared with most of the other schemes. It not only decreased the communication requirements, but also was more appropriate and more flexible for distributed systems. Thus, any user could be revoked temporarily and effected again after a while. The comparisons between [20] and our scheme shown in Table 1was from different aspects like public parameters size (PP), secret key size (SK) and ciphertext size (CT).

Table 1. Efficiency Comparison

	Our scheme	[20]
PP	$(n\|U\| + 4) \cdot \|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(n\|U\| + 6) \cdot \|\mathbb{G}\| + \|\mathbb{G}_T\| + \|\mathbb{Z}_p\|$
SK	$(n + \|S\| + 3) \cdot \|\mathbb{G}\|$	$(n + \|S\| + 3) \cdot \|\mathbb{G}\| \cdot \log n$
CT	$(ln + 1 + 2r) \cdot \|\mathbb{G}\| + \|\mathbb{G}_T\|$	$(ln + 3) \cdot \|\mathbb{G}\| + \|\mathbb{G}_T\|$

5. Conclusion

In this work, we design a method for efficient revocation within Ciphertext-Policy Attribute-Based Encryption scheme. Our main technical innovation is based on linear secret sharing and binary tree techniques, every user in system is assigned with both a set of attributes and a unique identifier. Any user can also be efficiently revoked by using this identifier. Furthermore, this technique resulted in two key contributions: the size of the cryptographic key material is smaller and encryption/decryption cannot be affected with an unbounded number of revoked users. Then, the scheme is proved to be secure under the q-MEBDH assumption in the standard model. The efficiency is also optimized that the size of user's private key has only a constant increase. The revocation information is embedded in the ciphertext so that the fine-grained access control is more flexible.

Acknowledgements

The paper is supported by the National Natural Science Foundation of China (No. 61272398), National Social Science Foundation of China (No. 13AXW010) and Beijing Natural Science Foundation (No. 4112053).

References

[1] A. Sahai and B. Waters, "Fuzzy Identity Based Encryption", EUROCRYPT'05, **(2005)**, pp. 457-473.
[2] V. Goyal, O. Pandey, A. Sahai and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data", ACM CCS'06, **(2006)**, pp. 89-98.
[3] R. Ostrovsky, A. Sahai and B. Waters, "Attribute-based encryption with non-monotonic access structures", ACM CCS'07, **(2007)**, pp. 195-203.
[4] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-policy attribute-based encryption", IEEE SP'07, **(2007)**, pp. 321-334.
[5] V. Goyal, A. Jain, O. Pandey and A. Sahai, "Bounded ciphertext policy attribute based encryption, ICALP'08, **(2008)**, pp. 579–591.
[6] X. Liang, Z. Cao, H. Lin, D. Xing. Provably secure and efficient bounded ciphertext policy attribute based encryption", ASIACCS'09, **(2009)**, pp. 343-352.
[7] B. Waters, "Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization", PKC'11, pp. 53-70, **(2011)**.
[8] D. Boneh, C. Gentry and B. Waters, "Collusion resistant broadcast encryption with short ciphertexts and private keys", CRYPTO, **(2005)**, pp. 258-275.
[9] C. Delerablee, P. Paillier and D. Pointcheval, "Fully collusion secure dynamic broadcast encryption with constant-size ciphertexts or decryption keys", Pairing, **(2007)**, pp. 39-59.
[10] D. Naor, M. Naor and J. Lotspiech, "Revocation and tracing schemes for stateless receivers", CRYPTO'01, pp. 41-62.
[11] M. Naor and B. Pinkas, "Efficient trace and revoke schemes", Financial Cryptography'00, **(2000)**, pp. 1-20.
[12] V. Goyal, "Certificate revocation using fine grained certificate space partitioning", FC'07, **(2007)**, pp. 247-259.
[13] M. Naor and K. Nissim, "Certificate revocation and certificate update", SSYM'98, **(1998)**, pp. 17.
[14] C. Gentry, "Certificate-based encryption and the certificate revocation problem", EUROCRYPT'03, **(2003)**, pp. 272–293.
[15] D. Boneh, X. Ding, G. Tsudik and M. Wong, "A method for fast revocation of public key certificates and security capabilities", SSYM'01, pp. 22–22, **(2001)**.
[16] B. Libert and J.-J. Quisquater, "Efficient revocation and threshold pairing based cryptosystems", PODC'03, **(2003)**, pp. 163-171.
[17] A. Boldyreva, V. Goyal and V. Kumar, "Identity-based encryption with efficient revocation", ACM CCS'08, pp. 417-426, **(2008)**.
[18] S. Yu, K. Ren, W. Lou, "Fdac: Toward fine-grained distributed data access control in wireless sensor networks", IEEE INFOCOM '11, **(2011)**, pp. 673–686.

[19] D. Boneh and B. Waters, "A fully collusion resistant broadcast, trace, and revoke system", ACM CCS'06, **(2006)**, pp. 211–220.
[20] X. Liang, R. Lu, X. Lin and X. (Sherman) Shen, "Ciphertext Policy Attribute Based Encryption with Efficient Revocation" Technique Report. BBCR-TR-200908. http://bbcr.uwaterloo.ca/~x27liang/.
[21] A. Lewko, A. Sahai and B. Waters, "Revocation systems with very small private keys", IEEE SP'10, **(2010)**, pp. 273-285.
[22] A. Beimel, "Secure Schemes for Secret Sharing and Key Distribution", PhD thesis, Israel Institute of Technology, Technion, Haifa, Israel, **(1996)**.
[23] R. Canetti, S. Halevi and J. Katz, "A Forward-Secure Public-Key Encryption Scheme", Advances in Cryptology –Eurocrypt, vol. 2656 of LNCS. Springer, **(2003)**.
[24] D. Boneh and X. Boyen, "Efficient Selective-ID Secure Identity Based Encryption without Random Oracles", Advances in Cryptology – Eurocrypt, Springer, vol. 3027 of LNCS, **(2004)**, pp. 223-238.
[25] M. Hinek, S. Jiang, R. Safavi-Naini and S. Shahandashti, "Attribute-Based Encryption with Key Cloning Protection", International Journal of Applied Cryptography archive, vol. 2, no. 3, **(2012)**, pp. 250-270.

