Computers, Materials & Continua CMC, vol.63, no.2, pp.633-647, 2020

Key-Private Identity-Based Proxy Re-Encryption

Chunpeng Ge[1, *], Jinyue Xia[2] and Liming Fang[1]

Abstract: An identity-based proxy re-encryption scheme (IB-PRE) allows a semi-trusted proxy to convert an encryption under one identity to another without revealing the underlying message. Due to the fact that the proxy was semi-trusted, it should place as little trust as necessary to allow it to perform the translations. In some applications such as distributed file system, it demands the adversary cannot identify the sender and recipient's identities. However, none of the exiting IB-PRE schemes satisfy this requirement. In this work, we first define the security model of key-private IB-PRE. Finally, we propose the first key-private IB-PRE scheme. Our scheme is chosen plaintext secure (CPA) and collusion resistant in the standard model.

Keywords: Proxy re-encryption, identity-based proxy re-encryption, key-private, collusion resistant.

1 Introduction

With the rapid development of cloud computing, more and more people prefer to outsource their personal data on the cloud due to the low cost of data maintenance. The privacy of data becomes a crucial problem when data is outsourced in the cloud. In some scenarios, the data owner wants to share his data with others without revealing the data to the untrusted cloud server. However, as user's data is encrypted with his own public key, the cloud server cannot decrypt the ciphertext and transmit the designate sharing user. Thus, a mechanism that enables sharing the outsourced encrypted data while preserving the privacy in untrusted cloud server is need.

Proxy re-encryption (PRE), first introduced by Blaze et al. [Blaze, Bleumer and Strauss (1998)], enables a semi-trusted proxy to transform a ciphertext from one key to another of the same message without relying on trusted parties. In a proxy re-encryption scheme, the proxy only needs a re-encryption key to convert the ciphertext without learning any information of the underlying message. In some applications such as distributed file systems [Ateniese, Fu, Green et al. (2005)], in addition to hiding the contents of files, it is also needed to hide the recipient's identity form the proxy. To capture this property,

[1] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, 210000, China.

[2] IBM, Research Triangle Park, North Carolina, 27709, USA.

[*] Corresponding Author: Chunpeng Ge. Email: gecp@nuaa.edu.cn.

Received: 25 January 2019; Accepted: 14 July 2019.

Ateniese et al. [Ateniese, Benson and Hohenberger (2009)] introduced the notion of key-private proxy encryption. In a key-private PRE scheme, it is impossible for the proxy and a set of colluding users to reveal the recipient's identity from a re-encryption key. However, their scheme only achieves CPA-security. In 2011, Shao et al. [Shao, Liu, Wei et al. (2011)] proposed a single-use unidirectional proxy re-encryption, which is anonymous and CCA-secure in the random oracle model. Following their work, Tang et al. [Tang, Lian, Zhao et al. (2018)] proposed a proxy re-encryption scheme with keyword search functionality.

To extend the notion of proxy re-encryption to the setting of Identity Based Encryption (IBE), Green et al. [Green and Ateniese (2007)] proposed the first identity-based proxy re-encryption (IB-PRE). In an IB-PRE scheme, the proxy can convert a ciphertext encrypted under Alice's identity into one encrypted under Bob's identity. They proposed two IB-PRE schemes, which are both secure in the random oracle model. Chun et al. [Chu and Tzeng (2007)] introduced two IB-PRE schemes, which are both secure in the standard model based on Waters IBE scheme [Water (2005)]. However, both above two schemes are not collusion resistant. In a collusion resistant IB-PRE scheme, the proxy colluding with delegatees is able to only decrypt the ciphertexts under the delegator's identity but cannot obtain the delegator's private key. Shao et al. [Shao and Cao (2012)] proposed a multi-use unidirectional IB-PRE which is both CCA-secure and collusion resistant. They presented a conversion from a strongly CPA-secure no-anonymous hierarchical identity-based encryption to a CCA-secure and collusion resistant multi-use unidirectional IB-PRE. To capture a fine-grained control over the delegation, Liang et al. [Liang, Liu, Tan et al. (2012)] introduced the notion of identity based conditional proxy re-encryption, in which only ciphertexts satisfying a special condition can be convert from Alice's identity to Bob's identity.

Unfortunately, none of the above IB-PRE schemes achieve the key-private property. This work focuses on filling such a gap. As far as we know, this is the first solution for key-private IB-PRE. We here compare our scheme with previous IB-PRE schemes in terms of security model in Tab. 1.

Table 1: Security comparison

Scheme	Key private?	Collusion resistant?	Without RO?
IB-PRE [Green and Ateniese (2007)]	No	No	No
IB-PRE [Chu and Tzeng (2007)]	No	No	Yes
IB-PRE [Shao and Cao (2012)]	No	Yes	Yes
IB-PRE [Liang, Liu, Tan et al. (2012)]	No	Yes	Yes
Our IB-PRE	Yes	Yes	Yes

1.1 Our contribution

In this paper, we first formulate the security model of key-private identity-based proxy re-encryption. Our security model considers not only the privacy of the content but also the privacy of the identities for the original ciphertext, re-encrypted ciphertext and the re-encryption key. Finally, we propose the first key-private identity-based proxy re-encryption, which is CPA-secure and collusion resistant under the truncated q decisional Diffie-Hellman exponent assumption (q-DDHE) without random oracles.

1.2 Roadmap

The rest of the paper is organized as follows. We first provide the basic primitives and our security model for key-private identity-based proxy re-encryption in Section 2. In Section 3, we present our key-private IB-PRE scheme and give the security proofs. Finally, we conclude the paper in Section 4.

2 Preliminaries

2.1 Negligible function

A function $\varepsilon(n): N \to R$ is said to be negligible if for all positive integer $c \in N$, there exists a $n_c \in N$ such that $\varepsilon(n) < n_c$ for all $n > n_c$.

2.2 Bilinear map

Let G and G_T be two multiplicative cyclic groups with the same prime order p, and g be a generator of G. A bilinear pairing is a map $e: G \times G \to G_T$ with the following properties [Boneh and Franklin (2001); Boneh and Boyen (2004)]:

(1) $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$ for all a, b are randomly chosen from Z_p^* and $g_1, g_2 \in G$.

(2) $e(g, g) \neq 1$.

(3) There is an efficient algorithm to compute $e(g_1, g_2)$ for all $g_1, g_2 \in G$.

2.3 Complexity assumption

The security of our system is based on a complexity assumption that we call the truncated q decisional Diffie-Hellman exponent assumption. The q-DDHE assumption is as below: Given a vector of $q+2$ elements $(g, g^{\alpha}, g^{(\alpha^2)}, \ldots, g^{(\alpha^q)}, T) \in G^{q+2}$ as input, it is hard to distinguish $T = g^{(\alpha^{q+1})}$ from a random value in G. Formally, for all probability polynomial time adversaries A, the following probability is negligible:

$$| \; Pr[\alpha, r \leftarrow Z_p^*; \quad T_0 = g^{(\alpha^{q+1})}; \quad T_1 = g^r; \quad z \in \{0,1\};$$

$$z' \leftarrow A(g, g^{\alpha}, g^{(\alpha^2)}, \ldots, g^{(\alpha^q)}, T_z): \quad z = z'] - \frac{1}{2}.$$

Now, we point that our complexity assumption is not easier than the decision version of truncated q-ABDHE assumption [Gentry (2006)] which is as follows: On given a vector of $q+4$ elements

$$(g', g'^{(\alpha^{q+2})}, g, g^{\alpha}, g^{(\alpha^2)}, \ldots, g^{(\alpha^q)}, T) \in G^{q+3} \times G_T,$$

it is hard to decide whether $T = e(g', g)^{\alpha^{q+1}}$ or a random element in G_T.

2.4 Identity-based proxy re-encryption

In this subsection, we will provide the definition of identity-based proxy re-encryption.

Definition 1 (identity-based proxy re-encryption). A non-interactive identity-based proxy re-encryption consists of the following algorithms [Green and Ateniese (2007); Chu and Tzeng (2007)]:

- Setup (λ): The Setup algorithm is run by the private key generator(PKG), on input a security parameter λ, the global public parameters PP and mast secret key msk are outputted.
- KeyGen (msk, id): On input the mast secret key msk and an identity id, output the private key d_{id} for identity id.
- Encrypt (id, m): On input an identity id and a message $m \in M$, output the ciphertext C_{id}. Here M denotes the message space.
- RKeyGen (d_{id_i}, id_i, id_j): On input identities id_i, id_j and the secret key d_{id_i}, output the re-encryption key $rk_{id_i \to id_j}$.
- ReEncrypt ($C_{id_i}, rk_{id_i \to id_j}$): On input a re-encryption key $d_{id_i \to id_j}$ and a ciphertext C_{id_i} corresponding to identity id_i, output the re-encrypted ciphertext C_{id_j}.
- Decrypt (C_{id}, d_{id}): On input a private key d_{id} and a ciphertext C_{id}, output the plaintext m or an error symbol $\perp$.

Note that we omit the global parameters PP as the other algorithms' input for simplicity. The correctness of key-private IBPRE means that, for any message $m \in M$, any $d_{id_i} \leftarrow KeyGen(msk, id_i)$, $d_{id_j} \leftarrow KeyGen(msk, id_j)$ any $rk_{id_i \to id_j} \leftarrow RKeyGen(d_{id_i}, id_i, id_j)$, we have $Pr[Decrypt(C_{id_i}, d_{id_i}) = m] = 1$ and

$$Pr[Decrypt(d_{id_j}, ReEncrypt(C_{id_i}, rk_{id_i \to id_j}) = m] = 1$$

2.5 Security notion

Before introducing our security model, we first analyze what security level is needed in a key-private IBPRE scheme. As defined in Green et al. [Green and Ateniese (2007)], a key-private IBPRE should first captures the privacy of the context and the delegator's secret key. At the meanwhile, as defined in Ateniese et al. [Ateniese, Beson and Hohenberger (2009)], the privacy of the identities should also be protected. Formally, the security can be divided into the following situations:

(1). Indistinguishability of encryptions for the original ciphertext. In this security game, the adversary cannot reveal the underlying context from the original ciphertext;

(2). Indistinguishability of encryptions for the re-encrypted ciphertext. In this security game, the adversary cannot reveal the underlying context from the encrypted ciphertext;

(3). Delegator's key security. In this security game, the proxy colluding with a set of delegatees cannot reveal the delegator's private key;

(4). Indistinguishability of keys for the original ciphertext. In this security game, the adversary cannot reveal the recipient's identity from the original ciphertext;

(5). Indistinguishability of keys for the re-encrypted ciphertext. In this security game, the adversary cannot reveal the recipient's identity from the re-encrypted ciphertext;

(6). Indistinguishability of keys for the re-encryption key. In this security game, the adversary cannot reveal the recipient's identity from the re-encryption key;

Remark:

(a). As described in Libert et al. [Libert and Vergnaud (2008)], the indistinguishability of encryptions for the re-encrypted ciphertext implies the delegator's key security. Hence in a key-private IBPRE scheme, it is only needed to consider the former one.

(b). As described in Green et al. [Green and Ateniese (2007)], the indistinguishability of keys for the re-encryption key implies the indistinguishability of keys for the re-encrypted ciphertext. Hence in a key-private IBPRE scheme, it is only needed to consider the former one.

(c). The Indistinguishability of encryptions for the original ciphertext is almost the same as the indistinguishability of keys for the original ciphertext expect except in the challenge phase. In the former challenge phase, the adversary is returned a challenge ciphertext $Encrypt(id_*, m_b)$, $b \in \{0,1\}$. While in the latter challenge phase, the adversary is returned a challenge ciphertext $Encrypt(id_b, m^*)$, $b \in \{0,1\}$. Like Gentry [Gentry (2006)], we can incorporate the indistinguishability of keys into the indistinguishability of encryptions through a simple modification. In the modified challenge phase, the adversary is given a challenge ciphertext $Encrypt(id_b, m_c)$, $b, c \in \{0,1\}$.

Next, we proposed our security model for a key-private IBPRE scheme. Our security model extends Green and Ateniese's security model [Green and Ateniese (2007)] not only in the form of key private, but also has many other advantages.

2.5.1 Indistinguishability of encryptions and keys under chosen-plaintext attack

The Indistinguishability of Encryptions and Keys under Chosen-Plaintext attack for the Original ciphertext (IE/IK-CPA-O) of key-private IBPRE captures the fact, it is impossible for an adversary to reveal the context and the recipient's identity from an original ciphertext under the CPA security. It is defined by the following game $Exp^{IE/IK-CPA-O}$ between a challenger C and an adversary A.

1. Setup. Run the Setup (λ) algorithm to get the (PP, msk), and give PP to A.

2. Query phase 1. A makes the following queries:

(a) Extract (id): run the KeyGen(msk, id) algorithm to get d_{id}, and returns d_{id} to.

(b) RKExtract(id_i, id_j): run the RKeyGen(d_{id_i}, id_i, id_j) algorithm to get $rk_{id_i \to id_j}$, and returns $rk_{id_i \to id_j}$ to A.

3. Challenge. Once A decides that phase 1 is over, it outputs two equal length message (m_0, m_1) and two challenge identities (id_0, id_1). The challenger C chooses two random bits $b, c \in \{0,1\}$ and sends the challenge ciphertext $C^* = Encrypt(id_b, m_c)$ to A. The restrictions is that, id_0, id_1 never appeared in the Extract(id) query.

4. Query phase 2. A continues making queries as in the Query phase 1, except making the Extract(id_0) and Extract(id_1) query.

5. Guess. A outputs the guess b', c'. The adversary wins if $b' = b$ and $c' = c$.

We say that a key-private IBPRE scheme is IE/IK-CPA-O secure, if the following probability is negligible for all probabilistic polynomial time adversary A:

$Adv_A^{IE/IK-CPA-O}(\lambda) = | Pr[b' = b \wedge c' = c] - 1/4 |$.

2.5.2 Indistinguishability of encryptions under chosen-plaintext attack

The Indistinguishability of Encryptions under Chosen-Plaintext attack for the Re-encrypted ciphertext (IE-CPA-R) of key-private IBPRE captures the fact, it is impossible for an adversary to reveal the context from a re-encrypted ciphertext under the CPA security. It is defined by the following game $Exp^{IE-CPA-R}$ between a challenger C and an adversary A.

1. Setup. Run the Setup(λ) algorithm to get the (PP, msk), and give PP to A.

2. Query phase 1. A makes the following queries:

(a) Extract(id): run the KeyGen(msk, id) algorithm to get d_{id}, and returns d_{id} to A.

(b) RKExtract(id_i, id_j): run the RKeyGen(d_{id_i}, id_i, id_j) algorithm to get $rk_{id_i \to id_j}$, and returns $rk_{id_i \to id_j}$ to A.

3. Challenge. Once A decides that phase 1 is over, it outputs two equal length message (m_0, m_1) and a challenge identity id^*. The challenger C chooses a random bit $b \in \{0,1\}$ and sends the challenge ciphertext $C^* = ReEncrypt(C_{id_i}, rk_{id_i \to id^*})$. The restrictions is that, id_i, id^* never appeared in the Extract(id) query.

4. Query phase 2. A continues making queries as in the Query phase 1, except making the Extract(id) query on identities id^*, id_i.

5. Guess. A outputs the guess b'. The adversary wins if $b' = b$.

We say that a key-private IBPRE scheme is IE-CPA-R secure, if the following probability is negligible for all probabilistic polynomial time adversary A:

$Adv_A^{IE-CPA-R}(\lambda) = | Pr[b' = b] - 1/2 |$.

2.5.3 Indistinguishability of encryptions under chosen-plaintext attack

The Indistinguishability of Keys under Chosen-Plaintext attack for the Re-encryption Key (IK-CPA-RK) of key-private IBPRE captures the fact, it is impossible for an adversary to reveal the identities from a re-encryption key under the CPA security. It is defined by the following game $Exp^{IK-CPA-RK}$ between a challenger C and adversary A.

1. Setup. Run the Setup (λ) algorithm to get the (PP,msk), and give PP to A.

2. Query phase 1. A makes the following queries:

(a) Extract(id): run the KeyGen(msk, id) algorithm to get d_{id}, and returns d_{id} to A.

(b) RKExtract(id_i, id_j): run the RKeyGen(d_{id_i}, id_i, id_j) algorithm to get $rk_{id_i \to id_j}$, and returns $rk_{id_i \to id_j}$ to A.

3. Challenge. Once A decides that phase 1 is over, it outputs two identities ID_I, ID_J, the challenge picks a random bit $b \in \{0,1\}$. If $b = 0$, then it sets rk^* as a random re-encryption key from the re-encryption key space; otherwise, it sets $rk^* = RKeyGen(d_{id_I}, id_I, id_J)$ and sends rk^* to the adversary A. The restrictions is that, id_I, id_J never appeared in the Extract(id) query.

4. Query phase 2. A continues making queries as in the Query phase 1, except making the Extract(id) query on identities id_I, id_J.

5. Guess. A outputs the guess b'. The adversary wins if $b' = b$.

We say that an key-private IBPRE scheme is IK-CPA-RK secure, if the following probability is negligible for all probabilistic polynomial time adversary A:

$Adv_A^{IK-CPA-RK}(\lambda) = |Pr[b' = b] - 1/2|$.

3 Our proposed key-private scheme

In this section, we first propose our key-private IBPRE scheme and then prove its CPA security, collusion resistance and key-private.

3.1 Technical difficulties and our approach

Before presenting our scheme, some important and necessary principles for designing key-private IBPRE should be mentioned. (i) the underlying identity based encryption must be key-private, otherwise when given the original ciphertext as challenge, the adversary can win the IE/IK-CPA-O trivially by using the underlying IBE; (ii) a key-private IBPRE must be collusion resistant, otherwise the adversary will win the IE-CPA-R game trivially. As in Chu et al. [Chu and Tzeng (2007)], let $Adv_A^{IK-CPA-RK}(\lambda) = |Pr[b' = b] - 1/2|$, the re-encryption key from identity id_1 to id_2 is set as $rk_{id_1 \to id_2} = (d_1 K^{-1}, d_2, R)$, where R is an encryption of K under identity id_2. When the proxy collude with delegates id_2, they can first recover K from the ciphertext R using the private of id_2. Then using K, they can recover id_1's private key (d_1, d_2). Also [Green and Ateniese (2007)] suffers from the same attack.

3.2 Our construction

Let G and G_T be bilinear group of prime order p, and g be a generator of G. Additionally, let $e: G \times G \to G_T$ denote the bilinear map. Our proposed scheme consists of the following algorithms:

- Setup(λ): Let λ be the security parameter, and (p, g, G, G_T, e) be the bilinear map parameters. The PKG picks random generators $g, h \in G$, random value $\alpha \in Z_p$ and a collusion resistant hash function $H: G_T \to Z_p$. It sets $g_1 = g^{\alpha} \in G$. The public parameters PP and master secret are set as:
  $PP = (g, g_1, h, H)$ $\quad msk = \alpha$.
- KeyGen(msk, id): To generate a private key for identity $id \in Z_p$, the PKG picks a random value $r_{id} \in Z_p$ and calculates $h_{id} = (hg^{-r_{id}})^{1/(\alpha - id)}$. Output the private key $d_{id} = (r_{id}, h_{id})$. If $\alpha = id$, the PKG aborts.
- Encrypt(id, m): On input an identity \$id\$ and a message $m \in G_T$, the sender picks a random value $s \in Z_p$ and sets

$C_1 = (g_1^s g^{-s \cdot id})$, $C_2 = g^s$, $C_3 = m \cdot e(g,h)^{-s}$.

Output the ciphertext $C = (C_1, C_2, C_3)$.

- RKeyGen(d_{id_i}, id_i, id_j): On input identities id_i, id_j and the secret key d_{id_i}, the re-encryption key $rk_{id_i \to id_j}$ is generated as follows:

(1). Choose random values $\theta \in G_T$ and $s' \in Z_p$, and compute $C_{1'} = (g_1^{s'} g^{-s' \cdot id_j})$, $C_{2'} = g^{s'}$, $C_{3'} = \theta \cdot e(g,h)^{-s'}$. Output the ciphertext $C' = (C_{1'}, C_{2'}, C_{3'})$.

(2). Choose a random value $t \in Z_p$, and sets $rk^{(1)}_{id_i \to id_j} = r_{id_i} \cdot H(\theta) + t$, $rk^{(2)}_{id_i \to id_j} = h^{H(\theta)}_{id_i}$, $rk^{(3)}_{id_i \to id_j} = g^t$, $rk^{(4)}_{id_i \to id_j} = C'$.

(3). Output the re-encryption key $rk_{id_i \to id_j} = (rk^{(1)}_{id_1 \to id_2}, rk^{(2)}_{id_1 \to id_2}, rk^{(3)}_{id_1 \to id_2}, rk^{(4)}_{id_1 \to id_2})$.

- ReEncrypt ($C_{id_i}, rk_{id_i \to id_j}$): On input a re-encryption key $rk_{id_i \to id_j} = (rk^{(1)}, rk^{(2)}, rk^{(3)}, rk^{(4)})$ and a ciphertext $C_{id_i} = (C_1, C_2, C_3)$ under identity id_i, the proxy proceeds as follows:

(1). Computers $\widetilde{C_3} = \dfrac{e(C_1, rk^{(2)}) \cdot e(g, C_2)^{rk^{(1)}}}{e(C_2, rk^{(3)})}$;

(2). Output the re-encrypted ciphertext $C_{id_i \to id_j} = (C_3, \widetilde{C_3}, rk^{(4)})$.

- Decrypt (C_{id}, d_{id}):

-If C_{id} is an original ciphertext, let $d_{id} = (r_{id}, h_{id})$ and $C_{id} = (C_1, C_2, C_3)$. Computer $m = C_3 \cdot e(C_1, h_{id}) \cdot e(g, C_2)^{r_{id}}$.

-If C_{id} is a re-encrypted ciphertext, let $C_{id} = (C_3, \widetilde{C_3}, rk^{(4)})$, where $rk^{(4)} = (C_{1'}, C_{2'}, C_{3'})$. Computer

$\theta = C_{3'} \cdot e(C_{1'}, h_{id}) \cdot e(g, C_{2'})^{r_{id}}$, $m = C_3 \cdot (\widetilde{C_3})^{1/H(\theta)}$.

3.3 Security of our scheme

Theorem 1. Our scheme is IE/IK-CPA-O secure without random oracles under the q-DDHE assumption.

Proof. Assume that there is an adversary A that can break the IE/IK-CPA-O security of our scheme with probability ε, then we can build an algorithm B that can solve the q-DDHE problem with probability ε', where

$$\varepsilon' \geq \frac{\varepsilon}{\dot{e}(1+q_e)}.$$

B inputs a q-DDHE instance $(g, A_1 = g^{\alpha}, A_2 = g^{\alpha^2}, \ldots, A_q = g^{\alpha^q}, T)$ and has to distinguish $T = A_{q+1} = g^{\alpha^{q+1}}$ from a random element in G.

Our approach to proving Theorem 1 closely follows the security proof for Gentry's scheme [Gentry 2006]. B first maintains the following tables which are initially empty.

- K^{List}: records the tuples (β, id, d_{id}), which are the information of secret keys;
- RK^{List}: records the tuples $(id_i, id_j, rk_{id_i \to id_j}, flag)$, which are the result of the queries to RKExtract(id_i, id_j), where $flag = 1$ denotes the re-encryption key is a valid one, and $flag = 0$ denotes the re-encryption key is a random value.

1. Setup: B generates a random polynomial $f(x) \in Z_p[x]$ of degree q. It sets $h = g^{f(\alpha)}$, computing h from $(g, A_1, \ldots, A_q)$. B also picks a collusion resistant hash function $H: G_T \to Z_p$. It sends the public key (g, A_1, h, H) to A. Note that, this assignment mens that, the mast secret key *msk* is α. Since g, α and $f(x)$ are chosen uniformly at random, h is uniformly random and this assignment has a distribution identical to that in the actual construction.

2. Query phase 1: A issues a series of queries to which B responds as follows:

(a). $Extract(id)$: B first searches K^{List}, if exists in K^{List}, returns d_{id} as the result. Otherwise, B generates a biased coin β so that $Pr[\beta = 1] = \delta$ for some δ that will be determined later.

- If $\beta = 0$, B outputs a random bit and aborts.

- If $\beta = 1$, if $id = \alpha$, we have that $Pr[id = \alpha] = 1/p$, B uses α to solve the q-DDHE problem. Else, let $F_{id}(x)$ denote the $q-1$ degree polynomial $(f(x) - f(id))/(x - id)$. B returns the private key $(r_{id}, h_{id}) = (f(id), g^{F_{id}(\alpha)})$ to the adversary and adds $(1, id, d_{id})$ to K^{List}. Note that, $g^{F_{id}(\alpha)} = g^{(f(\alpha)-f(id))/(\alpha - id)} = (hg^{-f(id)})^{1/(\alpha - id)}$, which is identical to the actual construction.

(b). $RKExtract(id_i, id_j)$: B first searches whether there is a tuple $(id_i, id_j, rk_{id_i \to id_j}, *)$ in K^{List}. If yes, B returns $rk_{id_i \to id_j}$ as the result, where * is the wildcard. Otherwise, B proceeds as follows:

- If $(1, id_i, d_{id_i})$ exits in K^{List}, B uses d_{id_i} to generater the re-encryption key $rk_{id_i \to id_j}$ visa algorithm $RKeyGen$ as in the real scheme. Returns the re-encryption key to A and adds $(id_i, id_j, rk_{id_i \to id_j}, 1)$ to K^{List}.

- Otherwise, B flips a biased coin β. If $\beta = 1$, B queries the $Extract(id_i)$ oracle to get d_{id_i}, and then generates $rk_{id_i \to id_j}$ visa algorithm $RKeyGen$ as in the real scheme. Returns the re-encryption key to A and adds $(1, id_i, d_{id_i})$ and $(id_i, id_j, rk_{id_i \to id_j}, 1)$ to K^{List} and RK^{List} respectively. If $\beta = 0$, B sets $rk^{(1)} = \sigma, rk^{(2)} = \phi_1, rk^{(3)} = \phi_2$ for randomly chosen $\sigma \in Z_p, \phi_1, \phi_2 \in G$. Then B constructs $rk^{(4)}$ to encrypt a random $\theta \in G_T$ as in the real scheme. B forwards the re-encryption key to A and adds $(id_i, id_j, rk_{id_i \to id_j}, 0)$ to RK^{List}.

3. Challenge: Once A decided that Query phase 1 is over, it outputs two equal length plaintexts (m_0, m_1) and two challenge identities (id_0, id_1), If $(1, id_0, d_{id_0})$ or $(1, id_1, d_{id_1})$ exit in K^{List}, B outputs random bits and aborts. Else if $\alpha \in \{id_0, id_1\}$, B uses α to solve the q-DDHE problem. Else B generates bits $b, c \in \{0,1\}$ and computes a private key (r_{id_b}, h_{id_b}) as in phase 1. Let $f_2(x) = x^{q+2}$ and $F_{2,id_b}(x) = (f_2(x) - f_2(id_b))/(x - id_b)$, B sets

$$C_1^* = g^{f_2(\alpha) - f_2(id_b)}, \qquad C_2^* = T \cdot \prod_{i=0}^{q} g^{F_{2,id_b,i} \cdot \alpha^i},$$

$$C_3^* = m_c / (e(C_1^*, h_{id_b}) \cdot e(g, C_2^*)^{r_{id_b}}),$$

where $F_{2,id_b,i}$ is the coefficient of x^i in $F_{2,id_b}(x)$. It sends (C_1^*, C_2^*, C_3^*) to A as the challenge ciphertext.

Note that, let $s^* = F_{2,id_b}(\alpha)$. If $T = A_{q+1} = g^{\alpha^{q+1}}$, we have:

$$C_1^* = g^{f_2(\alpha) - f_2(id_b)} = g^{F_{2,id_b}(\alpha) \cdot (\alpha - id_b)} = g_1^{s^*} g^{-s^* \cdot id_b},$$

$$C_2^* = T \cdot \prod_{i=0}^{q} g^{F_{2,id_b,i} \cdot \alpha^i} = g^{\alpha^{q+1}} \cdot \prod_{i=0}^{q} g^{F_{2,id_b,i} \cdot \alpha^i} = g^{s^*},$$

$$C_3^* = m_c / (e(C_1^*, h_{id_b}) \cdot e(g, C_2^*)^{r_{id_b}}) = m_c / (e(g_1^{s^*} g^{-s^* \cdot id_b}, h_{id_b}) \cdot e(g, g^{s^*})^{r_{id_b}}) = m_c \cdot e(g, h)^{-s^*}.$$

Thus, (C_1^*, C_2^*, C_3^*) is a valid ciphertext for (id_b, m_c).

4. Query phase 2: A continues making queries as in the query phase 1 with the restrictions described in the IE/IK-CPA-O game.

5. Guess: A outputs the guesses $b', c' \in \{0,1\}$. If $b' = b$ and $c' = c$, B outputs 1 meaning $T = g^{\alpha^{q+1}}$; else output 0 meaning T is a random value in G.

Probability analysis. If B does not abort, A's view is identical to the real scheme. We define Abort be the event of B's aborting during the simulation of $Extract$ query. Let

q_e denote the total number of $Extract$ queries, we have $Pr[\neg Abort] \geq \delta^{q_e} \cdot (\frac{p-1}{p})^{q_e} \triangleq \xi^{q_e} \geq \xi^{q_e}(1-\xi)$, which is maximized at $\delta_{opt} = \frac{q_e}{(1+q_e)}$. Using δ_{opt}, the probability $Pr[\neg Abort]$ is at least $1/\dot{e}(1+q_e)$, where $\dot{e}$ is the base of the nature logarithm. Therefore, we have $\varepsilon' \geq \frac{\varepsilon}{\dot{e}(1+q_e)}$.

This completes the proof of Theorem 1.

Theorem 2. Our scheme is IE-CPA-R secure without random oracles under the q-DDHE assumption.

Proof. Assume that there is an adversary A that can break the IE-CPA-R security of our scheme with probability ε, then we can build an algorithm B that can solve the q-DDHE problem with probability ε', where

$$\varepsilon' \geq \frac{\varepsilon}{\dot{e}(1+q_e)}.$$

B inputs a q-DDHE instance $(g, A_1 = g^{\alpha}, A_2 = g^{\alpha^2}, \ldots, A_q = g^{\alpha^q}, T)$ and has to distinguish $T = A_{q+1} = g^{\alpha^{q+1}}$ from a random element in G.

1. Setup: Same as the proof of Theorem 1.

2. Query phase 1: Same as the proof of Theorem 1.

3. Challenge: Once A decided that Query phase 1 is over, it outputs two equal length plaintexts (m_0, m_1) and an target identity id^*. If id^* exits in the former $Extract(id^*)$ query, B outputs random bits and aborts. Else if $\alpha = id^*$, B uses α to solve the q-DDHE problem. Else B generates a random bit $b \in \{0,1\}$. B also chooses random values $s \in Z_p, \theta \in G_T$ and computers

$$C_3^* = m_b \cdot e(g,h)^{-s}, \qquad \widetilde{C_3^*} = e(g,h)^{sH(\theta)}.$$

Next, B computes a private key (r_{id^*}, h_{id^*}) as in phase 1. Let $f_2(x) = x^{q+2}$ and $F_{2,id^*}(x) = (f_2(x) - f_2(id^*))/(x - id^*)$, B sets

$$C_1^{*\prime} = g^{f_2(\alpha) - f_2(id^*)}, \qquad C_2^{*\prime} = T \cdot \prod_{i=0}^{q} g^{F_{2,id^*,i} \cdot \alpha^i},$$

$$C_3^{*\prime} = \theta / (e(C_1^{*\prime}, h_{id^*}) \cdot e(g, C_2^{*\prime})^{r_{id^*}}),$$

where $F_{2,id_b,i}$ is the coefficient of x^i in $F_{2,id_b}(x)$. It sends $(C_3^*, \widetilde{C_3^*}, C_1^{*\prime}, C_2^{*\prime}, C_3^{*\prime})$ to A as the challenge ciphertext.

Note that, $(C_3^*, \widetilde{C_3^*}, C_1^{*\prime}, C_2^{*\prime}, C_3^{*\prime})$ is a valid ciphertext for (id^*, m_b).

4. Query phase 2: A continues making queries as in the query phase 1 with the restrictions described in the IE-CPA-R game.

5. Guess: A outputs the guesses $b' \in \{0,1\}$. If $b' = b$, B outputs 1 meaning $T = g^{\alpha^{q+1}}$; else output 0 meaning T is a random value in G.

Probability analysis. Same as the proof of Theorem 1.

This completes the proof of Theorem 2.

Theorem 3. Our scheme is IK-CPA-RK secure without random oracles under the q-DDHE assumption.

Proof. Assume that there is an adversary A that can break the IK-CPA-RK security of our scheme with probability ε, then we can build an algorithm B that can solve the q-DDHE problem with probability ε', where

$$\varepsilon' \geq \frac{\varepsilon}{\dot{e}(1+q_e)}.$$

B inputs a q-DDHE instance $(g, A_1 = g^{\alpha}, A_2 = g^{\alpha^2}, \ldots, A_q = g^{\alpha^q}, T)$ and has to distinguish $T = A_{q+1} = g^{\alpha^{q+1}}$ from a random element in G.

1. Setup: Same as the proof of Theorem 1.

2. Query phase 1: Same as the proof of Theorem 1.

3. Challenge: Once A decided that Query phase 1 is over, it outputs two identities id_I, id_J on which it wants to challenge. If id_I, id_J exist in the former $Extract(id)$ query, B outputs random bits and aborts. Else if $\alpha = id_I$ or $\alpha = id_J$, B uses α to solve the q-DDHE problem. Else, B computes two private key (r_{id_I}, h_{id_I}) and (r_{id_J}, h_{id_J}) as in phase 1. B also chooses random values $s, t \in Z_p, m, \theta \in G_T$. B computes the challenge re-encryption key as follows:

$$rk^{(1)} = r_{id_I} \cdot H(\theta) + t, \qquad rk^{(2)} = (h_{id_I})^{H(\theta)}, \qquad rk^{(3)} = g^t.$$

Let $f_2(x) = x^{q+2}$ and $F_{2,id_J}(x) = (f_2(x) - f_2(id_J))/(x - id_J)$, B also computers

$$C_{1'} = g^{f_2(\alpha) - f_2(id_J)}, \qquad C_{2'} = T \cdot \prod_{i=0}^{q} g^{F_{2,id_J,i} \cdot \alpha^i},$$

$$C_{3'} = \theta / (e(C_{1'}, h_{id_J}) \cdot e(g, C_{2'})^{r_{id_J}}).$$

B sets $rk^{(4)} = (C_{1'}, C_{2'}, C_{3'})$.

Finally, B chooses a random bit $b \in \{0,1\}$. If $b = 0$, B returns a random re-encryption key in the re-encryption key space to A. Else if $b = 1$, returns $(rk^{(1)}, rk^{(2)}, rk^{(3)}, rk^{(4)})$ as the challenge re-encryption key to A.

Note that, $(rk^{(1)}, rk^{(2)}, rk^{(3)}, rk^{(4)})$ is a valid re-encryption key from id_I to id_J.

4. Query phase 2: A continues making queries as in the query phase 1 with the restrictions described in the IK-CPA-RK game.

5. Guess: A outputs the guesses $b' \in \{0,1\}$. If $b' = b$, B outputs 1 meaning $T = g^{\alpha^{q+1}}$; else output 0 meaning T is a random value in G.

Probability analysis. Same as the proof of Theorem 1.

This completes the proof of Theorem 3.

4 Conclusions

In this paper, we formulate the security model of key-private identity-based proxy re-encryption and proposed the first key-private identity-based proxy re-encryption scheme. Our scheme is chosen plaintext secure and key-private without random oracles. Many interesting questions are still remaining to be solved.

CCA-secure. Designing chosen ciphertext secure and key-private constructions is very necessary. The technique introduced in Cantti et al. [Cantti, Halevi and Katz (2004); Fujisaki and Okamota (1999)] might possible approaches to achieve CCA-secure. We leave it as our future work.

IB-CREE of key-private and condition-private. Designing an identity-based conditional proxy re-encryption of key-private and condition-private remains an interesting work.

Acknowledgment: This work is supported by the National Natural Science Foundation of China (Nos. 61702236, 61672270, 61602216, 61872181) and Changzhou Sci & Tech Program (Grant No. CJ20179027).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Ateniese, G.; Benson, K.; Hohenberger, S. (2009): Key-private proxy re-encryption. *Proceedings of RSA*, vol. 5473, no. 6, pp. 279-294.

Ateniese, G.; Fu, K.; Green, M.; Hohenberger, S. (2005): Improved proxy re-encryption schemes with applications to secure distributed storage. *Proceedings of the 12th Annual Network and Distributed System Security Symposium*, vol. 9, no. 1, pp. 29-44.

Boneh, D.; Boyen, X. (2004): Efficient selective-ID based encryption without random oracles. *Proceedings of Eurocrypy*, vol. 24, no. 4, pp. 223-238.

Boneh, D.; Franklin, M. (2001): Identity-based encryption from the weil pairing. *Proceedings of Crypto*, vol. 2139, no. 8, pp. 231-229.

Blaze, M.; Bleumer, G.; Strauss, M. (1998): Divertible protocols and atomic proxy cryptography. *Proceedings of Eurocrypt*, vol. 1403, no. 12, pp. 127-144.

Canetti, R.; Halevi, S.; Katz, J. (2004): Chosen-ciphertext security from identity-based encryption. *Proceedings of PKC*, vol. 3027, no. 10, pp. 53-68.

Chu, C. K.; Tzeng, W. G. (2007): Identity-based proxy re-encryption without random oracles. *Proceedings of ISC*, vol. 4779, no. 6, pp. 189-202.

Fujisaki, E.; Okamota, T. (1999): How to enhance the security of public-key encryption at minimum cost. *Proceedings of Eurocrypt*, vol. 1560, no. 2, pp. 207-222.

Gentry, C. (2006): Practical identity-based encryption without random oracles. *Proceedings of Eurocrypt*, vol. 4004, no. 10, pp. 445-464.

Green, M.; Ateniese, G. (2007): Identity-based proxy re-encryption. *Proceedings of ACNS*, vol. 4521, no. 3, pp. 288-306.

Liang, K.; Liu, Z.; Tan, X.; Wang, D. S.; Tang, C. M. (2012): A CCA-secure identity-based conditional proxy re-encryption without random oracles. *Proceedings of International Conference on Information Security and Cryptology*, vol. 7839, no. 11, pp. 189-202.

Libert, B.; Vergnaud, D. (2008): Unidirectional chosen-ciphertext secure proxy re-encryption. *Proceedings of PKC*, vol. 4939, no. 2, pp. 360-379.

Shao, J.; Cao, Z. F. (2012): Multi-use unidirectional identity-based proxy re-encryption from hierarchical identity-based encryption. *Information Science*, vol. 206, no. 10, pp. 83-95.

Shao, J.; Liu, P.; Wei, G.; Lin, Y. (2011): Anonymous proxy re-encryption. *Security and Communication Networks*, vol. 5, no. 5, pp. 439-449.

Tang, Y. L.; Lian, H. H.; Zhao, Z. M.; Yan, X. X. (2018): A proxy re-encryption with keyword search scheme in cloud computing. *Computers, Materials & Continua*, vol. 56, no. 2, pp. 339-352.

Waters, B. (2005): Efficient identity-based encryption without random oracles. *Proceedings of Eurocrypt*, vol. 3494, no. 8, pp. 189-202.