How to Design Space Efficient Revocable IBE from Non-monotonic ABE

Huang Lin
Shanghai Jiao Tong University
University of Florida
linhuangsame@gmail.com

Zhenfu Cao
Shanghai Jiao Tong University
zfcao@sjtu.edu.cn

Yuguang Fang
University of Florida
fang@ece.ufl.edu

Muxin Zhou
Shanghai Jiao Tong University
muxin.zhou@gmail.com

Haojin Zhu
Shanghai Jiao Tong University
zhuhaojin@gmail.com

ABSTRACT

Since there always exists a possibility that some users' private keys are stolen or expired in practice, it is important for identity based encryption (IBE) system to provide a solution to revocation. The current most efficient revocable IBE system has a private key of size $\mathcal{O}(\log n)$ and update information of size $\mathcal{O}(r\log(\frac{n}{r}))$ where r is the number of revoked users. In this paper, we present a new revocable IBE system in which the private key only contains two group elements and the update information size is $\mathcal{O}(r)$. We show that the proposed constructions for the revocation mechanism are more efficient in terms of space cost and provide a generic methodology to transform a non-monotonic attribute based encryption into a revocable IBE. We also demonstrate how the proposed method can be employed to develop an efficient hierarchical revocable IBE system.

Categories and Subject Descriptors

E.3 [**Data**]: [Data Encryption]

General Terms

Algorithms, Security

Keywords

Revocable IBE, Non-monotonic, Attribute based encryption (ABE)

1. INTRODUCTION

Revocation is an important method in the setting of both the traditional public key encryption and identity based encryption (IBE) to fight against key compromise or expiration [1]. Since there are always some users whose private keys are either stolen or expired, the system has to provide some ways to revoke these keys to avoid any potential information leakage.

In the traditional public key encryption systems, there exists a public key infrastructure (PKI) responsible for handling necessary information used to verify the binding between a user's identities and its public key. An encryptor has to check the public key and the corresponding certificate for a receiver to make sure the binding is still valid before it carries out the encryption for the information intended for the receiver. Therefore, the PKI might publish some information such as the revocation list to inform the encryptor who can be ensured that the receiver is not on the revocation list. This is how the revocation problem is usually addressed in the traditional system [2, 3, 4]. However, this does not work well in the identity based encryption setting. The primitive idea of IBE came from the identity based signature first proposed by Shamir [1] to eliminate the verification of the binding between the identity and the public key. The encryptor only needs the public parameters and the identity of the receiver to complete the encryption steps under IBE system without any involvement of the private key generator (PKG), a trusted entity responsible for the private key generation and secure distribution. Therefore, this system does not provide any secure channel for the PKG to deliver the revocation list to the encryptor. When Boneh and Franklin proposed their first practical IBE solution [5], they also attempted to provide a solution for this revocation problem. Their basic idea is to renew the private key of each user in the system periodically, e.g., each week, and senders encrypt messages using the receiver's identities along with the current time period. However, it is easy to observe that this is not an efficient solution because the workload of the PKG would have been linearly dependent on the total number of the users in the system, which would be unbearable when the number of users is very large. To cut down the workload, several revocable IBE constructions based on a trusted mediator [6, 7] were proposed in the literature. However, these revocable IBE systems may not be desirable because the mediator has to be involved in the decryption steps. Recently, Yu et al. [8] proposed a revocable attribute based encryption scheme (ABE) which combines proxy re-encryption with the ABE. The trusted authority (TA) only needs to update the attribute master key according to the attribute revocation status in each time period, and issue proxy re-encryption key to the proxy servers. The proxy servers will then update secret keys for those unrevoked user attributes, and

ASIACCS'11, March 22-24, 2011, Hong Kong, China

also re-encrypt the ciphertext using the re-encryption key to make sure all the qualified and unrevoked users can successfully decrypt. Despite its technical novelty, the scheme still suffers from the inefficiency because the workload of the proxy servers will still be linearly dependent on the number of users.

In 2008, Boldyreva, Goyal and Kumar [9] proposed another revocable IBE system, called BGK system, with improved efficiency. Compared with the original revocation system by Boneh and Franklin with a $\mathcal{O}(n-r)$ key update complexity, their key update information is of size $\mathcal{O}(r\log(\frac{n}{r}))$ while the size of private key is $\mathcal{O}(\log n)$. The basic idea of their construction is to treat the identity and the time period as two independent attributes in the fuzzy identity based encryption [10]. The message is encrypted under these two attributes. In order to successfully open the encrypted message, the receiver should not only have the correct identity, but also the right time period. The published key update information aims to bind the unrevoked identities with the current time period. In other words, only those unrevoked users can use these information for the current time period to update their private keys. In order to reduce the workload of the PKG, they adopt the binary tree structure as their underlying tool. Therefore, although the key update information size is reduced to $\mathcal{O}(r\log(\frac{n}{r}))$, the private key size has to be $\mathcal{O}(\log n)$.

Libert and Vergnaud [11] proposed yet another similar scheme with similar efficiency to that of Boldyreva et al.'s construction [9] while their construction is proven secure under an adaptive model.

1.1 Our Contribution

As we observe, an important feature shared by the current two efficient revocable IBE systems [9, 11] is that a user has to get the key update information ku_t for the current time period as an input to generate the decryption key. In other words, if a user ω at the t'-th time period wants to decrypt a ciphertext generated at a past time period, say the t''-th ($t'' < t'$) time period, the update information for the t''-th time period should be accessible to the user. Besides, there is no way to guarantee that each user will update his/her decryption key whenever the update information is published especially if the update information is published frequently. This implies that the system has to store the update information for all the time periods in the system lifetime and make them publicly accessible. Although the space cost of the current revocable IBE schemes is low, it might not be desirable when the update information or the decryption keys piles up. Hence, it is very important to carefully investigate the space cost of the revocable IBE scheme more systematically, which is one of the major motivations of this paper.

In this paper, we propose a new revocable IBE scheme from non-monotonic attribute based encryption (ABE) scheme. The basic idea is to enforce a user to use a private key corresponding to a predicate using AND gate which connects time period and the negation of all the revoked identities. We use non-monotonic ABE as the underlying tool to guarantee that only the non-revoked users can successfully decrypt the ciphertext. The proposed scheme has a private key of constant size and update information of $\mathcal{O}(r)$ size. The security construction is based on the Decisional BDH assumption under selective-revocable-ID model as in [9]. It will be demonstrated that the proposed revocable IBE scheme is the most efficient one in terms of space cost. However, the decryption complexity in our scheme is dominated by $\mathcal{O}(r)$ group operations. Thus, the proposed scheme is especially suitable for the application scenarios with a small number of revocations, i.e., $r \ll n$, and fits better in applications in which the space cost, rather than the decryption efficiency, is a major concern. Moreover, the private storage requirement for each user can be significantly reduced because each user only needs to hold two group elements in our proposed scheme, and the storage requirement or transmission requirement for the system can also be reduced because the logarithmic factor is removed from the size of the published information. Thus, our scheme can be used in wireless sensor networks because the performance of our scheme is really close to the broadcast encryption scheme proposed in [12]. Besides, our proposed scheme also fits well to the systems where hybrid encryption or key encapsulation mechanism can be adapted because in this case the decryption efficiency is determined by the symmetric key decryption algorithm. Indeed, the optimal revocable IBE scheme might be obtained by combining the BGK system with our system, i.e., using our scheme when r is relatively small and using BGK when r exceeds a threshold.

Compared with the existing revocable IBE schemes using binary tree structure and fuzzy identity based encryption [9, 11], this paper constructs a revocable IBE scheme from attribute based encryption without any involvement of binary tree structure. We provide a generic methodology to design our revocable IBE scheme from the non-monotonic ABE scheme by using the existing two non-monotonic ABE schemes [13, 12]. We illustrate our approach by applying our design to the non-monotonic ABE proposed by Ostrovsky et al. [13].

The rest of this paper is organized as follows. In Section 2, we first introduce the basic complexity assumptions and definitions and present the security model for the revocable IBE scheme. In Section 3, we describe our revocable IBE scheme and highlight the corresponding methodology for our design. In last section, we draw several conclusions.

2. PRELIMINARIES

In this section, we first present some preliminaries we will use in this paper.

2.1 Decisional BDH assumption

The bilinear map (or pairing) is crucial to our design, some basic facts related to bilinear map are introduced here.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups of prime order p. Let g be a generator of $\mathbb{G}_1$ and $\hat{e}$ be a bilinear map, $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$. The bilinear map $\hat{e}$ has the following properties:

1. Bilinearity: for all $u, v \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_p$, we have $\hat{e}(u^a, v^b) = \hat{e}(u, v)^{ab}$.
2. Non degeneracy: $\hat{e}(g, g) \neq 1$.

$\mathbb{G}_1$ is a bilinear group if the group operation in $\mathbb{G}_1$ and the bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ are both efficiently computable. Note that the map $\hat{e}$ is symmetric since $\hat{e}(g^a, g^b) = \hat{e}(g, g)^{ab} = \hat{e}(g^b, g^a)$.

The security proof of the proposed scheme relies on the DBDH assumption, which is given below.

Definition 1. Decisional Bilinear Diffie-Hellman (DBDH) Assumption . Let $z_1, z_2, z_3, z \leftarrow \mathbb{Z}_p$ be chosen randomly, $\mathbb{G}_1$ be a cyclic group. The generator of this group is g. The Decisional BDH assumption is that no probabilistic polynomial time algorithm $\mathcal{B}$ can distinguish the tuple $(g_1 = g^{z_1}, g_2 = g^{z_2}, g_3 = g^{z_3}, \hat{e}(g,g)^{z_1 z_2 z_3})$ from the tuple $(g_1 = g^{z_1}, g_2 = g^{z_2}, g_3 = g^{z_3}, \hat{e}(g,g)^{z})$ with greater than a negligible advantage. The advantage of $\mathcal{A}$ is $|Pr[\mathcal{A}(g_1, g_2, g_3, \hat{e}(g,g)^{z_1 z_2 z_3}) = 1]$-$Pr[\mathcal{A}(g_1, g_2, g_3, \hat{e}(g,g)^{z})] = 1|$, where the probability is taken over the random choice of the generator g, the random choice of a, b, c, z in $\mathbb{Z}_p$, and the random bits assumed by $\mathcal{B}$.

2.2 Definition and Security Model

Definition The definition of revocable IBE ($\mathcal{RIBE}$) is shown as follow: A revocable IBE scheme is defined by seven algorithms which have their associated message space $\mathbf{M}$, identity space $\mathbf{I}$ and time space $\mathbf{T}$. Key authority maintains a revocation identity list rl recording all the revoked identity set R and state st. In what follows, an algorithm is called stateful only if it updates rl or st. The life time of the system is divided into periods during which update information is published.

The stateful **setup** algorithm $\mathcal{S}$ (run by the key authority) takes the input security parameter 1^{κ} and the number of users n, and outputs the public parameters PK, the master key MK, the revocation list rl (initially empty) and the state st.
The stateful **private key generation** algorithm $\mathcal{SK}$ (run by the key authority) takes the input public parameters PK, the master key MK, the identity $\omega \in \mathbf{I}$ and the state st, and outputs the private key SK_{ω} and an updated state st.
The **key update** generation algorithm $\mathcal{KU}$ (run by the key authority) takes the input public parameters PK, the master key MK, the key update time $t \in \mathbf{T}$, the revocation list rl and the state st, and outputs the key update KU_t.
The **encryption** algorithm $\mathcal{E}$ (run by the sender) takes the input public parameters PK, the identity $\omega \in \mathbf{I}$, the encryption time $t \in \mathbf{T}$ and the message $m \in \mathbf{M}$, and outputs the ciphertext c.
The deterministic **decryption** algorithm $\mathcal{D}$ (run by the receiver) takes the input private key SK_{ω}, the key update information KU_t and the ciphertext c, and outputs a message $m \in \mathbf{M}$ or, a special symbol $\perp$ indicating that the ciphertext is invalid.
The stateful **revocation** algorithm $\mathcal{R}$ (run by the key authority) takes the input identity to be revoked $\omega \in \mathbf{I}$, the revocation time $t \in \mathbf{T}$, the revocation list rl and the state st, and outputs an updated revocation list rl.

The consistency condition requires that for all $\kappa \in \mathbb{N}$ and polynomials (in κ) n, all PK and MK output by the setup algorithm $\mathcal{S}$, all $m \in \mathbf{M}$, $\omega \in \mathbf{I}$, $t \in \mathbf{T}$ and all possible valid states st and revocation lists rl, if identity ω was not revoked before or, at time t then the following experiment returns 1 with probability 1: $(SK_{\omega}, st) \overset{r}{\leftarrow} \mathcal{SK}(PK, MK, \omega, st)$; $KU_t \overset{r}{\leftarrow} \mathcal{KU}(PK, MK, t, rl, st)$, $c \overset{r}{\leftarrow} \mathcal{E}(PK, \omega, t, m)$; If $\mathcal{D}(SK_{\omega}, KU_t, c) = m$, then return 1, else return 0.

Security model In the following game, we define the selective-revocable-ID security for revocable IBE schemes. The security model imitates the definition of the selective-ID security for traditional IBE scheme while taking possible revocation into account. At the beginning of the game, the adversary declares the challenge time and identity. The adversary is able to revoke users of its choices (including the challenge identity) at any period of time and see all the key update information. The adversary is also allowed to see the private key of users including the challenge identity but when it was revoked prior or at the challenge time.

In the following, we only restrict in the security against the chosen plaintext attack and the definition for chosen-ciphertext attack can be found in [9], which is omitted in this paper.

The adversary first outputs the challenge identity and time, and also some information *state* it wants to preserve. Later it is given access to three oracles that correspond to the algorithms of the scheme. The oracles share state as are defined below:

- The *private key generation* oracle $\mathcal{SK}(\cdot)$ takes the input identity ω and runs $\mathcal{SK}(pk, mk, \omega, st)$ to return the private key sk_{ω}.
- The *key update* oracle $\mathcal{KU}(\cdot)$ takes the input time t and runs $\mathcal{KU}(pk, mk, t, rl, st)$ to return the key update ku_t.
- The *revocation* oracle $\mathcal{R}(\cdot, \cdot)$ takes input the identity ω and time t and runs $\mathcal{R}(\omega, t, rl, st)$ to the update rl.

The following two restrictions about the aforementioned oracles must hold: first, $\mathcal{KU}(\cdot)$ and $\mathcal{R}(\cdot, \cdot)$ can be queried on the time greater than or equal to the time of all previous queries, i.e., the adversary is allowed to query only in non-decreasing order of time. Also, the oracle $\mathcal{R}(\cdot, \cdot)$ cannot be queried on time t if $\mathcal{KU}(\cdot)$ was queried on t. Second, if $\mathcal{SK}(\cdot)$ was queried on the identity ω^*, then $\mathcal{R}(\cdot, \cdot)$ must be queried on ω^*, t for any $t \leq t^*$.

For adversary $\mathcal{A}$ and the number of users n, we define the following experiments for simulator construction:

- **Init**: The adversary declares the challenge identity ω^* and time t^*, that he/she wishes to be challenged upon.
- **Setup**: The challenger runs the *Setup* algorithm and gives the public parameters to the adversary.
- **Phase 1**: The adversary is allowed to issue the aforementioned three oracles *private key generation* oracle, *key update* oracle, and *revocation* oracle.
- **Challenge**: The adversary submits two equal length message $m_0, m_1 \in \mathcal{M}$. The challenger flips a coin $b \in \{0, 1\}$, and runs $\mathcal{E}(pk, \omega^*, t^*, m_b)$ to generate the challenge ciphertext c^* and pass it to the adversary.
- **Phase 2**: Phase 1 is repeated.
- **Guess**: The adversary outputs a guess b' of b.

The advantage of an adversary $\mathcal{A}$ in this game is defined as $Pr[b' = b] - \frac{1}{2}$.

Definition 2. The revocable IBE scheme is said to be sRID-CPA secure if all polynomial time adversaries have at most a negligible advantage in the above game.

3. $\mathcal{RIBE}$ FROM THE NON-MONOTONIC ABE

This section is divided into two parts: we will first introduce the primitive idea, and the concrete scheme based on the non-monotonic ABE proposed by Ostrovsky et al.

3.1 A Transformation based on Non-monotonic ABE

As mentioned before, our basic idea is to implement a revocable IBE scheme from a non-monotonic ABE scheme without any involvement with binary tree structure. Although the *NOT* gate in a non-monotonic ABE seems naturally to give the exclusion to some users, there is no straightforward path to implement this.

In our proposed scheme, the update information corresponds to a predicate $t \bigwedge_{i=1}^{r} \overline{\omega^{(i)}}$ where the identity set $\{\omega^{(i)}\}_{i=1}^{r}$ is the revocation set at the time period t. Here, $\overline{\omega^{(i)}}$ denotes the negation of $\omega^{(i)}$ in a non-monotonic ABE scheme, which means the respective identity is revoked in the revocable IBE scheme.

The system will enforce each user ω to use a private key corresponding to a predicate $t \bigwedge_{i=1}^{r} \overline{\omega^{(i)}} \bigwedge \omega$ to decrypt a ciphertext. A message for user ω will be encrypted under an attribute set $\{\omega, t\}$. If a user ω is revoked at time t, then ω must belong to the identity set $\{\omega^{(i)}\}_{i=1}^{r}$. In this case, this specific user will be forced to use a private key for a predicate $t \bigwedge_{i=1}^{j-1} \overline{\omega^{(i)}} \bigwedge \overline{\omega} \bigwedge_{i=j+1}^{r} \overline{\omega^{(i)}} \bigwedge \omega$ to decrypt the ciphertext. A key policy non-monotonic attribute based encryption(ABE) scheme[1] can be used to ensure that the decryption fails because the attribute set $\{\omega, t\}$ does not satisfy the predicate $t \bigwedge_{j=1}^{i-1} \overline{\omega^{(j)}} \bigwedge \overline{\omega} \bigwedge_{j=i+1}^{r} \overline{\omega^{(j)}} \bigwedge \omega$. The decryption is also guaranteed to be successful if the attribute set $\{\omega, t\}$ satisfies the respective predicate $t \bigwedge_{i=1}^{r} \overline{\omega^{(i)}} \bigwedge \omega$ when ω is unrevoked at the time t and hence is not in the revoked set $\{\omega^{(i)}\}_{i=1}^{r}$.

The tricky problem left is how we could force a user ω to use a private key corresponding to such a predicate, i.e., $t \bigwedge_{i=1}^{r} \overline{\omega^{(i)}} \bigwedge \omega$ in the decryption steps. In our design, we divide the system master key $MK = \alpha$ into two parts $MK_1 = \lambda$ and $MK_2 = \alpha - \lambda$, i.e., $MK = MK_1 + MK_2$. MK_1 is used to distribute private key for the user identity ω, and MK_2 is used to generate update information, i.e., a private key corresponding to a predicate $t \bigwedge_{i=1}^{r} \overline{\omega^{(i)}}$. A message will be encrypted under a public key corresponding to the system master key MK, and hence the decryptor has to use both private key and key update information in order to successfully decipher a message. We observe that collusion attack is impossible because all the unrevoked users hold information on the same partial master key MK_1 in their own private keys, and therefore, there is no way to gain any useful information on the system master key α by collusion.

We notice that our proposed transformation from a non-monotonic ABE to a $\mathcal{RIBE}$ is almost fully generic except that we require the two sub master keys for private key distribution and key update be combined as one system master key. Both of the two recently proposed non-monotonic ABE schemes [13, 14] can be used as our underlying scheme to design our secure $\mathcal{RIBE}$. Here, we use the OSW non-monotonic ABE [13] as an example to demonstrate how our transformation method works.

3.2 $\mathcal{RIBE}$ from the OSW Non-monotonic ABE

Let $\mathbb{G}_1$ be a bilinear group of prime order p, and let g be a generator of $\mathbb{G}_1$. In addition, let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ denote the bilinear map (the pairing function). A security parameter κ determines the size of the groups. We also define the Lagrange coefficient $\Delta_{i,J}$ for $i \in \mathbb{Z}_p$ and a set J of elements in $\mathbb{Z}_p : \Delta_{i,J}(x) = \prod_{j \in J, j \neq i} \frac{x-j}{i-j}$.

Setup $\mathcal{S}(1^{\kappa}, n)$: Randomly choose α, β uniformly from $\mathbb{Z}_p^*$. Set $g_1 = g^{\alpha}$ and $g_2 = g^{\beta}$. Randomly select two polynomials $h(x)$ and $q(x)$ of degree two with $q(0) = \beta$.

The public parameters PK contains $(g,\ g_1,\ g_2 = g^{q(0)},\ g^{q(1)},\ g^{q(2)};\ g^{h(0)},\ g^{h(1)},\ g^{h(2)})$. The master key is $MK = \alpha$. The public parameters define two publicly computable functions $T, V : \mathbb{Z}_p \rightarrow \mathbb{G}$, which are given as $T(x) = g_2^{x^2} \cdot g^{h(x)}$ and $V(x) = g^{q(x)}$. Besides, randomly choose λ uniformly from $\mathbb{Z}_p$ and set $MK_1 = \lambda$ and $MK_2 = \alpha - \lambda$.

Private Key Generation $\mathcal{SK}(MK_1,\ \omega,\ PK)$: Randomly choose ρ_ω uniformly from $\mathbb{Z}_p$. The algorithm outputs a private key for identity ω as
$SK_\omega = \left(D_\omega^{(1)}, D_\omega^{(2)}\right) = \left(g_2^{MK_1} \cdot T(\omega)^{\rho_\omega}, g^{\rho_\omega}\right) = (g_2^{\lambda} \cdot T(\omega)^{\rho_\omega}, g^{\rho_\omega})$

Key Update Generation $\mathcal{KU}(MK_2,\ PK,\ t,\ rl,\ st)$: This algorithm outputs a private key corresponding to the access structure $t \bigwedge_{i=1}^{r} \overline{\omega^{(i)}}$. For each revoked user $\omega^{(i)}, i \in [1, r]$, select two random values $\lambda_i, \rho_i \in (\mathbb{Z}_p)^2$ and publish the respective update information as

$$D_i = (D_i^{(3)}, D_i^{(4)}, D_i^{(5)}) = (g_2^{\lambda_i + \rho_i}, V(\omega^{(i)})^{\rho_i}, g^{\rho_i})$$

For the update time t, select $\rho_t \xleftarrow{r} \mathbb{Z}_p$ and publish the update information as
$D_t = (D_t^{(1)}, D_t^{(2)}) = (g_2^{MK_2 - \sum_{i=1}^{r} \lambda_i} \cdot T(t)^{\rho_t},\ g^{\rho_t})$
$= (g_2^{\alpha - \lambda - \sum_{i=1}^{r} \lambda_i} \cdot T(t)^{\rho_t},\ g^{\rho_t})$
Return the update information as
$KU_t = \left(\{(\omega^{(i)}, D_i)\}_{i=1}^{r}, D_t\right)$.

Encryption $\mathcal{E}(PK, \omega, t, M)$: Randomly choose s from $\mathbb{Z}_p$ and generate the ciphertext as
$C = \Big(E^{(1)} = M\hat{e}(g_1,\ g_2)^s,\ E^{(2)} = g^s,\ E_\omega^{(3)} = T(\omega)^s,$
$E_t^{(3)} = T(t)^s,\ E_\omega^{(4)} = V(\omega)^s,\ E_t^{(4)} = V(t)^s\Big)$

Decryption $\mathcal{D}(SK_\omega, KU_t, PK, C)$: If ω is not revoked at time t, then we have $\omega \notin \{\omega^{(i)}\}_{i=1}^{r}$. Using $D_\omega^{(1)}$ and $D_\omega^{(2)}$ from the secret key SK_ω of ω, compute the partial decryption as
$Z_\omega = \hat{e}(D_\omega^{(1)},\ E^{(2)}) / \hat{e}(D_\omega^{(2)},\ E_\omega^{(3)}) = \hat{e}(g_2^{\lambda} \cdot T(\omega)^{\rho_\omega},\ g^s) / \hat{e}(g^{\rho_\omega},$
$T(\omega)^s) = \hat{e}(g_2,\ g)^{s\lambda}$

Using $D_t^{(1)}$ and $D_t^{(2)}$ from the update information D_t corresponding to the update time t, compute the partial decryption for the update time t as
$Z_t = \frac{\hat{e}(D_t^{(1)},\ E^{(2)})}{\hat{e}(D_t^{(2)},\ E_t^{(3)})} = \frac{\hat{e}(g_2^{\alpha - \lambda - \sum_{i=1}^{r} \lambda_i} \cdot T(t)^{\rho_t},\ g^s)}{\hat{e}(g^{\rho_t},\ T(t)^s)}$
$= \hat{e}(g_2,\ g)^{s(\alpha - \lambda - \sum_{i=1}^{r} \lambda_i)}$

For each revoked user $\omega^{(i)}, i \in [1, r]$, compute Lagrangian coefficients $\{\sigma_x\}_{x \in \{\omega, t, \omega^{(i)}\}}$ such that $\sum_{x \in \{\omega, t, \omega^{(i)}\}} \sigma_x q(x) = q(0) = \beta$. Using the update information $D_i^{(3)}$, $D_i^{(4)}$, $D_i^{(5)}$, compute the corresponding partial decryption as

[1] See [13] for the concrete definition of key policy non-monotonic ABE

$$Z_i = \frac{\hat{e}(D_i^{(3)},\ E^{(2)})}{\hat{e}\Big(D_i^{(5)},\ \prod_{x\in\{\omega,t\}}\Big(E_x^{(4)}\Big)^{\sigma_x}\Big)\cdot\hat{e}\Big(D_i^{(4)},\ E^{(2)}\Big)^{\sigma_{\omega^{(i)}}}}$$

$$= \frac{\hat{e}(g_2^{\lambda_i+\rho_i},\ g^s)}{\hat{e}(g^{\rho_i},\ V(\omega)^{s\sigma_\omega}V(t)^{s\sigma_t})\cdot\hat{e}(V(\omega^{(i)})^{\rho_i},\ g^s)^{\sigma_{\omega^{(i)}}}}$$

$$= \frac{\hat{e}(g_2^{\lambda_i+\rho_i},\ g^s)}{\hat{e}(g^{\rho_i},\ g^{s[\sigma_\omega q(\omega)+\sigma_t q(t)]})\cdot\hat{e}(g^{\rho_i\sigma_{\omega^{(i)}}q(\omega^{(i)})},\ g^s)}$$

$$= \frac{\hat{e}(g_2^{\lambda_i+\rho_i},\ g^s)}{\hat{e}(g^{\rho_i},\ g^{sq(0)})}$$

$$= \hat{e}(g_2^{\lambda_i},\ g^s)$$

Finally, compute

$$\hat{e}(g_2,\ g)^{s\lambda}\cdot\hat{e}(g_2,\ g)^{s(\alpha-\lambda-\sum_{i=1}^{r}\lambda_i)}\cdot\prod_{i=1}^{r}\hat{e}(g_2,\ g)^{s\lambda_i}$$

$$= \hat{e}(g_2,\ g_1)^s \text{ and } \frac{E^{(1)}}{\hat{e}(g_2,\ g_1)^s} = M.$$

In the above construction, the private key only contains two group elements while the update information contains $3r+2$ group elements. This justifies our efficiency claims in the introduction.

The security of our proposed scheme can be summarized in the following theorem. The proof of this theorem will be provided in the final version.

Theorem 1. *If an adversary can break the proposed scheme in the sRID model, then a simulator can be constructed to play the Decisional BDH game with a non-negligible advantage.*

As a remark, the chosen ciphertext attack (CCA) secure construction can be given by applying the same methodology to the underlying CCA secure non-monotonic ABE scheme, which will be omitted here. Besides, the space cost could be further saved if random oracle is adopted.

4. CONCLUSIONS

This paper presents a revocable IBE scheme in which the private key only contains two group elements. The update information size depends on the number of the revoked users in the system. It has been shown that the proposed scheme is space efficient and provides a generic method to transform a non-monotonic attribute based encryption scheme into a revocable IBE scheme.

5. ACKNOWLEDGEMENT

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 60970110, 60972034 and the State Key Program of National Natural Science of China(Grant No. 61033014).

6. REFERENCES

[1] Adi Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO*, pages 47–53, 1984.

[2] William Aiello, Sachin Lodha, and Rafail Ostrovsky. Fast digital identity revocation (extended abstract). In *CRYPTO*, pages 137–152, 1998.

[3] Craig Gentry. Certificate-based encryption and the certificate revocation problem. In *EUROCRYPT*, pages 272–293, 2003.

[4] Vipul Goyal. Certificate revocation using fine grained certificate space partitioning. In *Financial Cryptography*, pages 247–259, 2007.

[5] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. In *CRYPTO*, pages 213–229, 2001.

[6] Xuhua Ding and Gene Tsudik. Simple identity-based cryptography with mediated rsa. In *CT-RSA*, pages 193–210, 2003.

[7] D. Boneh, X. Ding, G. Tsudik, and M. Wong. A method for fast revocation of public key certificates and security capabilities. In *10th USENIX Security Symposium*, pages 297–308, 2001.

[8] Shucheng Yu, Cong Wang, Kui Ren, and Wenjing Lou. Attribute based data sharing with attribute revocation. In *ASIACCS*, pages 261–270, 2010.

[9] Alexandra Boldyreva, Vipul Goyal, and Virendra Kumar. Identity-based encryption with efficient revocation. In *ACM Conference on Computer and Communications Security*, pages 417–426, 2008.

[10] Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *EUROCRYPT*, pages 457–473, 2005.

[11] Benoît Libert and Damien Vergnaud. Adaptive-id secure revocable identity-based encryption. In *CT-RSA*, pages 1–15, 2009.

[12] Amit Sahai and Brent Waters. Revocation systems with very small private keys. *http://eprint.iacr.org/2008/309.*

[13] Rafail Ostrovsky, Amit Sahai, and Brent Waters. Attribute-based encryption with non-monotonic access structures. In *ACM Conference on Computer and Communications Security*, pages 195–203, 2007.

[14] Brent Waters. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. *http://eprint.iacr.org/2008/290.*