On the Relation between Identity-Based Proxy Re-Encryption and Mediated Identity-Based Encryption

ZHONG-MEI WAN1,2, JIAN WENG3,4, XUE-JIA LAI1, SHENG-LI LIU1 AND JI-GUO LI5

1Department of Computer Science and Engineering
Shanghai Jiao Tong University
Shanghai 200240, P.R. China
2College of Science
Hohai University
Nanjing 210098, P.R. China
E-mail: wannmei@sjtu.edu.cn
3Department of Computer Science
Jinan University
Guangzhou 510630, P.R. China
4School of Information Systems
Singapore Management University
Singapore 178902, Singapore

Identity-based proxy re-encryption (IBPRE) is a useful primitive, in the sense that a semi-trust proxy can translate ciphertexts originally intended for one identity into ciphertexts intended for another identity. The proxy, however, cannot learn anything about the underlying plaintexts. Mediated identity-based encryption (MIBE), introduced by Ding et al., is particularly useful for the immediate revocation of identities. In this paper, we study the relation between IBPRE and MIBE. We show that, under the chosen-plaintext attack (CPA), IBPRE and MIBE are equivalent: we give a generic construction of CPA-secure IBPRE scheme from any CPA-secure MIBE scheme; and a generic construction for the opposite direction is also given. However, under the chosen-ciphertext attack (CCA), we show that IBPRE and MIBE are not equivalent: for an IBPRE scheme generically constructed from CCA-secure MIBE, we can give a concrete attack against this resulting IBPRE scheme; similarly, we also give a concrete attack against the MIBE scheme generically constructed from CCA-secure IBPRE. We believe that our results are theoretically interesting, since for the first time they clarify the relation between IBPRE and MIBE.

Keywords: identity-based proxy re-encryption, mediated identity-based encryption, chosen-plaintext attack, chosen-ciphertext attack, generic construction

1. INTRODUCTION

In 1998, Blaze, Bleumer and Strauss [5] introduced a cryptography primitive named proxy re-encryption (PRE), in which a proxy can transform ciphertexts computed under Alice’s public key into ciphertexts that can be decrypted by using Bob’s secret key. Such a system works as follows: Alice or a trusted third party generates a re-encryption key

Received January 12, 2009; revised July 1, 2009; accepted August 18, 2009.
Communicated by Wen-Guey Tzeng.
* Project (No. 60573032, 60773092, 90604036, 60873229, 60903178, 60842002, 60673070) supported by the National Natural Science Foundation of China, and Project No. 2007AA01Z409 supported by the National High-Tech Research and Development Plan of China.
and gives it to a proxy. On receiving a ciphertext encrypted under Alice’s public key, the proxy with the re-encryption key can transform the ciphertext by running the re-encryption algorithm, and sends the transformed ciphertext to Bob. Then Bob can decrypt it with his secret key. The proxy re-encryption system should at least satisfy the following requirements: (1) a proxy alone cannot obtain the underlying plaintext; (2) Bob cannot obtain the underlying plaintext without the proxy’s cooperation. A natural application of PRE is the encrypted email forwarding: when Alice is going on vacation, she can give a re-encryption key to the email server, and ask him to transfer Alice’s encrypted emails into Bob’s. Then Bob can read the encrypted e-mails for Alice. Once Alice comes back, the proxy stops transferring the e-mails. In ACNS’07, Green and Ateniese [17] extended PRE to identity-based settings, and introduced the notion of identity-based proxy re-encryption.

Mediated cryptography was introduced by Boneh et al. [8] as a method to allow immediate revocation of public keys. They suggested that such a primitive is particularly useful in government, corporation or military environments, where there may be an unexpected and immediate requirement to revoke a key when a user is suspected of being compromised, or when a user is removed from a position of authority. While previous revocation techniques cannot satisfy this requirement. The basic idea of mediated cryptography is to use an on-line mediator for every transaction. This on-line mediator is referred to as a SEM (Security Mediator) since it provides a control of security capabilities. If the SEM does not cooperate, then no transactions with the public key are possible any longer. Once the SEM is notified that a user’s key is to be revoked, he can immediately stop the services for this user. In CT-RSA’03, Ding and Tsudik [14] extended mediated cryptography to identity-based scenarios, and introduced the notion of mediated identity-based encryption.

In 2001, Boneh et al. [8] constructed the first mediated encryption and signature schemes using the RSA primitive. In CT-RSA’03, Ding et al. [14] introduced the notion of mediated identity-based encryption (MIBE), and proposed the first MIBE scheme based on RSA-OAEP [4]. Subsequently, Libert and Quisquater [19] re-formalized the security notion for MIBE, and proposed a new MIBE scheme with chosen-ciphertext security. Later, Baek and Zheng [3] also proposed a new MIBE scheme secure against chosen-ciphertext attack.

In Eurocrypt’98, Blaze et al. [5] introduced the primitive of proxy re-encryption (PRE), and proposed a bidirectional PRE scheme. In 2005, Ateniese et al. [1, 2] presented the first examples of unidirectional proxy re-encryption schemes. The constructions [1, 2] are only chosen-plaintext secure. In ACM CCS’07, Canetti et al. [10] presented a construction of CCA-secure bidirectional PRE scheme from bilinear pairings. Later, Libert and Quisquater [19] presented a CCA-secure unidirectional PRE scheme from bilinear pairings. In CANS’08, Deng et al. [13] presented a CCA-secure bidirectional PRE scheme without pairings. In PKC'09, Shao can Cao et al. [21] proposed a unidirectional PRE without pairings, and claimed that their scheme is CCA-secure in the random oracle model. However, Weng et al. [25] pointed out that Shao-Cao’s scheme is not CCA-secure by giving a concrete attack. Weng et al. [24, 25] also presented a more efficient CCA-secure unidirectional PRE scheme without pairings.

To control the proxy at a fine-grained level, Tang [22] and Weng et al. [26] inde-
pendently introduced a variant of PRE named conditional proxy re-encryption (C-PRE). In such systems, ciphertexts are generated with respect to a certain condition, and the proxy can translate a ciphertext only if the associated condition is satisfied. Recently, Weng et al. [27] re-formalized the definition and security notions for C-PRE systems, and presented a more efficient C-PRE scheme. Chu et al. [12] introduced a generalized concept named conditional proxy broadcast re-encryption (CPBRE), and proposed a replayable CCA-secure CPBRE scheme.


Another kind of cryptosystems related to proxy re-encryption should be mentioned is the proxy encryption cryptosystem [15, 20]. As argued in [10, 18], proxy re-encryption schemes are a (strict) subset of proxy encryption schemes, since in proxy encryption systems, the delegatee is required to obtain and store an additional secret for each delegation. In contrast, the delegatee in proxy re-encryption systems only needs to store his own decryption key.

In this paper, we study the relation between identity-based proxy re-encryption (IBPRE) and mediated identity-based encryption (MIBE). We show that, under the chosen-plaintext attack (CPA), IBPRE and MIBE are equivalent. Concretely, we give a generic construction of CPA-secure IBPRE scheme from any CPA-secure MIBE scheme; and a generic construction for the opposite direction is also given. However, under the chosen-ciphertext attack (CCA), we show that IBPRE and MIBE are not equivalent: for an IBPRE scheme generically constructed from CCA-secure MIBE, we can give a concrete attack against this resulting IBPRE scheme; similarly, we also give a concrete attack against the MIBE scheme generically constructed from CCA-secure IBPRE. We believe that our results are theoretically interesting, since for the first time they clarify the relation between IBPRE and MIBE.

The rest of this paper is organized as follows. In section 2, we review the definitions and security notions for identity-based proxy re-encryption (IBPRE) and mediated identity-based encryption (MIBE). In section 3, we indicate the equivalence between IBPRE and MIBE under the chosen-plaintext attacks, and we also show the inequivalence between IBPRE and MIBE under the chosen-ciphertext attacks. Finally, section 4 concludes this paper.

2. PRELIMINARIES

In this section, we will review the formal definitions and security models for identity-based proxy re-encryption and mediated identity-based encryption.

2.1 Identity-based Proxy Re-encryption

An identity-based proxy re-encryption scheme consists of the following algorithms:
Setup(κ): The setup algorithm takes as input a security parameter κ. It generates the public parameters param and the corresponding master secret key msk. We write (param, msk) ← Setup(κ).

Extract(msk, ID): The key generation algorithm takes as input the master secret key msk and an identity ID. It outputs a secret key skID for identity ID. We write skID ← Extract(msk, ID).

Encrypt(param, ID, m): The encryption algorithm takes as input the public parameters param, an identity ID and a plaintext m ∈ M. It outputs a ciphertext C encrypted under ID. We write C ← Encrypt(param, ID, m).

RKGen(skID1, ID2): The re-encryption key generation algorithm takes as input a secret key skID1 and an identity ID2. It outputs a re-encryption key rkID1→ID2. We write rkID1→ID2 ← RKGen(skID1, ID2).

ReEncrypt(rkID1→ID2, CID1): The re-encryption algorithm takes as input a ciphertext CID1 under the identity ID1, and a re-encryption key rkID1→ID2. It outputs a transformed ciphertext CID2 under the identity ID2. For national convenience, we write CID2 ← ReEncrypt(rkID1→ID2, CID1).

Decrypt(skID, CID): The decryption algorithm takes as input a secret key skID and a ciphertext CID. It outputs a message m ∈ M or the distinguished symbol ⊥. We write m ← Decrypt(skID, CID).

Roughly speaking, the correctness requires that, for any m ∈ M and any identity ID1 and ID2, the following equations should hold:

Decrypt(Extract(msk, ID1), Encrypt(param, ID1, m)) = m;
Decrypt(Extract(msk, ID2), ReEncrypt(rkID1→ID2, Encrypt(param, ID1, m))) = m;

where rkID1→ID2 ← RKGen(Extract(msk, ID1), ID2).

Next, we review the chosen-plaintext as well as the chosen-plaintext security notions for IBPRE systems as defined in [11, 17]. The security for an IBPRE scheme is defined via the following game Exp^A,IND-PrID-ATK, where ATK ∈ {CPA, CCA}:

Initialize The challenger C runs algorithm Setup(κ), and gives A the resulting public parameters param, keeping the master key msk to itself.

Phase 1 In this phase, adversary A adaptively issues a polynomial bounded number of queries:

Key extraction queries CExt(·): On receiving a query CExt(ID), challenger C first runs algorithm Extract to obtain skID. It then sends skID to A.

Re-encryption key generation queries CExt(·): When A issues a re-encryption key generation query CExt(ID1, ID2), challenger C first runs algorithm Extract to generate the secret key skID1. Next, it runs algorithm RKGen to generate the re-encryption key rkID1→ID2. Finally, it returns rkID1→ID2 to A. If ATK = CCA, A can make the additional queries:

Re-encryption queries CExt(·): On receiving a query CExt(CID1, ID1, ID2), C first runs algorithm Extract to generate the secret key skID1, and then runs algorithm RKGen to
generate the re-encryption key $rk_{ID_1\rightarrow ID_2}$. Then it runs algorithm ReEncrypt($rk_{ID_1\rightarrow ID_2}$, $C_{ID_1}$), and returns the resulting ciphertext $C_{ID_2}$ to $A$.

**Decryption queries** $C_{\text{Dec}}(\cdot, \cdot)$: On receiving a query $C_{\text{Dec}}(C, ID)$, $C$ first runs algorithm Extract to generate the secret key $sk_{ID}$, and then returns $\text{Decrypt}(sk_{ID}, C)$ to $A$.

**Challenge** Once $A$ decides that Phase 1 is over, it outputs a target identity $ID^*$ and two equal-length plaintexts $m_0$ and $m_1$, on which it wishes to be challenged. Challenger $C$ flips a random coin $\beta \in \mathbb{R} \{0, 1\}$, and computes $C^* = \text{Encrypt}(\text{param}, ID^*, m_\beta)$.

Finally, $C$ returns $C^*$ as the challenge ciphertext to $A$.

**Query Phase 2** $A$ continues to make queries as in the Query Phase 1, and $C$ responds to these queries as in Phase 1.

**Guess** Finally, $A$ outputs a guess $\beta' \in \mathbb{R} \{0, 1\}$.

During the above game, it is required that the following restrictions should be simultaneously satisfied:

(i) $A$ cannot issue a query $C_{\text{Ext}}(ID^*)$;
(ii) For any identity $ID'$, $A$ cannot simultaneously issue $C_{\text{Ext}}(ID^*, ID')$ and $C_{\text{Ext}}(ID')$;
(iii) For any identity $ID'$, $A$ cannot simultaneously issue $C_{\text{Ext}}(C^*, ID^*, ID')$ and $C_{\text{Ext}}(ID')$;
(iv) $A$ cannot issue the query $C_{\text{Dec}}(C^*, ID^*)$;
(v) $A$ cannot issue the query $C_{\text{Dec}}(C^*, ID^*)$ such that $C^* = \text{ReEncrypt}(rk_{ID^*\rightarrow ID'}, C^*)$.

We refer the above adversary $A$ as an IND-PrID-ATK adversary, and define its advantage in attacking the IBPRE scheme as

$$\text{Adv}^{\text{IND-PrID-ATK}}_A = |\Pr[\beta = \beta'] - 1/2|$$

where the probability is taken over the random coins consumed by the challenger and the adversary.

**Definition 1** We say that an IBPRE scheme is IND-PrID-ATK-secure, $\text{ATK} \in \{\text{CPA, CCA}\}$ for all probabilistic polynomial time IND-PrID-ATK adversary $A$, we have that the advantage $\text{Adv}^{\text{IND-PrID-ATK}}_A(\kappa)$ is negligible in $\kappa$.

### 2.2 Mediated Identity-based Encryption

Formally, a mediated identity-based (MIBE) scheme consists of the following five algorithms:

- **MIBSetup($\kappa$):** The setup algorithm takes as input a security parameter. It generates the public parameters $\text{param}$ and the corresponding master secret key $\text{msk}$. We write $(\text{param}, \text{msk}) \leftarrow \text{MIBSetup}(\kappa)$. 

MIBExtract($msk$, ID): The key generation algorithm takes as input the master secret key $msk$ and an identity ID. It outputs a SEM’s secret key $d_{ID,sem}$ and a user’s secret key $d_{ID,user}$. We write $(d_{ID,sem}, d_{ID,user}) \leftarrow \text{MIBExtract}(msk, ID)$.

MIBEncrypt($param$, ID, $m$): The encryption algorithm takes as input the public parameters $param$, an identity ID and a plaintext $m$. It outputs a ciphertext $C$ encrypted under ID. We write $C \leftarrow \text{MIBEncrypt}(param, ID, m)$.

DecS($d_{ID,sem}$, $C$): The SEM decryption algorithm takes as input the SEM’s secret key $d_{ID,sem}$ and a ciphertext $C$ under identity ID. It outputs a token $T$ (or “⊥” if $C$ is invalid), which can be seen as a partial decryption result of $C$. We write $T \leftarrow \text{DecS}(d_{ID,sem}, C)$.

DecU($d_{ID,user}$, $C$, $T$): The user decryption algorithm takes as input a ciphertext $C$ under identity ID, a token $T$ and the user’s secret key $d_{ID,user}$. It outputs the plaintext $m$ (or “⊥” if $(C, T)$ is invalid). We write $m \leftarrow \text{DecU}(d_{ID,user}, C, T)$.

Roughly speaking, the correctness requires that, for any $m \in M$ and any identity ID, the following equation should hold:

$$\text{DecU}(d_{ID,user}, C, \text{DecS}(d_{ID,sem}, C)) = m;$$

where $C \leftarrow \text{MIBEncrypt}(param, ID, m)$ and $(d_{ID,sem}, d_{ID,user}) \leftarrow \text{MIBExtract}(msk, ID)$.

Next, we review the security notions for MIBE systems [9]. The security for an IB-PRE scheme is defined via the following game Exp$^\text{IND-MID-ATK}_A$, where ATK $\in \{\text{CPA, CCA}\}$:

Initialize The challenger runs algorithm MIBSetup($\kappa$), and gives $A$ the resulting public parameters $param$, keeping the master key $msk$ to itself.

Phase 1 In this phase, adversary $A$ adaptively issues a polynomial bounded number of queries:

User key extraction queries $C_{\text{ExtraU}}(\cdot)$: On receiving a $C_{\text{ExtraU}}(ID)$ query, challenger $C$ first runs algorithm MIBExtract to obtain the user part of secret key $d_{ID,user}$, and then returns it to $A$.

SEM key extraction queries $C_{\text{ExtraS}}(\cdot)$: On receiving a $C_{\text{ExtraS}}(ID)$ query, challenger $C$ first runs algorithm MIBExtract to obtain the SEM part of secret key $d_{ID,sem}$, and then sends it to $A$.

If ATK = CCA, $A$ can make the additional queries:

SEM decryption queries $C_{\text{DecS}}(\cdot, \cdot)$: On receiving a query $C_{\text{DecS}}(C, ID)$, $C$ first runs algorithm MIBExtract to obtain $d_{ID,sem}$, and then runs $T \leftarrow \text{DecS}(d_{ID,sem}, C)$. Finally, it returns $T$ to $A$.

User decryption queries $C_{\text{DecU}}(\cdot, \cdot, \cdot)$: On receiving a query $C_{\text{DecU}}(C, T, ID)$, $C$ first runs algorithm MIBExtract to generate $d_{ID,user}$, and then runs $m \leftarrow \text{DecU}(d_{ID,user}, C, T)$. Finally, it returns $m$ to $A$. 

Challenge  Once $A$ decides that Phase 1 is over, it outputs a target identity ID* and two equal-length plaintexts $m_0$ and $m_1$, with the restriction that $A$ did not issue the query $C_{\text{extral}}(\text{ID}*).$ Challenger $C$ picks a random bit $b \in \{0, 1\}$, and sets the challenge ciphertext to be $C^* = \text{MIBEncrypt}(\text{param, } m_b, \text{ID}^*)$, which is returned to $A$.

Phase 2  $A$ continues to issue queries as in Phase 1, and $C$ responds to these queries as in Phase 1. Here it is required that, $A$ cannot issue the user key extraction query $C_{\text{extral}}(\text{ID}^*)$. If $\text{ATK} = \text{CCA}$, $A$ is additionally disallowed to issue the user decryption query $C_{\text{decu}}(C^*, T^*, \text{ID}^*)$ with $T^* = \text{DecS}(d_{\text{ID}, \text{sem}}, C^*)$.

Guess  Finally, $A$ outputs a guess $b' \in \{0, 1\}$.

We refer to the above adversary $A$ as an IND-MID-ATK adversary, and define its advantage in attacking the MIBE scheme as

$$\text{Adv}_{A}^{\text{IND-MID-ATK}} = |\Pr[b' = b] - 1/2|$$

where the probability is taken over the random coins consumed by the challenger and the adversary.

Definition 2  We say that a MIBE scheme is IND-MID-ATK-secure, $\text{ATK} \in \{\text{CPA, CCA}\}$, if for all probabilistic polynomial time algorithm $A$, his advantage $\text{Adv}_{A}^{\text{IND-MID-ATK}}(\kappa)$ is negligible in $\kappa$.

3. RELATIONS BETWEEN IBPRE AND MIBE

In this section, we will classify the relation between IBPRE systems and MIBE systems. On the one hand, under the CPA-security, we show that IBPRE and MIBE are equivalent. Concretely, we can give a generic construction of CPA-secure IBPRE scheme from any CPA-secure MIBE scheme; and a generic construction for the opposite direction can also be given. On the other hand, under the CCA-security, we will show that IBPRE and MIBE are not equivalent. Specifically, for an IBPRE scheme generically constructed from CCA-secure MIBE, we can give a concrete attack against this resulting IBPRE scheme; similarly, we can also give a concrete attack against the MIBE scheme generically constructed from CCA-secure IBPRE.

3.1 Equivalence of MIBE and IBPRE under the CPA-Security

3.1.1 IND-MID-CPA-secure MIBE $\Rightarrow$ IND-PrID-CPA-secure IBPRE

We here show how to construct an IND-PrID-CPA-secure IBPRE scheme from an IND-MID-CPA-secure MIBE scheme. The intuition behind our construction is stated as follows: For a user ID with secret key $(d_{\text{ID}_{\text{sem}}, d_{\text{ID}_{\text{user}}}})$, he gives $d_{\text{ID}_{\text{user}}}$ to the proxy as the re-encryption key, and encrypts $d_{\text{ID}_{\text{sem}}}$ under the delegatee’s identity to generate a ciphertext $R$, which is also given to the proxy. To re-encrypted a ciphertext, the proxy uses
\( d_{ID, user} \) to run algorithm DecS and obtain a partial decryption result \( T \). Then the proxy appends \( R \) to \( T \), and gives them to the delegatee. Using his own secret key, the delegatee first recovers \( d_{ID, user} \) from \( R \), and then he can use \( d_{ID, user} \) to further recover the plaintext from \( T \). Concretely, let \( \Pi_{\text{MIBE}} = (\text{MIBSetup}, \text{MIBExtract}, \text{MIBEncrypt}, \text{DecS}, \text{DecU}) \) be an IND-MID-CPA-secure MIBE scheme, then an IND-PrID-CPA-secure IBPRE scheme \( \Pi_{\text{IBPRE}} = (\text{Setup}, \text{Extract}, \text{Encrypt}, \text{RKGen}, \text{ReEncrypt}, \text{Decrypt}) \) can be constructed as shown in Fig. 1.

**Theorem 1** The resulting IBPRE scheme as shown in Fig. 1 is IND-PrID-CPA-secure, assuming the underlying MIBE scheme is IND-MID-CPA-secure.

**Fig. 1.** Generic construction of IBPRE from MIBE.
Proof: Suppose there exists an adversary $A$ who can break the IND-PrID-CPA security of the IBPRE scheme. Then we will show how to construct an adversary $B$ against the IND-MID-CPA security of the underlying MIBE scheme. At the beginning, $B$ is given by his challenger $C$ a public parameters $param$. Next, according to the Exp$_{IND$-$PrID$-$CPA}$ game, $B$ acts as the challenger for $A$ and interacts with $A$ as follows:

Initialize $B$ forwards $param$ to $A$.

Phase 1 In this phase, $B$ answers a series of queries for $A$ as follows:

Key extraction queries $C_{ex}()$: When $A$ issues a query $C_{ex}(ID)$, $B$ forwards ID to its challenger as a SEM key extraction query $C_{ex}(ID)$ and a user key extraction query $C_{ex}(ID)$, and then is given the user part of secret key $d_{ID, user}$ and the SEM part of secret key $d_{ID, sem}$. Then $B$ sets $sk_{ID} = (d_{ID, sem}, d_{ID, user})$ and returns $sk_{ID}$ to $A$.

Re-encryption key generation queries $C_{rk}(·, ·)$: When $A$ issues a re-encryption key generation query $C_{rk}(ID_1, ID_2)$, $B$ first issues queries $C_{ex}(ID_1)$ and $C_{ex}(ID_2)$ to its own challenger, and then is given the corresponding user part of secret key $d_{ID_1, user}$ and the SEM part of secret key $d_{ID_1, sem}$. Then $B$ runs $MIBEncrypt(param, ID_2, d_{ID_1, user})$ to generate a ciphertext $R$. Finally, $B$ returns $rk_{ID_1 → ID_2} = (d_{ID_1, sem}, R)$ to $A$.

Challenge Once $A$ decides that Phase 1 is over, according to the constrains specified in the Exp$_{IND$-$PrID$-$CPA}$ game, he returns a target identity $ID^*$ and two equal-length plaintexts $(m_0, m_1)$. $B$ forwards $ID^*$ and $(m_0, m_1)$ to its challenger, and then is given a target ciphertext $C^*$. Then $B$ forwards $C^*$ to $A$.

Phase 2 $A$ continues to issue the rest of queries, and $B$ answers these queries as follows:

Key extraction queries $C_{ex}()$: When $A$ issues a query $C_{ex}(ID)$, $B$ forwards ID to its challenger as a SEM key extraction query $C_{ex}(ID)$ and a user key extraction query $C_{ex}(ID)$, and then is given the user part of secret key $d_{ID, user}$ and the SEM part of secret key $d_{ID, sem}$. Then $B$ sets $sk_{ID} = (d_{ID, sem}, d_{ID, user})$ and returns $sk_{ID}$ to $A$.

Re-encryption key generation queries $C_{rk}(·, ·)$: When $A$ issues a re-encryption key generation query $C_{rk}(ID_1, ID_2)$, $B$ responds according to the following cases:

1. $ID_1 ≠ ID^*$: In this case, $B$ first issues queries $C_{ex}(ID_1)$ and $C_{ex}(ID_2)$ to its own challenger, and then is given the corresponding user part of secret key $d_{ID_1, user}$ and the SEM part of secret key $d_{ID_1, sem}$. Then $B$ runs $MIBEncrypt(param, ID_2, d_{ID_1, user})$ to generate a ciphertext $R$. Finally, $B$ returns $rk_{ID_1 → ID_2} = (d_{ID_1, sem}, R)$ to $A$.

2. $ID_1 = ID^*$: Note that according to the Exp$_{IND$-$PrID$-$CPA}$ game, in this case, $A$ is disallowed to issue an extraction query on $ID_2$. To answer the query $C_{ex}(ID_1, ID_2)$ for $A$, $B$ first issues the query $C_{ex}(ID_2)$ to its own challenger, and then is given the corresponding SEM part of secret key $d_{ID_1, sem}$ (note that according to the Exp$_{IND$-$MID$-$CPA}$ game, $B$ is allowed to issues such a query). Then $B$ picks a random element $d'$ from the space of user part of secret keys, and runs algorithm $MIBEncrypt(param, ID_2, d')$ to obtain a ciphertext $R$. Then $B$ sets $rk_{ID_1 → ID_2} = (d_{ID_1, sem}, R')$, and returns $rk_{ID_1 → ID_2}$ to $A$. 

Guess Finally, \( A \) outputs a bit \( \beta' \). \( B \) sets \( b' = \beta' \) and returns it to its challenger \( C \) as its own answer.

This completes the description of the simulation. It is clear that, the responds to the key extraction queries for \( A \) are perfect, and the responds to the re-encryption key generation queries are also perfect if \( ID_1 \neq ID^* \). While for the case \( ID_1 = ID^* \), we have the following lemma:

Lemma 1 For the case \( ID_1 = ID^* \), the responds to the re-encryption key generation queries are computationally indistinguishable from the real environment, assuming the underlying MIBE scheme is IND-MID-CPA-secure.

Proof: (Sketch) The proof of this lemma is similar to the Lemma 1 in [11] and the Claim in [17]. We first show that the responds to the re-encryption key generation queries are computationally indistinguishable from the real environment. Recall that according to the \( \text{Exp}_{\text{IND-PrID-CPA}} \) game, since \( ID_1 = ID^* \), we know that the secret key for \( ID_2 \) must be unknown for adversary \( A \), which further implies that \( A \) does not know \( d_{ID_2,\text{user}} \). Then since the underlying MIBE scheme is IND-MID-CPA-secure, it is impossible for \( A \) to distinguish the ciphertext \( R' = \text{IBEEncrypt}(\text{param}, ID_2, d') \) and the ciphertext \( R = \text{MIBEncrypt}(\text{param}, ID_2, d_{ID^*,\text{user}}) \). So, the re-encryption key provided for \( A \) is indistinguishable from the real one, and hence the responds to the re-encryption key generation queries are computationally indistinguishable from the real environment. Therefore, the simulation provided for \( A \) is computationally indistinguishable from the real environment. Therefore, if \( A \) can correctly guess the challenge, then so does \( B \). In other word, if \( A \) can break the IND-PrID-CPA security of our IBPRE scheme, then \( B \) can break the IND-MID-CPA security of the underlying MIBE scheme. This concludes the proof of Theorem 1.

3.1.2 IND-PrID-CPA-secure IBPRE \( \Rightarrow \) IND-MID-CPA-secure MIBE

We here show how to construct an IND-MID-CPA-secure MIBE scheme from an IND-PrID-CPA-secure IBPRE scheme. The intuition behind our construction is stated as follows:

For the user \( ID \) with secret key \( sk_{ID} \) and the user \( ID' \) with secret key \( sk_{ID'} \), the PKG first run algorithm \( \text{RKGen}(sk_{ID}, ID') \) to generate a re-encryption key \( rk_{ID \rightarrow ID'} \) and then sets \( d_{ID,\text{user}} = sk_{ID} \) and \( d_{ID,\text{sem}} = rk_{ID \rightarrow ID'} \). To decrypt a ciphertext, the security mediator runs algorithm \( \text{ReEncrypt}(d_{ID,\text{sem}}, C) \) to generate a token \( T \) using his own secret key \( d_{ID,\text{sem}} \) and gives \( T \) to the user. Using his own secret key \( d_{ID,\text{user}} \), the user can recover the plaintext from \( T \). Concretely, for an IND-PrID-CPA-secure identity-based proxy re-encryption scheme \( \prod_{\text{IBPRE}} = (\text{Setup}, \text{Extract}, \text{Encrypt}, \text{RKGen}, \text{ReEncrypt}, \text{Decrypt}) \), we can construct an IND-MID-CPA-secure mediated identity-based encryption scheme \( \prod_{\text{MIBE}} = (\text{MIBSetup}, \text{MIBExtract}, \text{MIBEncrypt}, \text{DecS}, \text{DecU}) \) as shown in Fig. 2.

Theorem 2 The resulting MIBE scheme as shown in Fig. 2 is IND-MID-CPA-secure, assuming the underlying IBPRE scheme is IND-PrID-CPA-secure.

Proof: Suppose there exists an adversary \( A \) who can break the IND-MID-CPA security.
of the MIBE scheme. Then we will show how to construct an adversary $B$ against the IND-PrID-CPA security of the underlying IBPRE scheme.

\begin{verbatim}
MIBSetup(κ):
    Run (msk, param) ← Setup(κ)
    Return (param, msk)

MIBExtract(msk, ID)
    Run skID ← Extract(msk, ID)
    Pick a random identity ID’ with ID’ ≠ ID
    Run skID’ ← Extract(msk, ID’)
    Run rkID→ID’ ← RKGen(skID, ID’)
    Set dID,user = skID
    Set dID,sem = rkID→ID’
    Run (dID,sem, dID,user)

MIBEncrypt(param, ID, m)
    Run C ← Encrypt(param, ID, m)
    Return C

DecS(dID,sem, C)
    Run T ← ReEncrypt(dID,sem, C)
    Return T

DecU(dID,sem, C, T)
    Decrypt(dID,sem, T)
    Return m
\end{verbatim}

Fig. 2. Generic construction of MIBE from IBPRE.

At the beginning, $B$ is given by his challenger $C$ a public parameters $param$. Next, according to the Exp^{IND-MID-CPA} game, $B$ acts as the challenger for $A$ and interacts with $A$ as follows:

**Initialize**  
$B$ forwards $param$ to $A$.

**Phase 1**  
At the beginning of this phase, $B$ maintains a list $D^{list}$ which is initially empty. Then $B$ answers a series of queries for $A$ as follows:

**User key extraction queries** $C_{\text{ExtU}}(.)$: When $A$ issues a query $C_{\text{ExtU}}(ID)$, $B$ acts as follows:

1. $B$ first searches in list $D^{list}$ whether there exits a tuple (ID, ID’) for ID. If not such tuple exists, $B$ picks a random identity ID’ from the identity space, and insert (ID, ID’) into $D^{list}$.
2. $B$ issues a key extraction query $C_{\text{ExtU}}(ID’)$ to its challenger, and then is given the corresponding secret key $sk_{ID’}$.
3. $B$ defines $d_{ID,\text{user}} = sk_{ID’}$ and returns $d_{ID,\text{user}}$ to $A$. 

SEM key extraction queries $O_{ExtraS}(\cdot)$: When $A$ issues a query $O_{DecS}(C, ID)$, $B$ acts as follows:

1. $B$ first searches in list $D^{list}$ whether there exits a tuple (ID, ID') for ID. If not such tuple exists, $B$ picks a random identity ID' from the identity space, and insert (ID, ID') into $D^{list}$.
2. $B$ issues a re-encryption key generation query $O_{RK}(ID, ID')$ to its challenger. Then $B$ is given a re-encryption key $rk_{ID \rightarrow ID'}$.
3. $B$ sets $d_{ID, sem} = rk_{ID \rightarrow ID'}$ and returns $d_{ID, sem}$ to $A$.

Challenge Once $A$ decides that Phase 1 is over, according to the restraints specified in the Exp_{IND-MID-CPA} game, he returns a target identity ID* and two equal-length plaintexts $(m_0, m_1)$. $B$ forwards ID* and $(m_0, m_1)$ to his challenger, and then is given a target ciphertext $C^*$. Then $B$ forwards $C^*$ to $A$.

Phase 2 $A$ continues to issue the rest of queries as in Phase 1, with the restraints as specified in the Exp_{IND-MID-CPA} game. $B$ answers these queries as follows:

User key extraction queries $O_{ExtraU}(\cdot)$: When $A$ issues a query $O_{ExtraU}(ID)$, $B$ acts as follows:

1. $B$ first searches in list $D^{list}$ whether there exits a tuple (ID, ID') for ID. If not such tuple exists, $B$ picks a random identity ID' from the identity space such that ID' $\neq$ ID*, and insert (ID, ID') into $D^{list}$.
2. $B$ issues a key extraction query $O_{ExtraU}(ID')$ to its challenger, and then is given the corresponding secret key $sk_{ID'}$.
3. $B$ defines $d_{ID, user} = sk_{ID'}$ and returns $d_{ID, user}$ to $A$.

SEM key extraction queries $O_{ExtraS}(\cdot)$: When $A$ issues a query $O_{ExtraS}(ID)$, $B$ acts as follows:

1. $B$ first searches in list $D^{list}$ whether there exits a tuple (ID, ID') for ID. If not such tuple exists, $B$ picks a random identity ID' from the identity space such that ID' $\neq$ ID*, and insert (ID, ID') into $D^{list}$.
2. $B$ issues a re-encryption key generation query $O_{RK}(ID, ID')$ to its challenger (note that $B$ is allowed to issue such a query even if ID = ID*). Then $B$ is given a re-encryption key $rk_{ID \rightarrow ID'}$.
3. $B$ sets $d_{ID, sem} = rk_{ID \rightarrow ID'}$, and returns $d_{ID, sem}$ to $A$.

Guess Finally, $A$ outputs a bit $b'$. $B$ sets $\beta' = b'$ and returns it to its challenger $C$ as its own answer.

From the above description of the simulation, it can easily be seen that, the simulation provided for $A$ is indistinguishable from the real environment with overwhelming probability. Therefore, if $A$ can break the IND-MID-CPA security of the MIBE scheme, then $B$ can break the IND-PrID-CPA security of the underlying IBPRE scheme. This concludes the proof of Theorem 2.
3.2 Inequivalence of IBPRE and MIBE under the CCA-Security

In this subsection, we will show the separated results between IND-MID-CCA-secure IBE and IND-PrID-CCA-secure IBPRE.

3.2.1 IND-MID-CCA-secure MIBE $\Rightarrow$ IND-PrID-CCA-secure IBPRE

We first show that, even if the underlying MIBE scheme is IND-MID-CCA-secure, the resulting IBPRE as shown in Fig. 1 is not IND-PrID-CCA-secure. Concretely, we can give an adversary $A_{IBPRE}$ against the IND-PrID-CCA-security of the resulting IBPRE scheme. Adversary $A_{IBPRE}$ works as follows:

1. In Phase 1, $A_{IBPRE}$ does not issue any query.
2. In Challenge phase, $A_{IBPRE}$ outputs a target identity ID* and two equal-length plaintexts $m_0$ and $m_1$. Then $A_{IBPRE}$ is given a challenge ciphertext $C*$ under the identity ID* for the plaintext $m_\beta$, where $\beta$ is the random bit chosen by the challenger.
3. In Phase 2, $A_{IBPRE}$ works as follows:
   (a) $A_{IBPRE}$ first issues a key extraction query $O_{Ext(ID_2)}$ with $ID_2 \neq ID*$, and then is given the secret key $sk_{ID_2} = (d_{ID_2, nom}, d_{ID_2, user})$.
   (b) $A_{IBPRE}$ generates a ciphertext $C'$ under the target identity ID*, and then issues a re-encryption query $O_{RE}(C', ID_*, ID_2)$ to obtain a transformed ciphertext $C'_{ID_2} = (T', R, C')$, where $R = MIBEncrypt(param, ID_2, d_{ID_2, aux})$. Note that it is legal for $A_{IBPRE}$ to issue this query $O_{RE}(C', ID_*, ID_2)$, since $C'_0 \neq C_*$. Then, using $sk_{ID_2}, A_{IBPRE}$ can decrypt $R$ and obtain $d_{ID_2, user}$.
   (c) $A_{IBPRE}$ issues a re-encryption query $O_{DecS}(C_*, ID_*)$ to obtain a transformed ciphertext $C'_{ID_2} = (T'_*, R, C'_*)$. Note that it is legal for $A_{IBPRE}$ to issue this query, since he does not issue the extraction query $O_{Ext(ID_3)}$.
   (d) Now, using $d_{ID_2, user}, A_{IBPRE}$ can obtain the underlying plaintext $m_\beta$ by running $\text{DecU}(d_{ID_2, user}, C'_*, T'_*)$. With $m_\beta$, he can certainly know the random bit $\delta$ selected by the challenger in the Challenge phase.

Therefore, $A_{IBPRE}$ can break the IND-PrID-CCA security of the IBPRE scheme.

3.2.2 IND-PrID-CCA-secure IBPRE $\Rightarrow$ IND-MID-CCA-secure MIBE

Next, we will show that, even if the underlying IBPRE scheme is IND-PrID-CCA-secure, the resulting MIBE as shown in Fig. 2 is not IND-MID-CCA-secure. Concretely, we can give an adversary $A_{MIBE}$ against the IND-MID-CCA-security of the resulting MIBE scheme. Adversary $A_{MIBE}$ works as follows:

1. In Phase 1, $A_{MIBE}$ does not issue any query.
2. In Challenge phase, $A_{MIBE}$ outputs a target identity ID* and two equal-length plaintexts $m_0$ and $m_1$. Then $A_{MIBE}$ is given a challenge ciphertext $C*$ under the identity ID* for the plaintext $m_b$, where $b$ is the random bit chosen by the challenger.
3. In Phase 2, $A_{MIBE}$ works as follows:
   (a) $A_{MIBE}$ first issues a SEM decryption query $O_{DecS}(C_*, ID_*)$ and obtain a token $T_*$. 
(b) $A_{\text{MIBE}}$ picks a random ciphertext $C'$ with $C' \neq C^*$. Then $A_{\text{MIBE}}$ issues a user decryption query $O_{\text{DecU}}(C^*, T^*, \text{ID}^*)$. Note that according to the constraints described in the $\text{Exp}^{\text{IND-MID-CCA}}$ game, $A_{\text{MIBE}}$ is allowed to issue this query, since $C' \neq C^*$. Note also that, according to the algorithm $\text{DecU}$ described in Fig. 2, the output of $\text{DecU}(d_{\text{id, user}}, C, T)$ has nothing to do with the component $C$. So, the respond of $O_{\text{DecU}}(C^*, T^*, \text{ID}^*)$ will return $m_b$ to $A_{\text{MIBE}}$.

(c) With $m_b$, $A_{\text{MIBE}}$ can certainly know the random bit $b$ selected by the challenger in the Challenge phase.

Therefore, $A_{\text{MIBE}}$ can break the IND-MID-CCA security of the MIBE scheme.

4. CONCLUSION

We studied the relation between identity-based proxy re-encryption (IBPRE) and mediated identity-based encryption (MIBE). We showed the equivalence between IBPRE and MIBE under the chosen-plaintext attacks, by giving a generic construction of CPA-secure IBPRE from CPA-secure MIBE, as well as a generic construction of CPA-secure MIBE from CPA-secure IBPRE. We further showed that under the chosen-ciphertext attacks, IBPRE and MIBE are not equivalent: for an IBPRE scheme generically constructed from CCA-secure MIBE, we are able to give a concrete attack against this resulting IBPRE scheme; similarly, we can also give a concrete attack against the MIBE scheme generically constructed from CCA-secure IBPRE.

ACKNOWLEDGEMENT

We are grateful to the anonymous reviewers for their helpful comments and suggestions for the improvements of this paper.

REFERENCES


22. Q. Tang, “Type-based proxy re-encryption and its construction,” in *Proceedings of the...*
Zhong-Mei Wan (万中美) received her B.S. degrees in Department of Mathematics from Zhe Jiang University, Hangzhou, P.R. China, in 2001. Since 2001, she has been working at the College of Science, Hohai University, Nanjing. She is now a Ph.D. candidate in Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai. Her research interests include the analysis, design, and application of digital signatures, and certificateless cryptography.

Jian Weng (翁健) received his M.S. and B.S. degrees in Computer Science and Engineering from South China University of Technology, Guangzhou, P.R. China, in 2004 and 2000, respectively. He received his Ph.D. degree in Computer Science and Engineering from Shanghai Jiao Tong University, Shanghai, in 2008. Then he is an associate professor in Jinan University, Guangzhou. Now he is a postdoc in School of Information Systems, Singapore Management University, Singapore. His research interests include key-exposure protection mechanism, provable security and pairing-based cryptography.

Xuejia Lai (来学嘉) received Ph.D. of sc. Techn. in 1992 from the Swiss Federal Institute of Technology, Zurich. He is a professor of Shanghai Jiao Tong University since 2004. His work has been concentrated in cryptography during the past 25 years, including the design and analysis of block ciphers, stream ciphers and DNA cipher, differential cryptanalysis of block ciphers, and
analysis and hash functions. He is co-inventor (together with J. L. Massey) of the IDEA cipher. In 1994, he joined r3 security engineering ag which became Entrust Technologies Switzerland since June 1998. He was at S.W.I.S. Group, Switzerland since 2001. He has published a book “On the Design and Security of Block Ciphers” (Hartung-Gorre Verlag, 1992) and more than 50 papers.

**Sheng-Li Liu (刘胜利)** received her B.S. degree, M.S. degree and Ph.D. degree from Xidian University in 1995, 1998 and 2000 respectively. From 2000 till 2002, she continued her research on cryptography and got another Ph.D. degree at Technische Universiteit Eindhoven, the Netherlands. Since 2002, she has been working at the Department of Computer Science and Engineering, Shanghai Jiaotong University. She is now a professor and her research interests include ID-based cryptography, pairing-based cryptosystems, and information-theoretic security.

**Ji-Guo Li (李继国)** received his M.S. and Ph.D. degree both from Harbin Institute of Technology; he is currently a professor and doctoral supervisor of Hohai University. His research interests include cryptography theory and its application, network and information security, wireless sensor networks etc. He has published about 40 technical papers.