Anonymous Multireceiver Identity-Based Encryption

Chun-I Fan, Ling-Ying Huang, and Pei-Hsiu Ho

Abstract—Recently, many multireceiver identity-based encryption schemes have been proposed in the literature. However, none can protect the privacy of message receivers among these schemes. In this paper, we present an anonymous multireceiver identity-based encryption scheme where we adopt Lagrange interpolating polynomial mechanisms to cope with the above problem. Our scheme makes it impossible for an attacker or any other message receiver to derive the identity of a message receiver such that the privacy of every receiver can be guaranteed. Furthermore, the proposed scheme is quite receiver efficient since each of the receivers merely needs to perform twice of pairing computation to decrypt the received ciphertext. We prove that our scheme is secure against adaptive chosen plaintext attacks and adaptive chosen ciphertext attacks. Finally, we also formally show that every receiver in the proposed scheme is anonymous to any other receiver.

Index Terms—Anonymity, multireceiver encryption, pairings, identity-based encryption.

1 Introduction

In secure multireceiver communications, to prevent unauthorized accesses, messages are usually encrypted and the encryption keys had better be changed for different sessions. Whenever a new member joins a communication group, the system assigns a key to the member for future decryption operations. If some member quits the group, the system should revoke her/his decryption key and disable the decryption. The system must deal with the problem of key management effectively such that the entire communication protocol is efficient. Recently, many researchers focused on this topic and proposed many interesting protocols.

In 2001, Boneh and Franklin [4] proposed an identity-based encryption scheme with Weil pairing. In 2003, they presented a fully functional identity-based encryption scheme (BF-IBE) [4] with a technique which was proposed by Fujisaki and Okamoto [9] for improvement. Their scheme only uses the identity (ID) of a receiver as her/his public key. A sender can encrypt a message and sends the ciphertext to the receiver. Every user can select her/his ID freely, where some meaningful or easily-memorized strings are usually selected as IDs. Moreover, the problem of the authentication for public keys can also be solved if we take IDs to form the public keys. Thus, there are lots of researches [1], [7], [8], [19], [22] related to identity-based public-key cryptosystems in recent years.

Du et al. [8] presented an identity-based broadcast encryption scheme for key distribution in 2005. They improved BF-IBE by matrix operations for encryption and decryption. In 2006, Lee et al. [13] proposed three public-key broadcast encryptions with bilinear mapping. In the same year, Yang et al. [22] improved identity-based public key encryption with bilinear mapping to be identity-based broadcast encryption. Nevertheless, there are some problems about key change and no scalability in joining new members. In 2005, Wang and Wu [19] proposed an identity-based multicast encryption scheme which has a key generation center and a group center. All users do not need any computation during the rekeying process. However, the sender must be the group center. Besides, the problems of key updating were discussed frequently, but no efficient solution has been proposed. Hence, we hope to solve the problems of key renewal and make anyone be able to act as a sender.

In a multireceiver encryption environment, a sender can randomly choose receivers. Every multireceiver encryption scheme can be transformed into a broadcast encryption scheme or a multicast encryption scheme. Beak et al. [1] proposed a multireceiver identity-based key encryption along with a formal definition and security model for the kind of schemes. They proved the security in the selective-ID model using the random oracle technique [2]. Their second scheme was employed in the REACT scheme proposed by Okamoto and Pointcheval [17]. In 2006, Lu and Hu [14] presented a multirecipient public key encryption with pairing. Their scheme can be applied to broadcast sensitive information in an unsafe distributed environment. Chatterjee and Sarkar [7] proposed the first protocol to achieve sublinear ciphertext sizes in multireceiver identity-based encryption setting.

All of the multireceiver identity-based encryption schemes proposed in the literature [1], [7], [14] cannot protect the privacy of receivers or do not contain any discussion on this issue. Nevertheless, more and more users gradually pay attention to their privacy such that the issue of privacy protection is urgently desired to be addressed in most cryptographic protocols, including multireceiver identity-based encryption schemes. Although [1], [7], may be able

- *The authors are with the Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung 804, Taiwan, ROC. E-mail: cifan@faculty.nsysu.edu.tw, {eegeegfish, peyhsiu}@gmail.com.*

Manuscript received 7 Mar. 2008; revised 18 Dec. 2008; accepted 23 Nov. 2009; published online 14 Jan. 2010.
Recommended for acceptance by H. Shen.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-2008-03-0104.
Digital Object Identifier no. 10.1109/TC.2010.23.

0018-9340/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society

to achieve receiver anonymity, which still needs to be formally proved, by covering the information about the identities of the selected receivers in a ciphertext, it is inefficient since each receiver has to perform much computation to decrypt the ciphertext. In this paper, we propose a provably secure and efficient multireceiver identity-based encryption scheme which can achieve the anonymity for every receiver against any other receiver. Everyone can receive a ciphertext broadcasted by a sender, but only the receivers selected by the sender can decipher the ciphertext successfully. Besides, one can examine whether herself/himself is a selected receiver or not. Nobody, except the sender, knows who the other receivers are. Lagrange interpolating polynomial theorem [11] has been widely used in threshold schemes and traitor tracing, but the theorem is rarely applied to encryption/decryption. In this paper, we present how to design an efficient anonymous multireceiver identity-based encryption scheme based on Lagrange interpolating polynomial theorem.

A traditional encryption protocol is that a sender encrypts and sends a message to a receiver by using the receiver's public key or a secret key shared between them. In this way, the sender generates an individual ciphertext for a receiver. Thus, for n receivers, where n is a positive integer, the sender must produce n different ciphertexts for the same message. In a multireceiver encryption scheme, such as the schemes of [1], [7], [8], [13], [14], [19], [22], a sender only needs to generate one ciphertext of a message for n receivers. Hence, it can largely reduce the sender's computation and communication cost, but the receivers' identities are known to everyone. Our scheme makes it possible for a sender to produce only one ciphertext of a message for n receivers where nobody, except the sender, knows any of the receivers' identities. Not only does the proposed scheme reduce the sender's computation and communication cost but it protects the privacy of the receivers as well.

Using our scheme in pay-TV or streaming audio/video services, a service provider only generates one ciphertext for a TV program and sends the ciphertext to all of the receivers who have ordered the program. Each receiver takes her/his personal private key to decrypt the ciphertext. In some situations, such as ordering sensitive TV programs, a receiver or customer usually expects that any other receiver or customer does not know her/his identity when ordering the TV programs. Our scheme can cope with the above problem to provide anonymity such that none, except the service provider, can derive the receivers' identities. In addition, for sensitive video sharing, by using our scheme, a sender chooses a set of receivers and produces one ciphertext of a video, and then the sender broadcasts the ciphertext under the privacy protection for the receivers. Everyone gets the ciphertext, but only the receivers the sender chose are able to decrypt the ciphertext where each receiver does not know who the other receivers are.

The rest of this paper is organized as follows: In Sections 2 and 3, we introduce some preliminaries about mathematical backgrounds and security definitions. In Section 4, we present and analyze the proposed scheme. The security of our scheme is formally proved in Section 5. We evaluate the performance of the proposed scheme in Section 6. In Section 7, we compare our scheme with the others in performance and other aspects. Finally, we give a concluding remark of this paper in Section 8.

2 Preliminaries

In this section, we review the polynomial interpolation method, the characteristics of bilinear groups, and some related hard problems.

2.1 Polynomial Interpolation

Lagrange Interpolating Polynomial Theorem. *Let*

$$\sum_{i=1}^{t} F_i(x) = \sum_{i=0}^{t-1} a_i x^i$$

be a polynomial of degree $t-1 \geq 0$ that passes through the t points $(x_1, y_1), (x_2, y_2), \ldots, (x_t, y_t)$ where for each i,

$$F_i(x) = y_i \prod_{1 \leq j \neq i \leq t} \frac{x - x_j}{x_i - x_j} = \begin{cases} y_i, & \text{if } x = x_i. \\ 0, & \text{if } x \in \{x_1, \ldots, x_t\} - \{x_i\}. \end{cases}$$

2.2 Bilinear Groups

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of prime order q and let P be a generator of $\mathbb{G}_1$. A bilinear mapping $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ has the following properties:

1. Bilinearity: $e(aP, bQ) = e(P, Q)^{ab}$, $\forall P$, $Q \in \mathbb{G}_1$ and a, $b \in \mathbb{Z}_q$.
2. Nondegeneracy: There exist P, $Q \in \mathbb{G}_1$ such that $e(P, Q) \neq 1$.
3. Computability: There exists an efficient algorithm to compute $e(P, Q)$ for any P, $Q \in \mathbb{G}_1$.

The above properties also imply:

$$e(P+Q, R) = e(P, R) \cdot e(Q, R), \forall P, Q, R \in \mathbb{G}_1.$$
$$e(P, Q+R) = e(P, Q) \cdot e(P, R), \forall P, Q, R \in \mathbb{G}_1.$$

2.3 Hard Problems

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups of the same order q. Let $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a bilinear mapping and let P be a generator of $\mathbb{G}_1$.

The Computational Diffie-Hellman (CDH) Problem. *Given $<P, aP, bP>$ for some a, $b \in \mathbb{Z}_q^*$, compute abP.*

The Bilinear Diffie-Hellman (BDH) Problem. *Given $<P, aP, bP, cP>$ for some a, b, $c \in \mathbb{Z}_q^*$, compute $e(P, P)^{abc} \in \mathbb{G}_2$.*

The Co-Bilinear Diffie-Hellman (Co-BDH) Problem [23]. *Given $<P, aP, bP, Q>$ for some $a, b \in \mathbb{Z}_q^*$ and $Q \in_R \mathbb{G}_1$, compute $e(P, Q)^{ab}$.*

The Co-Decisional Bilinear Diffie-Hellman (Co-DBDH) Problem [20]. *Given $<P, aP, bP, Q, \mathcal{Z}>$ for some $a, b \in \mathbb{Z}_q^*$, $Q \in_R \mathbb{G}_1$ and $\mathcal{Z} \in_R \mathbb{G}_2$, decide if $\mathcal{Z} = e(P, Q)^{ab}$.*

The Modified Decisional Bilinear Diffie-Hellman (DBDH-M) Problem [6]. *Given $<P, aP, bP, \mathcal{U}>$ for some $a, b \in \mathbb{Z}_q^*$ and $\mathcal{U} \in_R \mathbb{G}_1$, decide if $\mathcal{U} = ab^2P$.*

Definition 1 (The Co-DBDH Assumption). *We define that an algorithm $\mathcal{B}$ with an output $\beta \in \{0, 1\}$ has advantage ε in solving the Co-DBDH problem if*

$$\begin{aligned} |Pr[\mathcal{B}(P, aP, bP, Q, e(P, Q)^{ab}) = 1] \\ - Pr[\mathcal{B}(P, aP, bP, Q, \mathcal{Z}) = 1]| \geq \varepsilon \end{aligned}$$

where the probability is over the random choice of $a, b \in \mathbb{Z}_q^*$ and the random choice of $Q \in \mathbb{G}_1$ and $\mathcal{Z} \in \mathbb{G}_2$. We say that the (τ, ε)-Co-DBDH assumption holds if no polynomial-time algorithm has advantage ε within running time τ in solving the Co-DBDH problem.

Definition 2 (The DBDH-M Assumption). *We define that an algorithm $\mathcal{B}$ with an output $\beta \in \{0, 1\}$ has advantage ε in solving the DBDH-M problem if*

$$|Pr[\mathcal{B}(P, aP, bP, ab^2P) = 1] \\ - Pr[\mathcal{B}(P, aP, bP, \mathcal{U}) = 1]| \geq \varepsilon$$

where the probability is over the random choice of $a, b \in \mathbb{Z}_q^*$ and the random choice of $\mathcal{U} \in \mathbb{G}_1$. We say that the (τ, ε)-DBDH-M assumption holds if no polynomial-time algorithm has advantage ε within running time τ in solving the DBDH-M problem.

3 Security Definitions

According to [1], [3], [5], [7], [10], we present a general model and security notions for anonymous multireceiver IBE (Identity-Based Encryption) schemes. The security notions are "Indistinguishability of encryptions under selective multi-ID, chosen plaintext attacks" (IND-sMID-CPA) [1], "Indistinguishability of encryptions under selective multi-ID, chosen ciphertext attacks" (IND-sMID-CCA) [1], "Anonymous indistinguishability of encryptions under selective-ID, chosen plaintext attacks" (ANON-sID-CPA) [3], and "Anonymous indistinguishability of encryptions under selective-ID, chosen ciphertext attacks" (ANON-sID-CCA) [3]. They are described as follows:

Definition 3 (Multireceiver IBE). *A general multireceiver IBE scheme contains a set of four algorithms:* Setup, Extract, Encrypt, *and* Decrypt.

- **Setup**: *The Private Key Generator (PKG) runs this algorithm to generate a master key s and PKG's public parameters params which include a description of the groups $\mathbb{G}_1$, $\mathbb{G}_2$, and the mapping e. Note that the PKG's public parameters params are publicly known while the master key s is kept secret.*
- **Extract**: *This is a private key extraction algorithm. Providing an identity ID_i received from a user, the PKG's master key s, and public parameters params as inputs, the PKG runs this algorithm to generate a private key d_i associated with ID_i. We also denote $d_i = Extract(params, s, ID_i)$. The identity ID_i is used as a public key while d_i is the corresponding private key.*
- **Encrypt**: *Providing multiple identities $(ID_1, \dots, ID_t)$ of the receivers, the PKG's public parameters params, and a plaintext message M as inputs, the sender runs this algorithm to generate a ciphertext C which is an encryption of M. We denote $C = Encrypt(params, ID_1, \dots, ID_t, M)$.*
- **Decrypt**: *Providing a ciphertext C, the PKG's public parameters params, an identity ID_i, and the private key d_i as inputs, the receiver with identity ID_i runs this algorithm to obtain a decryption result D, which is either a certain plaintext message or not. We denote $D = Decrypt(params, C, ID_i, d_i)$.*

Definition 4 (IND-sMID-CPA). *Let $\mathcal{A}$ be a polynomial-time attacker. Let $\prod$ be a general multireceiver IBE scheme. $\mathcal{A}$ interacts with a Challenger in the following game:*

- **Setup**: *The Challenger runs the* Setup *algorithm. It gives the attacker $\mathcal{A}$ the resulting public parameters $params$. It keeps the master key secret.*
- **Phase 1**: *$\mathcal{A}$ outputs target multiple identities $(ID_1, \dots, ID_t)$ where t is a positive integer.*
- **Phase 2**: *$\mathcal{A}$ issues private key extraction queries. Upon receiving a private key extraction query, denoted by ID_j, the Challenger runs the private key extraction algorithm to get $d_j = Extract(params, s, ID_j)$. The only constraint is that $ID_j \neq ID_i$ for $i = 1, \dots, t$.*
- **Challenge**: *$\mathcal{A}$ outputs a target plaintext pair (M_0, M_1). Upon receiving (M_0, M_1), the Challenger randomly chooses $\beta \in \{0, 1\}$ and creates a target ciphertext $C = Encrypt(params, ID_1, \dots, ID_t, M_\beta)$. Then the Challenger returns C to $\mathcal{A}$.*
- **Phase 3**: *$\mathcal{A}$ issues private key extraction queries as those in* Phase 2.
- **Guess**: *Finally, $\mathcal{A}$ outputs its guess $\beta' \in \{0, 1\}$ and wins the game if $\beta' = \beta$.*

Such an attacker $\mathcal{A}$ is referred to as an IND-sMID-CPA attacker. As we did above, we define $\mathcal{A}$'s guessing advantage

$$\mathbf{Adv}_{\prod}^{\text{IND-sMID-CPA}}(\mathcal{A}) = |Pr[\beta = \beta'] - \frac{1}{2}|.$$

The scheme $\prod$ is said to be (τ, ε)-IND-sMID-CPA secure if for any IND-sMID-CPA attacker $\mathcal{A}$, within polynomial running time τ, the guessing advantage $\mathbf{Adv}_{\prod}^{\text{IND-sMID-CPA}}(\mathcal{A})$ is less than ε.

Definition 5 (IND-sMID-CCA). *Let $\mathcal{A}$ be a polynomial-time attacker. Let $\prod$ be a general multireceiver IBE scheme. $\mathcal{A}$ interacts with a Challenger in the following game:*

- **Setup**: *The Challenger runs the* Setup *algorithm. It gives the attacker $\mathcal{A}$ the resulting public parameters $params$. It keeps the master key secret.*
- **Phase 1**: *$\mathcal{A}$ outputs target multiple identities $(ID_1, \dots, ID_t)$ where t is a positive integer.*
- **Phase 2**: *$\mathcal{A}$ issues private key extraction queries. Upon receiving a private key extraction query, denoted by ID_j, the Challenger runs the private key extraction algorithm to get $d_j = Extract(params, s, ID_j)$. The only constraint is that $ID_j \neq ID_i$ for $i = 1, \dots, t$.*
- **Phase 3**: *$\mathcal{A}$ issues decryption queries for target identities. Upon receiving a decryption query, denoted by (C^*, ID_i) for some $i \in \{1, \dots, t\}$, the Challenger generates a private key associated with ID_i, which is denoted by d_i. The Challenger returns $D = Decrypt(params, C^*, ID_i, d_i)$ to $\mathcal{A}$.*
- **Challenge**: *$\mathcal{A}$ outputs a target plaintext pair (M_0, M_1). Upon receiving (M_0, M_1), the Challenger randomly chooses $\beta \in \{0, 1\}$ and creates a target ciphertext $C = Encrypt(params, ID_1, \dots, ID_t, M_\beta)$. Then the Challenger returns C to $\mathcal{A}$.*
- **Phase 4**: *$\mathcal{A}$ issues private key extraction queries as those in* Phase 2 *and decryption queries for target identities as those in* Phase 3 *where a restriction here is that $C^* \neq C$.*

- **Guess**: *Finally, $\mathcal{A}$ outputs its guess $\beta' \in \{0,1\}$ and wins the game if $\beta' = \beta$.*

Such an attacker $\mathcal{A}$ is referred to as an IND-sMID-CCA attacker. As we did above, we define $\mathcal{A}$'s guessing advantage

$$\mathbf{Adv}_{\prod}^{\text{IND-sMID-CCA}}(\mathcal{A}) = |Pr[\beta = \beta'] - \frac{1}{2}|.$$

The scheme $\prod$ is said to be (τ, ε)-IND-sMID-CCA secure if for any IND-sMID-CCA attacker $\mathcal{A}$, within polynomial running time τ, the guessing advantage

$$\mathbf{Adv}_{\prod}^{\text{IND-sMID-CCA}}(\mathcal{A})$$

is less than ε.

Definition 6 (ANON-sID-CPA). *Let $\mathcal{A}$ be a polynomial-time attacker. Let $\prod$ be a general multireceiver IBE scheme. $\mathcal{A}$ interacts with a Challenger in the following game:*

- **Setup**: *The Challenger runs the* Setup *algorithm. It gives the attacker $\mathcal{A}$ the resulting public parameters params. It keeps the master key secret.*
- **Phase 1**: *$\mathcal{A}$ outputs a target identity pair (ID_1, ID_2). Upon receiving (ID_1, ID_2), the Challenger randomly chooses $\beta \in \{1,2\}$.*
- **Phase 2**: *$\mathcal{A}$ issues private key extraction queries. Upon receiving a private key extraction query, denoted by ID_j, the Challenger runs the private key extraction algorithm to get $d_j = Extract(params, s, ID_j)$. The only constraint is that $ID_j \neq ID_i$ for $i = 1, 2$.*
- **Challenge**: *$\mathcal{A}$ outputs a target plaintext M. The Challenger creates a target ciphertext $C = Encrypt$ $(params, ID_\beta, M)$ and then returns C to $\mathcal{A}$.*
- **Phase 3**: *$\mathcal{A}$ issues private key extraction queries as those in* Phase 2.
- **Guess**: *Finally, $\mathcal{A}$ outputs its guess $\beta' \in \{1,2\}$ and wins the game if $\beta' = \beta$.*

Such an attacker $\mathcal{A}$ is referred to as an ANON-sID-CPA attacker. As we did above, we define $\mathcal{A}$'s guessing advantage

$$\mathbf{Adv}_{\prod}^{\text{ANON-sID-CPA}}(\mathcal{A}) = |Pr[\beta = \beta'] - \frac{1}{2}|.$$

The scheme $\prod$ is said to be (τ, ε)-ANON-sID-CPA secure if for any ANON-sID-CPA attacker $\mathcal{A}$, within polynomial running time τ, the guessing advantage

$$\mathbf{Adv}_{\prod}^{\text{ANON-sID-CPA}}(\mathcal{A})$$

is less than ε.

Definition 7 (ANON-sID-CCA). *Let $\mathcal{A}$ be a polynomial-time attacker. Let $\prod$ be a general multireceiver IBE scheme. $\mathcal{A}$ interacts with a Challenger in the following game:*

- **Setup**: *The Challenger runs the* Setup *algorithm. It gives the attacker $\mathcal{A}$ the resulting public parameters params. It keeps the master key secret.*
- **Phase 1**: *$\mathcal{A}$ outputs a target identity pair (ID_1, ID_2). Upon receiving (ID_1, ID_2), the Challenger randomly chooses $\beta \in \{1,2\}$.*
- **Phase 2**: *$\mathcal{A}$ issues private key extraction queries. Upon receiving a private key extraction query, denoted by ID_j, the Challenger runs the private key extraction algorithm to get $d_j = Extract(params, s, ID_j)$. The only constraint is that $ID_j \neq ID_i$ for $i = 1, 2$.*
- **Phase 3**: *$\mathcal{A}$ issues decryption queries for target identities. Upon receiving a decryption query, denoted by (C^*, ID_i) for some $i \in \{1,2\}$, the Challenger generates a private key associated with ID_i, which is denoted by d_i. The Challenger returns $D = Decrypt(params, C^*, ID_i, d_i)$ to $\mathcal{A}$.*
- **Challenge**: *$\mathcal{A}$ outputs a target plaintext M. The Challenger creates a target ciphertext $C = Encrypt(params, ID_\beta, M)$ and then returns C to $\mathcal{A}$.*
- **Phase 4**: *$\mathcal{A}$ issues private key extraction queries as those in* Phase 2 *and decryption queries for target identities as those in* Phase 3 *where a restriction here is that $C^* \neq C$.*
- **Guess**: *Finally, $\mathcal{A}$ outputs its guesses $\beta' \in \{1,2\}$ and wins the game if $\beta' = \beta$.*

Such an attacker $\mathcal{A}$ is referred to as an ANON-sID-CCA attacker. As we did above, we define $\mathcal{A}$'s guessing advantage

$$\mathbf{Adv}_{\prod}^{\text{ANON-sID-CCA}}(\mathcal{A}) = |Pr[\beta = \beta'] - \frac{1}{2}|.$$

The scheme $\prod$ is said to be (τ, ε)-ANON-sID-CCA secure if for any ANON-sID-CCA attacker $\mathcal{A}$, within polynomial running time τ, the guessing advantage

$$\mathbf{Adv}_{\prod}^{\text{ANON-sID-CCA}}(\mathcal{A})$$

is less than ε.

4 Our Proposal

In this section, based on BF-IBE scheme [4], we present an anonymous multireceiver identity-based encryption scheme. Our scheme adopts bilinear pairings on elliptic curves. Let $\mathbb{G}_1$ be an additive group and $\mathbb{G}_2$ be a multiplicative group where both of them are cyclic and each of them is with prime order q. Let P be a randomly chosen generator of $\mathbb{G}_1$ and e be a bilinear mapping such that $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.

In our scheme, a sender chooses t receivers and prepares t points (x_1, y_1), $(x_2, y_2), \ldots, (x_t, y_t)$ for them. For every receiver i, the sender sets x_i as the root of $F_i(x) = y_i$ where the receiver's identity, ID_i, is mapped into x_i in $\mathbb{Z}_q^*$, and then computes $y_i = yQ_i$ as the personal private key of the receiver where y is randomly chosen in $\mathbb{Z}_q^*$ and ID_i is also mapped into Q_i in $\mathbb{G}_1^*$. $\mathbb{G}_1^*$ denotes the set $\mathbb{G}_1/\{\mathbb{O}\}$, and $\mathbb{O}$ is the identity element in the additive group $\mathbb{G}_1$.

The polynomial

$$f_i(x) = F_i(x)/y_i = \prod_{1 \leq j \neq i \leq t} \frac{x - x_j}{x_i - x_j} = \begin{cases} 1, & \text{if } x = x_i \\ 0, & \text{if } x \in \{x_1, \ldots, x_t\} - \{x_i\} \end{cases}$$

is used for achieving receiver anonymity. In the *Encrypt* phase of our scheme, the sender computes a parameter R_i for each receiver i by using the above polynomial. The sender takes all R_i's and the other parameters to form a ciphertext encrypted by a secret key σ and then broadcasts it. To decrypt the ciphertext, receiver i takes all R_i's and her/his x_i to reconstruct $\lambda = F_i(x_i)$, which is y_i. Then, the receiver computes the secret key σ via her/his private key and λ. Finally, the receiver can decrypt the ciphertext by using σ.

4.1 The Proposed Scheme

Setup: The algorithm works as follows:

1. Pick an integer $s \in \mathbb{Z}_q^*$ and an element $P_1 \in \mathbb{G}_1$ at random.
2. Set $P_{pub} = sP$.
3. Choose five cryptographic one-way hash functions

$$H : \{0,1\}^* \to \mathbb{Z}_q^*, H_1 : \{0,1\}^* \to \mathbb{G}_1^*,$$
$$H_2 : \mathbb{G}_2 \to \{0,1\}^w, H_3 : \{0,1\}^w \times \{0,1\}^* \to \mathbb{Z}_q^*,$$
$$H_4 : \{0,1\}^w \to \{0,1\}^w$$

for some positive integer w. The symmetric encryption and decryption functions with a key k are represented by E_k and D_k, respectively.

4. Publish the system parameters $params = <q, \mathbb{G}_1, \mathbb{G}_2, e, n, P, P_1, P_{pub}, H, H_1, H_2, H_3, H_4>$ and keep the master key s secret.

Extract: Input $params$, s, and an identity $ID_i \in \{0,1\}^*$ for $i \in [1, n]$. The algorithm performs the following tasks:

1. Compute $Q_i = H_1(ID_i) \in \mathbb{G}_1^*$.
2. Set the private key $d_i = s(P_1 + Q_i)$.

Encrypt: Input $params$, a plaintext message M, and select t, $1 \le t \le n$, identities $(ID_1, \ldots, ID_t)$ of the receivers whom the sender wants to send the ciphertext of M to. The algorithm performs the following tasks:

1. Pick a string $\sigma \in \{0,1\}^w$ at random and set $r = H_3(\sigma, M)$.
2. Pick an integer $\alpha \in \mathbb{Z}_q^*$ at random and set $y = \alpha^{-1} r \bmod q$.
3. For $i = 1, \ldots, t$, compute $x_i = H(ID_i)$ and $Q_i = H_1(ID_i)$.
4. For $i = 1, \ldots, t$, compute

$$f_i(x) = \prod_{1 \le j \ne i \le t} \frac{x - x_j}{x_i - x_j} = a_{i,1} + a_{i,2}x + \cdots + a_{i,t}x^{t-1}$$

where $a_{i,1}, \ldots, a_{i,t} \in \mathbb{Z}_q$.

5. For $i = 1, \ldots, t$, compute

$$R_i = \sum_{j=1}^{t} a_{j,i} y Q_j = \sum_{j=1}^{t} b_j Q_j$$

where $b_j = a_{j,i} y \in \mathbb{Z}_q$.

6. Set the ciphertext $C = <R_1, \ldots, R_t, rP, \alpha P_{pub}, \sigma \oplus H_2(e(P_{pub}, P_1)^r), E_{H_4(\sigma)}(M)>$.

Decrypt: Input the ciphertext $C = <R_1, \ldots, R_t, U_1, U_2, V, W>$, $params$, an identity ID_i, and the private key d_i of the receiver with identity ID_i. To decrypt C, the algorithm performs the following tasks:

1. Compute $x_i = H(ID_i)$.
2. Set $\lambda = R_1 + x_i R_2 + \cdots + (x_i{}^{t-1} \bmod q) R_t$.
3. Compute

$$\sigma' = V \oplus H_2\left(\frac{e(U_1, d_i)}{e(U_2, \lambda)}\right).$$

4. Compute $M' = D_{H_4(\sigma')}(W)$.
5. Set $r' = H_3(\sigma', M')$. Test whether $U_1 = r'P$ or not. If true, accept the plaintext message M', i.e., $M' = M$; otherwise, reject the ciphertext.

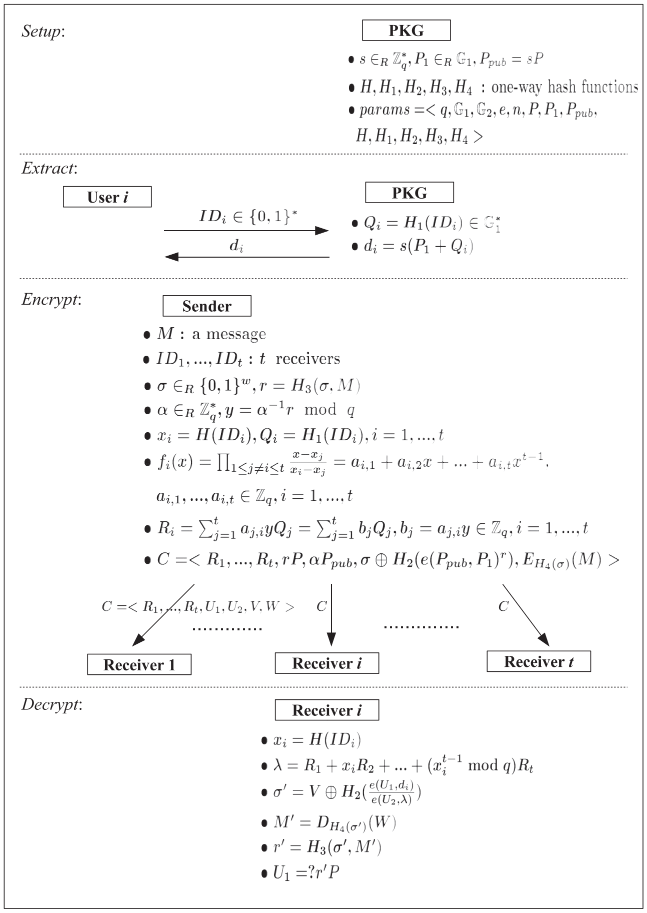
Setup:
PKG
$\bullet\ s \in_R \mathbb{Z}_q^*, P_1 \in_R \mathbb{G}_1, P_{pub} = sP$
$\bullet\ H, H_1, H_2, H_3, H_4$: one-way hash functions
$\bullet\ params =< q, \mathbb{G}_1, \mathbb{G}_2, e, n, P, P_1, P_{pub}, H, H_1, H_2, H_3, H_4 >$
Extract:
User i
PKG
$ID_i \in \{0,1\}^*$
d_i
$\bullet\ Q_i = H_1(ID_i) \in \mathbb{G}_1^*$
$\bullet\ d_i = s(P_1 + Q_i)$
Encrypt:
Sender
$\bullet\ M$: a message
$\bullet\ ID_1, ..., ID_t$: t receivers
$\bullet\ \sigma \in_R \{0,1\}^w, r = H_3(\sigma, M)$
$\bullet\ \alpha \in_R \mathbb{Z}_q^*, y = \alpha^{-1} r \mod q$
$\bullet\ x_i = H(ID_i), Q_i = H_1(ID_i), i = 1, ..., t$
$\bullet\ f_i(x) = \prod_{1 \le j \ne i \le t} \frac{x - x_j}{x_i - x_j} = a_{i,1} + a_{i,2}x + ... + a_{i,t}x^{t-1}$, $a_{i,1}, ..., a_{i,t} \in \mathbb{Z}_q, i = 1, ..., t$
$\bullet\ R_i = \sum_{j=1}^{t} a_{j,i} y Q_j = \sum_{j=1}^{t} b_j Q_j, b_j = a_{j,i} y \in \mathbb{Z}_q, i = 1, ..., t$
$\bullet\ C =< R_1, ..., R_t, rP, \alpha P_{pub}, \sigma \oplus H_2(e(P_{pub}, P_1)^r), E_{H_4(\sigma)}(M) >$
$C =< R_1, ..., R_t, U_1, U_2, V, W >$
C
C
Receiver 1
Receiver i
Receiver t
Decrypt:
Receiver i
$\bullet\ x_i = H(ID_i)$
$\bullet\ \lambda = R_1 + x_i R_2 + ... + (x_i^{t-1} \mod q) R_t$
$\bullet\ \sigma' = V \oplus H_2(\frac{e(U_1, d_i)}{e(U_2, \lambda)})$
$\bullet\ M' = D_{H_4(\sigma')}(W)$
$\bullet\ r' = H_3(\sigma', M')$
$\bullet\ U_1 =? r'P$

Fig. 1. The proposed anonymous multireceiver identity-based encryption scheme.

A PKG is established to run ***Setup***. When a user gives her/his identity to the PKG, the PKG inputs its public system parameters, the master key, and the user's identity to ***Extract*** and returns a private key to the user. Users who have obtained their private keys from the PKG are called members. A user who sends out a message is said to be a sender. A sender can input the PKG's system parameters, the identities of selected members, and a plaintext message to ***Encrypt*** to get a ciphertext and then broadcasts it. The members the sender designated are called the receivers. When receiving a ciphertext, every member can input the PKG's system parameters, the ciphertext, her/his identity, and her/his own private key to ***Decrypt***. If the member is a receiver, then ***Decrypt*** returns the plaintext message; else it returns "reject". The proposed scheme is also illustrated in Fig. 1.

4.2 Correctness

The decryption of our scheme is justified by the following: For each i with $1 \le i \le t$, we have that

$$\begin{aligned}\lambda &= R_1 + x_i R_2 + \cdots + x_i^{i-1} R_i + \cdots + x_i^{t-1} R_t \\ &= (a_{1,1} y Q_1 + \cdots + a_{t,1} y Q_t) \\ &\quad + (x_i a_{1,2} y Q_1 + \cdots + x_i a_{t,2} y Q_t) + \cdots \\ &\quad + (x_i^{i-1} a_{1,i} y Q_1 + \cdots + x_i^{i-1} a_{t,i} y Q_t) + \cdots \\ &\quad + (x_i^{t-1} a_{1,t} y Q_1 + \cdots + x_i^{t-1} a_{t,t} y Q_t) \\ &= (a_{1,1} + a_{1,2} x_i + \cdots + a_{1,t} x_i^{t-1}) y Q_1 \\ &\quad + (a_{2,1} + a_{2,2} x_i + \cdots + a_{2,t} x_i^{t-1}) y Q_2 + \cdots \\ &\quad + (a_{i,1} + a_{i,2} x_i + \cdots + a_{i,t} x_i^{t-1}) y Q_i + \cdots \\ &\quad + (a_{t,1} + a_{t,2} x_i + \cdots + a_{t,t} x_i^{t-1}) y Q_t \\ &= y Q_i (\text{by Lagrange interplating polynomial theorem})\end{aligned}$$

$$\begin{aligned}\frac{e(U_1, d_i)}{e(U_2, \lambda)} &= \frac{e(rP, s(Q_i + P_1))}{e(\alpha P_{pub}, y Q_i)} = \frac{e(rP, sQ_i) e(rP, sP_1)}{e(P_{pub}, Q_i)^{\alpha y}} \\ &= \frac{e(P_{pub}, Q_i)^r e(P_{pub}, P_1)^r}{e(P_{pub}, Q_i)^r} = e(P_{pub}, P_1)^r.\end{aligned}$$

Thus,

$$\sigma' = V \oplus H_2\left(\frac{e(U_1, d_i)}{e(U_2, \lambda)}\right) = V \oplus H_2(e(P_{pub}, P_1)^r) = \sigma \text{ and}$$
$$M' = D_{H_4(\sigma')}(W) = D_{H_4(\sigma)}(E_{H_4(\sigma)}(M)) = M.$$

4.3 Discussions

The proposed scheme is based on some hard problems such that it satisfies forward security, backward security, and anonymity. The details are described as follows:

Forward secrecy [16]. The members who have quit the group should not be able to know the later session keys. In our scheme, the session key σ is randomly chosen in every session. If the members of the group are different from before, the sender will compute $f_i(x)'s$ again such that the members who have quit the group cannot obtain any useful information to compute new session keys.

Backward secrecy [16]. New members should not be able to know the session keys generated before they join the group. The discussions of this case are similar to those of forward secrecy.

Receiver anonymity. Everyone can receive transmitted messages because they are broadcasted. Only the designated receivers can decrypt them successfully. Every receiver knows whether herself/himself is one of the designated receivers, but she/he cannot determine the others.

5 Security Proofs

We now prove that our scheme is IND-sMID-CPA secure, IND-sMID-CCA secure, ANON-sID-CPA secure, and ANON-sID-CCA secure under the Co-DBDH assumption and the DBDH-M assumption.

5.1 Confidentiality

Semantic security is a necessary security requirement for identity-based encryptions. It means that no useful information about a plaintext message can be gleaned from the corresponding ciphertext.

Theorem 1. *The proposed multireceiver IBE scheme is $(\tau, q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}, q_1, q_2, \varepsilon)$-IND-sMID-CCA secure under the (τ', ε')-Co-DBDH assumption, where $\varepsilon' \geq \varepsilon$ and $\tau' \approx \tau + (q_{H_1} + q_{H_3} + q_1)\mathcal{O}(\tau_1) + q_2\mathcal{O}(\tau_1 + \tau_2) + (q_H + q_{H_2} + q_{H_4})\mathcal{O}(1)$. ($q_1, q_2, q_H, q_{H_1}, q_{H_2}, q_{H_3}$, and q_{H_4} denote the number of private key extraction queries, decryption queries, and queries to the hash functions H, H_1, H_2, H_3, H_4, respectively. τ_1 and τ_2 denote the computing time for a multiplication in $\mathbb{G}_1$ and a pairing e, respectively.)*

Proof. Assume that an IND-sMID-CCA attacker $\mathcal{A}$ has advantage

$$\mathbf{Adv}_{\prod}^{\text{IND-sMID-CCA}}(\mathcal{A}) \geq \varepsilon$$

within running time τ where $\prod$ is the proposed scheme. Suppose that $\mathcal{A}$ makes at most q_1 private key extraction queries, at most q_2 decryption queries, and at most $q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}$ queries to the hash functions H, H_1, H_2, H_3, H_4, respectively. We will show how an algorithm $\mathcal{B}$ can solve the Co-DBDH problem with advantage ε' within running time τ', where $\varepsilon' \geq \varepsilon$ and $\tau' \approx \tau + (q_{H_1} + q_{H_3} + q_1)\mathcal{O}(\tau_1) + q_2\mathcal{O}(\tau_1 + \tau_2) + (q_H + q_{H_2} + q_{H_4})\mathcal{O}(1)$, by utilizing $\mathcal{A}$.

First, $\mathcal{B}$ is given $<q, \mathbb{G}_1, \mathbb{G}_2, e, P, aP, bP, Q, \mathcal{Z}>$ as an instance of the Co-DBDH problem. $\mathcal{B}$ can simulate the Challenger to execute each phase of the IND-sMID-CCA game (**Definition 5**) for $\mathcal{A}$ as follows:

***Phase* 1**: Suppose that $\mathcal{A}$ outputs target multiple identities $(ID_1, \ldots, ID_t)$ where t is a positive integer.

Setup: $\mathcal{B}$ sets $P_1 = Q$ and $P_{pub} = bP$, and gives the attacker $\mathcal{A}$ $<q, \mathbb{G}_1, \mathbb{G}_2, e, n, P, P_1, P_{pub}, H, H_1, H_2, H_3, H_4>$ as the public parameters of the proposed scheme where H, H_1, H_2, H_3, and H_4 are random oracles controlled by $\mathcal{B}$ as follows:

Let H**List**, H_1**List**, H_2**List**, H_3**List**, and H_4**List** be used for storing the results of querying H, H_1, H_2, H_3, and H_4, respectively.

H**-query.** Input an identity ID_j to H where $j \in [1, n]$: If there exists (ID_j, x_j) in H**List**, return x_j. Otherwise, do the following:

1. Pick an integer $x_j \in \mathbb{Z}_q^*$ at random.
2. Put (ID_j, x_j) in H**List**.
3. Return x_j.

H_1**-query.** Input an identity ID_j to H_1 where $j \in [1, n]$: If there exists (ID_j, l_j, Q_j) in H_1**List**, return Q_j. Otherwise, do the following:

1. Pick an integer $l_j \in \mathbb{Z}_q^*$ at random.
2. If $ID_j = ID_i$ for some $i \in \{1, \ldots, t\}$, then compute $Q_j = l_j P$; else compute $Q_j = l_j P - P_1$.
3. Put (ID_j, l_j, Q_j) in H_1**List**.
4. Return Q_j.

H_2**-query.** Input an element Z_j in $\mathbb{G}_2$ to H_2 where $j \in [1, q_{H_2}]$: If there exists (Z_j, δ_j) in H_2**List**, return δ_j. Otherwise, do the following:

1. Pick a string $\delta_j \in \{0, 1\}^w$ at random.
2. Put (Z_j, δ_j) in H_2**List**.
3. Return δ_j.

H_3**-query.** Input a pair (σ_j, M_j) to H_3 where $j \in [1, q_{H_3}]$: If there exists $(\sigma_j, M_j, \rho_j, \Gamma_j)$ in H_3**List**, return ρ_j. Otherwise, do the following:

1. Pick an integer $\rho_j \in \mathbb{Z}_q^*$ at random and compute $\Gamma_j = \rho_j P$.
2. Put $(\sigma_j, M_j, \rho_j, \Gamma_j)$ in H_3**List**.
3. Return ρ_j.

H_4**-query.** Input a string σ_j to H_4 where $j \in [1, q_{H_4}]$: If there exists (σ_j, ζ_j) in H_4**List**, return ζ_j. Otherwise, do the following:

1. Pick a string $\zeta_j \in \{0,1\}^w$ at random.
2. Put (σ_j, ζ_j) in H_4**List**.
3. Return ζ_j.

***Phase* 2**: $\mathcal{A}$ issues private key extraction queries. Upon receiving a private key extraction query, denoted by ID_j with $ID_j \neq ID_i$ for $i \in \{1, \ldots, t\}$, $\mathcal{B}$ does the following:

1. If there exists (ID_j, l_j, Q_j) in H_1**List**, then compute
$$\begin{aligned} d_j &= l_j P_{pub} (= l_j bP = bl_j P = b(l_j P - P_1 + P_1) \\ &= b(Q_j + P_1)); \end{aligned}$$
else randomly choose $l_j \in \mathbb{Z}_q^*$, compute $d_j = l_j P_{pub}$, $Q_j = l_j P - P_1$, and put (ID_j, l_j, Q_j) in H_1**List**.
2. Return d_j to $\mathcal{A}$.

***Phase* 3**: $\mathcal{A}$ issues decryption queries for target identities. Upon receiving a decryption query, denoted by (C^*, ID_i) where $i \in \{1, \ldots, t\}$ and $C^* = <R_1, \ldots, R_t, U_1, U_2, V, W>$, $\mathcal{B}$ does the following:

1. Search H_3**List** to get (M_j, ρ_j) when $\Gamma_j = U_1$. If not found, return "reject" to $\mathcal{A}$.
2. Compute $x_i = H(ID_i)$.
3. Compute $\lambda = R_1 + x_i R_2 + \cdots + (x_i{}^{t-1} \bmod q) R_t$.
4. Compute
$$\sigma' = V \oplus H_2\left(\frac{e(P_{pub}, \rho_j P_1) e(U_1, l_i P_{pub})}{e(U_2, \lambda)}\right),$$
where
$$\begin{aligned} e(P_{pub}, \rho_j P_1) e(U_1, l_i P_{pub}) &= e(bP, \rho_j P_1) e(U_1, l_i bP) \\ &= e(\rho_j P, bP_1) e(U_1, bl_i P) = e(U_1, b(P_1 + Q_i)) \\ &= e(U_1, d_i). \end{aligned}$$
5. Test whether $M_j = D_{H_4(\sigma')}(W)$ or not. If not, return "reject" to $\mathcal{A}$; else return M_j to $\mathcal{A}$.

Challenge: $\mathcal{A}$ outputs a target plaintext pair (M_0, M_1). Upon receiving (M_0, M_1), $\mathcal{B}$ does the following:

1. Randomly choose $\beta \in \{0,1\}$.
2. For $i = 1, \ldots, t$, search H_1**List** to get l_i that corresponds to ID_i.
3. Pick an integer $\alpha \in \mathbb{Z}_q^*$ and a string $\sigma \in \{0,1\}^w$ at random.
4. Set $U_1 = aP = rP$ and $\mathcal{K} = \mathcal{Z}$.
5. For $i = 1, \ldots, t$, compute
$$f_i(x) = \prod_{1 \le j \ne i \le t} \frac{x - x_j}{x_i - x_j} = a_{i,1} + a_{i,2} x + \cdots + a_{i,t} x^{t-1}$$
where $a_{i,1}, \ldots, a_{i,t} \in \mathbb{Z}_q$.
6. For $i = 1, \ldots, t$, compute
$$\begin{aligned} R_i &= \sum_{j=1}^{t} a_{j,i} \alpha^{-1} l_j U_1 \left(= \sum_{j=1}^{t} a_{j,i} \alpha^{-1} r l_j P = \sum_{j=1}^{t} a_{j,i} y Q_j \right. \\ &\left. = \sum_{j=1}^{t} b_j Q_j \right). \end{aligned}$$
7. Create a target ciphertext $C = <R_1, \ldots, R_t$, U_1, αP_{pub}, $\sigma \oplus H_2(\mathcal{K})$, $E_{H_4(\sigma)}(M_\beta)>$.
8. Return C to $\mathcal{A}$.

***Phase* 4**: $\mathcal{A}$ issues private key extraction queries as those in *Phase* 2 and decryption queries for target identities as those in *Phase* 3 where a restriction here is that $C^* \neq C$.

Guess: Finally, $\mathcal{A}$ outputs its guess $\beta' \in \{0,1\}$. If $\beta' = \beta$, then $\mathcal{B}$ outputs 1. Otherwise, $\mathcal{B}$ outputs 0.

If $\mathcal{K} = e(P,Q)^{ab}$, then $\sigma \oplus H_2(\mathcal{K}) = \sigma \oplus H_2(e(bP,Q)^a) = \sigma \oplus H_2(e(P_{pub}, P_1)^r)$. Hence, C is a valid ciphertext. Otherwise, $\mathcal{K}$ is a randomly chosen element of $\mathbb{G}_2$. As the construction above, $\mathcal{B}$ successfully simulates the random oracles $\{H, H_1, H_2, H_3, H_4\}$, the private key extraction, and the decryption oracles in *Phase* 2, *Phase* 3, and *Phase* 4. Hence, we get $Pr[\mathcal{B}(P, aP, bP, Q, e(P,Q)^{ab}) = 1] = Pr[\beta' = \beta]$ where $|Pr[\beta' = \beta] - \frac{1}{2}| \geq \varepsilon$, and $Pr[\mathcal{B}(P, aP, bP, Q, \mathcal{Z}) = 1] = Pr[\beta' = \beta] = \frac{1}{2}$ when $\mathcal{Z}$ is randomly chosen from $\mathbb{G}_2$. Therefore, we have

$$\begin{aligned} &|Pr[\mathcal{B}(P, aP, bP, Q, e(P,Q)^{ab}) = 1] \\ &\quad - Pr[\mathcal{B}(P, aP, bP, Q, \mathcal{Z}) = 1]| \geq \left| \left(\frac{1}{2} \pm \varepsilon \right) - \frac{1}{2} \right| = \varepsilon. \end{aligned}$$

Thus, $\varepsilon' \geq \varepsilon$ and $\tau' \approx \tau + (q_{H_1} + q_{H_3} + q_1)\mathcal{O}(\tau_1) + q_2 \mathcal{O}(\tau_1 + \tau_2) + (q_H + q_{H_2} + q_{H_4})\mathcal{O}(1)$, where τ_1 and τ_2 denote the computing time for a multiplication in $\mathbb{G}_1$ and a pairing e, respectively. □

Theorem 2. *The proposed multireceiver IBE scheme is $(\tau, q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}, q_1, \varepsilon)$-IND-sMID-CPA secure under the (τ', ε')-Co-DBDH assumption, where $\varepsilon' \geq \varepsilon$ and $\tau' \approx \tau + (q_{H_1} + q_{H_3} + q_1)\mathcal{O}(\tau_1) + (q_H + q_{H_2} + q_{H_4})\mathcal{O}(1)$.*

Proof. Since the $(\tau, q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}, q_1, \varepsilon)$-IND-sMID-CPA security is a special case of the $(\tau, q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}, q_1, q_2, \varepsilon)$-IND-sMID-CCA security with $q_2 = 0$, **Theorem 2** holds. □

5.2 Receiver Anonymity

Receiver anonymity means that every user only knows whether she/he is one of the exact receivers of a ciphertext, while she/he cannot determine whether any other user is an exact receiver or not.

Theorem 3. *The proposed multireceiver IBE scheme is $(\tau, q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}, q_1, q_2, \varepsilon)$-ANON-sID-CCA secure under the (τ', ε')-DBDH-M assumption, where $\varepsilon' \geq \varepsilon$ and*

$$\begin{aligned} \tau' \approx \tau &+ (q_{H_1} + q_{H_3} + q_1)\mathcal{O}(\tau_1) + q_2 \mathcal{O}(\tau_1 + \tau_2) \\ &+ (q_H + q_{H_2} + q_{H_4})\mathcal{O}(1). \end{aligned}$$

Proof. Assume that an ANON-sID-CCA attacker $\mathcal{A}$ has advantage $\mathbf{Adv}_{\Pi}^{\text{ANON-sID-CCA}}(\mathcal{A}) \geq \varepsilon$ within running time τ

where $\prod$ is the proposed scheme. Suppose that $\mathcal{A}$ makes at most q_1 private key extraction queries, at most q_2 decryption queries, and at most q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4} queries to the hash functions H, H_1, H_2, H_3, H_4, respectively. We can show how an algorithm $\mathcal{B}$ can solve the DBDH-M problem with advantage ε' within running time τ', where $\varepsilon' \geq \varepsilon$ and $\tau' \approx \tau + (q_{H_1} + q_{H_3} + q_1)\mathcal{O}(\tau_1) + q_2\mathcal{O}(\tau_1 + \tau_2) + (q_H + q_{H_2} + q_{H_4})\mathcal{O}(1)$, by utilizing $\mathcal{A}$.

First, $\mathcal{B}$ is given $<q, \mathbb{G}_1, P, aP, bP, \mathcal{U}>$ as an instance of the DBDH-M problem. $\mathcal{B}$ can simulate the Challenger to execute each phase of the ANON-sID-CCA game (**Definition 7**) for $\mathcal{A}$ as follows:

***Phase* 1**: Suppose that $\mathcal{A}$ outputs a target identity pair (ID_1, ID_2).

Setup: $\mathcal{B}$ randomly chooses $\beta \in \{1, 2\}$ and $\omega_1, \omega_2 \in \mathbb{Z}_q^*$. $\mathcal{B}$ sets $P_{pub} = \omega_1 bP$ and $P_1 = \omega_2 P$, and gives the attacker $\mathcal{A}$ $<q, \mathbb{G}_1, \mathbb{G}_2, e, n, P, P_1, P_{pub}, H, H_1, H_2, H_3, H_4>$ as the public parameters of the proposed scheme where H, H_1, H_2, H_3, and H_4 are random oracles controlled by $\mathcal{B}$ as follows:

H_1**-query.** Input an identity ID_j to H_1 where $j \in [1, n]$: If there exists (ID_j, l_j, Q_j) in H_1**List**, return Q_j. Otherwise, do the following:

1. Pick an integer $l_j \in \mathbb{Z}_q^*$ at random.
2. If $ID_j = ID_\beta$, then $Q_j = aP$; else if $ID_j \neq ID_i$ for each $i \in \{1, 2\}$, then compute $Q_j = l_j P - P_1$; else compute $Q_j = l_j P$.
3. Put (ID_j, l_j, Q_j) in H_1**List**.
4. Return Q_j.

The other parts are the same as those in the proof of **Theorem 1**.

***Phase* 2**: Upon receiving a private key extraction query ID_j with $ID_j \neq ID_i$ for each $i \in \{1, 2\}$, $\mathcal{B}$ performs the same operations as those in the proof of **Theorem 1**.

***Phase* 3**: Upon receiving a decryption query (C^*, ID_i) where $i \in \{1, 2\}$ and $C^* = <R_1, U_1, U_2, V, W>$, $\mathcal{B}$ performs the following:

1. Search H_3**List** to get (M_j, ρ_j) when $\Gamma_j = U_1$. If not found, return "reject" to $\mathcal{A}$.
2. Compute $\lambda = R_1$.
3. Compute
$$\sigma' = V \oplus H_2\left(\frac{e(\rho_j bP, \omega_1(P_1 + Q_i))}{e(U_2, \lambda)}\right).$$
4. Test whether $M_j = D_{H_4(\sigma')}(W)$ or not. If not, return "reject" to $\mathcal{A}$; else return M_j to $\mathcal{A}$.

Challenge: $\mathcal{A}$ outputs a target plaintext M. Upon receiving M, $\mathcal{B}$ does the following:

1. Pick a string $\sigma \in \{0, 1\}^w$ at random.
2. Set $U_1 = bP = rP$, $U_2 = \omega_1 P$, and $R_1 = \mathcal{U}$.
3. Compute $\mathcal{K} = e(U_1, P_{pub})^{\omega_2}$.
4. Create a target ciphertext $C = <R_1, U_1, U_2, \sigma \oplus H_2(\mathcal{K}), E_{H_4(\sigma)}(M)>$.
5. Return C to $\mathcal{A}$.

***Phase* 4**: $\mathcal{A}$ issues private key extraction queries as those in *Phase* 2 and decryption queries for target identities as those in *Phase* 3 where $C^* \neq C$ is a restriction here.

Guess: Finally, $\mathcal{A}$ outputs its guess $\beta' \in \{1, 2\}$. If $\beta' = \beta$, then $\mathcal{B}$ outputs 1. Otherwise, $\mathcal{B}$ outputs 0. If $R_1 = ab^2P$, then

$$\begin{aligned}\mathcal{K} &= e(U_1, P_{pub})^{\omega_2} = e(bP, \omega_1 bP)^{\omega_2} = e(bP, \omega_1\omega_2 bP \\ &\quad + \omega_1 abP - \omega_1 abP) \\ &= \frac{e(bP, \omega_1 b(aP + \omega_2 P + aP - aP))}{e(\omega_1 P, ab^2 P)} \\ &= \frac{e(bP, \omega_1 b(Q_\beta + P_1))}{e(\omega_1 P, R_1)} = \frac{e(U_1, d_\beta)}{e(U_2, \lambda)}.\end{aligned}$$

Thus,

$$\sigma \oplus H_2(\mathcal{K}) = \sigma \oplus H_2\left(\frac{e(U_1, d_\beta)}{e(U_2, \lambda)}\right).$$

Hence, C is a valid ciphertext. Otherwise, R_1 is a randomly chosen element of $\mathbb{G}_1$. As the construction above, $\mathcal{B}$ successfully simulates the random oracles $\{H, H_1, H_2, H_3, H_4\}$, the private key extraction, and the decryption oracles in *Phase* 2, *Phase* 3, and *Phase* 4. Hence, we get

$$Pr[\mathcal{B}(P, aP, bP, ab^2P) = 1] = Pr[\beta' = \beta]$$

where

$$\left|Pr[\beta' = \beta] - \frac{1}{2}\right| \geq \varepsilon,$$

and $Pr[\mathcal{B}(P, aP, bP, \mathcal{U}) = 1] = Pr[\beta' = \beta] = \frac{1}{2}$ when $\mathcal{U}$ is randomly chosen from $\mathbb{G}_1$. Therefore, we have

$$\begin{aligned}&|Pr[\mathcal{B}(P, aP, bP, ab^2P) = 1] - Pr[\mathcal{B}(P, aP, bP, \mathcal{U}) = 1]| \\ &\geq \left|\left(\frac{1}{2} \pm \varepsilon\right) - \frac{1}{2}\right| = \varepsilon.\end{aligned}$$

Thus, $\varepsilon' \geq \varepsilon$ and

$$\begin{aligned}\tau' &\approx \tau + (q_{H_1} + q_{H_3} + q_1)\mathcal{O}(\tau_1) + q_2\mathcal{O}(\tau_1 + \tau_2) \\ &\quad + (q_H + q_{H_2} + q_{H_4})\mathcal{O}(1),\end{aligned}$$

where τ_1 and τ_2 denote the computing time for a multiplication in $\mathbb{G}_1$ and a pairing e, respectively. □

Theorem 4. *The proposed multireceiver IBE scheme is* $(\tau, q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}, q_1, \varepsilon)$*-ANON-sID-CPA secure under the* (τ', ε')*-DBDH-M assumption, where* $\varepsilon' \geq \varepsilon$ *and* $\tau' \approx \tau + (q_{H_1} + q_{H_3} + q_1)\mathcal{O}(\tau_1) + (q_H + q_{H_2} + q_{H_4})\mathcal{O}(1)$.

Proof. Since the $(\tau, q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}, q_1, \varepsilon)$-ANON-sID-CPA security is a special case of the $(\tau, q_H, q_{H_1}, q_{H_2}, q_{H_3}, q_{H_4}, q_1, q_2, \varepsilon)$-ANON-sID-CCA security with $q_2 = 0$, **Theorem 4** holds. □

TABLE 1
Performance Comparisons among [1], [7], [8], [13], [14], [19], [22] and Our Scheme

	DWGW [8]	LHL [13]	YCMW [22]	WW [19]	BSS [1]	LH [14]	CS [7]	Ours
Cost of Encryption	$(t+1)GM$ $+(2t-2)GA$ $+1e$ $+1E$	$(\frac{t}{2}+2)GM$ $+(\frac{t}{2}-1)GA$ $+1e$ $+1E$	$3GM+1e$ $+1E$	$2GM$ $+1e$ $+1E$	$(t+2)GM$ $+tGA$ $+1e$	$(a+3)GM$ $+1e$	$(jt+\tau+2)GM$ $+(jn)GA$ $+1e$ $+1E$	$((t^2+t+3)GM$ $+(t^2-t)GA$ $+(t^2+1)MUL$ $+1EX+1e$ $+1E)^*$
Cost of Decryption	tGM $+(t-2)GA$ $+2e$ $+1D$	$(\frac{t}{2}-1)GM$ $+(\frac{t}{2}-2)GA$ $+2e+1D$	$2GM+2e$ $+1D$	$1GM+2e$ $+1D$	$1GM+2e$	$2(a+b)GM$ $+1GA$ $+(a+b+1)e$	$(j\hat{S})GM$ $+(j\hat{S}-j+1)GA$ $+2e+1D$	$(t-1)GM$ $+(t-1)GA$ $+2e$ $+1D$
Cost of Key Generation	$1GM$	$1GM$	$1GM$	$3GM$	$1GM$	$(a+b+1)GM$ $+1GA$	$(jn+n+2)GM$ $+(j+1)GA$	$1GM$ $+1GA$
Number of Public Parameters	8	$2t+4$	5	9	8	9	$8+n+jn$	13
Size of Secret Key	ν	ν	ν	ν+μ	ν	ν	2ν	ν
Size of Ciphertext	$(t+1)\nu$ $+\|M\|$	$(t+1)\nu$ $+\|M\|$	$2\nu+\phi$ $+\|M\|$	2ν $+w+\|M\|$	$(t+1)\nu$ $+\|M\|$	$(a+2)\nu$ $+\|M\|$	$(t+1)\nu$ $+\varpi+\|M\|$	$(t+2)\nu$ $+w+\|M\|$
Hardness Assumption	BDH	BDHE	k-BDHI	BDH	BDH Gap-BDH	BDH Gap-BDH	DBDHE	Co-DBDH DBDH-M
Random Oracle Using	No	Yes	Yes	Yes	Yes	Yes	No	Yes

*By applying the pre-computation shown in Section VI, the cost is $((t+3)GM+1e+1E)$.

- a, b: two positive integers, where $a+b=d$ and d is the degree of a polynomial
- $|ID|$: the bit length of an identity
- j: an integer, $1 \leq j \leq |ID|$
- n: the number of all users
- t: the number of receivers, $1 \leq t \leq n$
- S: the set of receivers
- τ: the serial number of a receiving subgroup, $1 \leq \tau \leq t$
- $\hat{S}$: the number of the subgroups which S is divided into
- GA: addition in $\mathbb{G}_1$ or multiplication in $\mathbb{G}_2$
- GM: multiplication in $\mathbb{G}_1$ or exponentiation computation in $\mathbb{G}_2$
- e: bilinear pairing mapping
- EX: exponentiation computation in $\mathbb{Z}_q^*$
- MUL: multiplication in $\mathbb{Z}_q^*$
- E: symmetric encryption
- D: symmetric decryption
- ν: the bit length of an element in $\mathbb{G}_1$
- μ: the bit length of an element in $\mathbb{Z}_q^*$
- BDHE: Given $\mathbb{G}_1$, $\mathbb{G}_2$, $e(\cdot,\cdot)$, q, and $< P, Q, aP, ..., a^{l-1}P, a^{l+1}P, ..., a^{2l}P >$ for $a \in_R \mathbb{Z}_q$, compute $e(P,Q)^{a^l}$.
- DBDHE: Given $\mathbb{G}_1$, $\mathbb{G}_2$, $e(\cdot,\cdot)$, q, and $< P, Q, aP, ..., a^{l-1}P, a^{l+1}P, ..., a^{2l}P >$ for $a \in_R \mathbb{Z}_q$ and given $Z \in \mathbb{G}_2$, decide Z is a random string or $Z = e(P,Q)^{a^l}$.
- k-BDHI: Given $\mathbb{G}_1^*$, $\mathbb{G}_2^*$, $e(\cdot,\cdot)$, q, and $< g, g^t, g^{t^2}, ..., g^{t^{k-1}} > \in (\mathbb{G}_1^*)^k$, compute $e(g,g)^{1/t}$.
- Gap-BDH: Given $< \mathbb{G}_1, \mathbb{G}_2, e(\cdot,\cdot), q, P, aP, bP, cP >$, compute $e(P,P)^{abc}$ with the help of the BDH oracle, which, given $< P, aP, bP, cP, R >$, outputs 1 if $R = e(P,P)^{abc}$ and 0 otherwise.
- $|M|$: the bit length of a plaintext message
- ϖ: the total bit length of the identities of all receivers
- ϕ: the bit length of an element in $\mathbb{G}_2$
- w: the bit length of a symmetric encryption/decryption key, $128 \leq w \leq 256$

TABLE 2
Properties Comparisons

	DWGW[8]	LHL[13]	YCMW[22]	WW[19]	BSS[1]	LH[14]	CS[7]	Ours
Free Sender$^{(1)}$	○	○	×	×	○	○	○	○
Multicast$^{(2)}$	×	○	×	○	○	○	○	○
Multi-Receiver$^{(3)}$	×	×	×	×	○	○	○	○
Receiver Anonymity$^{(4)}$	×	×	×	×	×	×	×	○
Reuse Key in Joining$^{(5)}$	○	×	○	○	○	×	○	○
Reuse Key in Quitting$^{(6)}$	○	×	×	○	○	×	○	○
Off-line PKG$^{(7)}$	○	×	×	×	○	×	○	○
Single PKG$^{(8)}$	○	○	○	×	○	×	○	○
ID Based	○	×	○	○	○	×	○	○

○: Yes; ×: No

(1): Each of the members can broadcast a message.

(2): A sender can send a message to the groups the sender freely chose.

(3): A sender can send a message to the receivers the sender freely chose.

(4): Each receiver is anonymous to an attacker or any other receiver.

(5): The members do not need to change their private keys when a new member joins the group.

(6): The members do not need to change their private keys when a member quits the group.

(7): The private key generator only generates the private keys. It does not need to participate in the other processes.

(8): There is only one private key generator.

6 Performance Evaluation

As to the computation cost for the sender, some steps in *Encrypt*, shown below, can be precomputed by the sender as long as the receivers have been selected:

1. Pick an integer $\alpha \in \mathbb{Z}_q^*$ at random.
2. For $i = 1, \ldots, t$, compute $x_i = H(ID_i)$ and $Q_i = H_1(ID_i)$.
3. For $i = 1, \ldots, t$, compute
$$f_i(x) = \prod_{1 \leq j \neq i \leq t} \frac{x - x_j}{x_i - x_j} = a_{i,1} + a_{i,2}x + \cdots + a_{i,t}x^{t-1}.$$
4. For $i = 1, \ldots, t$, compute $Y_i = \sum_{j=1}^{t} b_j Q_j$, where $b_j = a_{j,i}\alpha^{-1} \bmod q$.
5. Keep $(\{Y_i, \forall i = 1, \ldots, t\}, \alpha)$.

If the sender wants to encrypt a message M, then

1. Pick a string $\sigma \in \{0,1\}^w$ at random and set $r = H_3(\sigma, M)$.
2. For $i = 1, \ldots, t$, compute $R_i = rY_i$.
3. Set the ciphertext $C = <R_1, \ldots, R_t, rP, \alpha P_{pub}, \sigma \oplus H_2(e(P_{pub}, P_1)^r), E_{H_4(\sigma)}(M)>$.

Therefore, the computation cost for the sender can be reduced to one pairing, one exponentiation computation in $\mathbb{G}_2$, $(t+2)$ multiplications in $\mathbb{G}_1$, and a symmetric encryption. In [7], the PKG computes each user's personal private key by using polynomial interpolation in the *Key Generation* (*Extract*) phase, and in the *Encapsulation* (*Encrypt*) phase, a sender computes the message encryption/decryption key via polynomial interpolation, too. In our scheme, when a sender wants to send a message to a set of receivers, she/he computes the message encryption/decryption key and hides their identities by using polynomial interpolation in the *Encrypt* phase only. Thus, each user in [7] stores more secret parameters than ours, but the computation cost of our scheme is higher than that of [7]. Considering precomputation in both [7] and ours, by the method shown in Section 6, the computation cost of our scheme for encryption is less than that of [7] and each user's secret parameters of our scheme are still fewer than those of [7], too.

The performance comparsions among [1], [7], [8], [13], [14], [19], [22] and our scheme are summarized in Table 1.

7 Comparisons and Remarks

Finally, we compare our scheme with the other encryption schemes proposed in the literature [1], [7], [8], [13], [14], [19], [22]. The comparisons are summarized in Table 2.

The scheme [1] may be able to achieve receiver anonymity if a ciphertext is not accompanied with a label that contains information about how the byte-order of the ciphertext is associated with each selected receiver. By using this way, each selected receiver must decrypt each part of the ciphertext until she/he can decrypt the ciphertext successfully. Thus, every selected receiver has to compute $(t+1)$, in average, bilinear pairings for the decryption, and every nonselected receiver must perform $2t$ bilinear pairing computations to ensure that she/he is not a selected receiver of the ciphertext, where t is the number of the selected receivers of the ciphertext. By applying the above way to [7], the scheme may also be able to satisfy receiver anonymity with the same decryption cost as the above. Note that the security of receiver anonymity in the two solutions derived from [1], [7] still needs to be proved formally. In our proposed scheme, it only requires two bilinear pairing computations for each selected or nonselected receiver to perform the decryption. The decryption cost is much lower than that of the solutions derived from [1], [7]. This is because that our scheme is tailored for receiver anonymity. Furthermore, the security of receiver anonymity has been formally proved in Theorems 3 and 4.

8 Conclusions

In this paper, we have proposed an efficient and secure anonymous multireceiver identity-based encryption scheme.

The scheme makes it impossible for an attacker or any other receiver to derive the identity of a message receiver such that the privacy of every receiver can be guaranteed. Our scheme is much more efficient for receivers than two straightforward solutions. Furthermore, we have also formally demonstrated that the proposed scheme can meet the indistinguishability of encryptions under the selective multi-ID, chosen plaintext attacks (IND-sMID-CPA), the indistinguishability of encryptions under the selective multi-ID, chosen ciphertext attacks (IND-sMID-CCA), the anonymous indistinguishability of encryptions under selective-ID, chosen plaintext attacks (ANON-sID-CPA), and the anonymous indistinguishability of encryptions under selective-ID, chosen ciphertext attacks (ANON-sID-CCA).

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments on their paper. A partial result of this work has been presented at National Information Security Conference, Taiwan, ROC, 2007 and won the Best Student Paper Award. This work was supported in part by the National Science Council of the ROC (Taiwan) under grants NSC98-2219-E-110-001 and NSC96-2221-E-110-071-MY3.

REFERENCES

[1] J. Baek, R. Safavi-Naini, and W. Susilo, "Efficient Multi-Receiver Identity-Based Encryption and Its Application to Broadcast Encryption," *Public Key Cryptography—PKC 2005,* pp. 380-397, Springer, 2005.

[2] M. Bellare and P. Rogaway, "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols," *Proc. ACM CCCS '93,* pp. 62-73, 1993.

[3] J. Bethencourt, H. Chan, A. Perrig, E. Shi, and D. Song, "Anonymous Multi-Attribute Encryption with Range Query and Conditional Decryption," technical report, Carnegie Mellon Univ., CMU-CS-06-135, 2006.

[4] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," *SIAM J. Computing,* vol. 32, no. 3, pp. 586-615, 2003.

[5] X. Boyen and B. Waters, "Anonymous Hierarchical Identity-Based Encryption (without Random Oracles)," *Advances in Cryptology—CRYPTO 2006,* Springer, Cryptology ePrint Archive, Report 2006/085, http://eprint.iacr.org/2006/085.pdf, 2006.

[6] H. Chabanne, D.H. Phan, and D. Pointcheval, "Public Traceability in Traitor Tracing Schemes," *Advances in Cryptology—EUROCRYPT 2005,* pp. 542-558, Springer, 2005.

[7] S. Chatterjee and P. Sarkar, "Multi-Receiver Identity-Based Key Encapsulation with Shortened Ciphertext," *Progress in Cryptology—INDOCRYPT 2006,* pp. 394-408, Springer , 2006.

[8] X. Du, Y. Wang, J. Ge, and Y. Wang, "An ID-Based Broadcast Encryption Scheme for Key Distribution," *IEEE Trans. Broadcasting,* vol. 51, no. 2, pp. 264-266, June 2005.

[9] E. Fujisaki and T. Okamoto, "Secure Integration of Asymmetric and Symmetric Encryption Schemes," *Proc. Advances in Cryptology—CRYPTO '99,* pp. 537-554, 1999.

[10] C. Gentry, "Practical Identity-Based Encryption without Random Oracles," *Advances in Cryptology—EUROCRYPT 2006,* pp. 445-464, Springer, 2006.

[11] F.B. Hildebrand, *Introduction to Numerical Analysis,* second ed. Dover, 1974.

[12] L. Hu, D.G. Feng, and T.H. Wen, "Fast Multiplication on a Family of Koblitz Elliptic Curves," *J. Software,* vol. 14, no. 11, pp. 1907-1910, 2003.

[13] J.W. Lee, Y.H. Hwang, and P.J. Lee, "Efficient Pubic Key Broadcast Encryption Using Identifier of Receivers," *Information Security Practice and Experience,* pp. 153-164, Springer, 2006.

[14] L. Lu and L. Hu, "Pairing-Based Multi-Recipient Public Key Encryption," *Proc. 2006 Int'l Conf. Security Management,* pp. 159-165, 2006.

[15] V.S. Miller, "The Weil Pairing, and Its Efficient Calculation," *J. Cryptology,* vol. 17, pp. 235-261, 2004.

[16] R. Molva and A. Pannetrat, "Network Security in the Multicast Framework," *Advanced Lectures in Networking,* pp. 59-82, Springer, 2002.

[17] T. Okamoto and D. Pointcheval, "REACT: Rapid Enhanced-Security Asymmetric Cryptosystem Transform," *Topics in Cryptology CT-RSA 2001,* pp. 159-174, Springer-Verlag, 2001.

[18] M. Scott, N. Costigan, and W. Abdulwahab, "Implementing Cryptographic Pairings on Smartcards," Cryptology ePrint Archive, Report 2006/144, http://eprint.iacr.org/2006/144.pdf, 2006.

[19] L. Wang and C.-K. Wu, "Efficient Identity-Based Multicast Scheme from Bilinear Pairing," *IEE Proc. Comm.,* vol. 152, no. 6, pp. 877-882, 2005.

[20] V.K. Wei, T.H. Yuen, and F. Zhang, "Group Signature where Group Manager Members Open Authority are Identity-Based," *Information Security and Privacy,* pp. 468-480, Springer, 2005.

[21] E.D. Win, S. Mister, B. Prennel, and M. Wiener, "On the Performance of Signature Based on Elliptic Curves," *Algorithmic Number Theory,* pp. 252-266, Springer, 1998.

[22] C. Yang, X. Cheng, W. Ma, and X. Wang, "A New ID-Based Braodcast Encryption Scheme," *Autonomic and Trusted Computing 2006,* pp. 487-492, Springer-Verlag, 2006.

[23] T.H. Yuen and V.K. Wei, "Fast and Proven Secure Blind Identity-Based Signcryption from Pairings," *Topics in Cryptology CT-RSA 2005,* pp. 305-322, Springer, 2005.

Chun-I Fan received the MS degree in computer science and information engineering from National Chiao Tung University, Taiwan, in 1993, and the PhD degree in electrical engineering at National Taiwan University in 1998. From 1999 to 2003, he was an associate researcher and project leader of Telecommunication Laboratories, Chunghwa Telecom Co., Ltd, Taiwan. In 2003, he joined the faculty of the department of computer science and engineering, National Sun Yat-sen University, Kaohsiung, Taiwan, and is a full professor now. He won the Dragon Thesis Award from Acer Foundation and Best Thesis Award from Institute of Information & Computing Machinery in 1999, Best Student Paper Awards in National Conference on Information Security 1998 and 2007. He also was the editor-in-chief of *Information Security Newsletter*, Chinese Cryptology and Information Security Association. He was a program co-chair of ACM International Workshop on Cross Layer Design 2008, an international advisor of the International Congress on Pervasive Computing and Management 2008, and program committeeman of several international conferences. His current research interests include information security, cryptographic protocols, wireless security, and electronic commerce, and he has published over 80 papers in journals, books, and conference proceedings. He is a member of the IEEE and the IEEE Computer Society.

Ling-Ying Huang received the BS degree in applied mathematics from Fu Jen Catholic University, Taipei, Taiwan, and the MS degree in computer science and engineering at National Sun Yat-sen University, Kaohsiung, Taiwan. Her current research interests include cryptography, identity authentication, and information security.

Pei-Hsiu Ho received the MS degree in information management from Southern Taiwan University of Technology in 2003. She is now working toward the PhD degree in computer science and engineering at National Sun Yat-sen University. From 2003 to 2005, she was a software engineer in Asia Pacific Telecom Co., Ltd, Taiwan. Her current research interests include information security and cryptographic protocols about electronic cash and digital signatures.