JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 34, 1517-1541 (2018)
DOI: 10.6688/JISE.201811_34(6).0009

Provably CCA-Secure Anonymous Multi-Receiver Certificateless Authenticated Encryption

YI-FAN TSENG[1] AND CHUN-I FAN[2,*]
Department of Computer Science and Engineering
National Sun Yat-sen University
Kaohsiung, 80424 Taiwan
E-mail: yftseng1989@gmail.com[1]; cifan@mail.cse.nsysu.edu.tw[2]

Multi-receiver encryption allows a sender to choose a set of authorized receivers and send them a message securely and efficiently. Only one ciphertext corresponding to the message is generated regardless of the number of receivers. Thus it is practical and useful for video conferencing systems, pay-per-view channels, distance education, and so forth. In 2010, for further protecting receivers' privacy, anonymous multi-receiver identity-based (ID-based) encryption was first discussed, and from then on, many works on the topic have been presented so far. To deal with the key escrow problem inherited from ID-based encryption (IBE), Islam *et al*. proposed the first anonymous multi-receiver certificateless encryption (AMRCLE) in 2014. In 2015, Hung *et al*. proposed a novel AMRCLE to improve the efficiency. However, we found that their security proofs are flawed, *i.e*., the simulation cannot be successfully performed. In this paper, we present a novel AMRCLE scheme with CCA security in confidentiality and anonymity against both Type I and Type II adversaries. Moreover, the identity of the sender of a ciphertext can be authenticated by the receiver after a successful decryption. To the best of our knowledge, the proposed scheme is the first CCA secure AMRCLE scheme, and furthermore, we also pioneer in achieving sender authentication in AMRCLE.

Keywords: anonymity, multi-receiver encryption, chosen-ciphertext attacks, certificateless encryption, sender authentication

1. INTRODUCTION

Multi-receiver encryption is a practical and useful methodology for a sender to compute and transmit only one ciphertext corresponding to a message for multiple receivers. By using such encryption schemes, the communication cost is greatly decreased. Thus it is popular among some advanced services such as video conferencing, pay-per-view TV, and distance education. An important issue in such services is the authentication of the sender, that is, the source and legality of the digital products should be guaranteed. Many researchers focused on this topic and have proposed interesting results [1, 8, 26].

In some situations, such as ordering sensitive TV programs, anonymity of receivers might be required. In such environment, a customer may expect that her/his identity is not revealed when communication is proceeded. Motivated from this requirement, Fan *et al*. [9] first introduced the concept of anonymous mutli-receiver ID-based encryption (AMRIBE) in 2010. Also, a multi-receiver ID-based encryption scheme using Lagrange interpolating polynomials is proposed in [9]. From then on many related results have

Received March 23, 2017; revised June 18, 2017; accepted July 21, 2017.
Communicated by Hung-Min Sun.
* The corresponding author.

been proposed [4-7, 11, 13, 16, 18-20, 22-33]. Very recently, Fan *et al*. [10] proposed a novel type of AMRIBE with sender authentication called anonymous multi-receiver identity-based authenticated encryption. Their work is formally proved to be CCA security both in confidentiality and anonymity. Some analyses to the previous works have also been proposed in [10].

However, those schemes are constructed in ID-based cryptosystem [3], which means the key escrow problem exists in these schemes. Since the private keys of users' are generated by a third party, key generation center (KGC), it can decrypt all the ciphertexts and reveal the identity of any receiver. Once the KGC is compromised, it would cause great damage to the system. It seems that the use of AMRIBE might be limited to small and closed groups with a fully trusted third authority. Therefore, in view of the aforementioned reasons, Islam *et al*. [14] proposed an anonymous multi-receiver certificateless encryption (AMRCLE) scheme in 2014. However, we found that their security proofs are flawed. In their security proofs, the simulator cannot successfully create the challenge ciphertext, and thus, it fails in the simulation. There is another AMRCLE has been proposed by Hung *et al*. [12] in 2015. They claimed that their scheme has better efficiency compared with Islam *et al*.'s scheme. Nevertheless, we found the their security proofs is unable to cover all possible attackers since some specific restriction is made to the attackers.

In this manuscript, the first AMRCLE scheme with CCA security is presented. Our work is based on the result proposed in [10]. The scheme of [10] is constructed on ID-Based setting, and thus the key escrow problem exists, where KGC can decrypt all ciphertexts and get the identity of any cipheretxt receiver. Our work adopts the concept of certificateless cryptosystems instead of ID-Based setting, which makes it possible for the proposed scheme to be secure even against a malicious KGC. We provide formal proofs to demonstrate the proposed scheme as being CCA secure against both Type I and Type II attackers in the random oracle model [2]. Furthermore, it achieves sender authentication, which makes it possible for a receiver to confirm the sender of a ciphertext after a successful decryption, such that it also is the first anonymous multi-receiver certificateless authenticated encryption (AMRCLAE) scheme in the literature.

2. PRELIMINARIES

In this section, we define anonymous multi-receiver certificateless encryption and review some hard problems and assumption.

Definition 1: The definition of AMRCLE was given by Islam *et al*. in [14]. In this paper, we define anonymous multi-receiver certificateless authenticated encryption (AMRCLAE). An AMRCLAE scheme consists of the following algorithms:

- **Setup** takes as input a security parameter l. It returns a master secret key *msk* and system parameters params.
- **PartialKeyExtract** takes as input the master secret key msk and a user's identity $ID_i \in \{0, 1\}^*$, then returns the partial private key D_i of the user.
- **SetSecretValue** takes as input a user's identity ID_i and outputs the secret value x_i of the user.

– **SetPrivateKey** takes as input params, a user's identity ID_i and user's partial private key D_i. It returns the private key S_i of the user.
– **SetPublicKey** takes as input params, a user's identity ID_i and the user's partial private key D_i. It returns the public key PK_i of the user.
– **Encrypt** takes as input a message m, a sender's private key S_s, and an identity list (ID_1, ID_2, …, ID_t) and returns a ciphertext C. We write $C = Encrypt(params, S_s, (ID_1, ID_2, …, ID_t), m)$.
– **Decrypt** takes as input a ciphertext C and the private key S_i of a user with identity ID_i and returns a message m. We write $m = Decrypt(params, C, S_i)$.

Let G_1 and G_2 be two cyclic groups of prime order q, P be a generator of G_1, and e: $G_1 \times G_1 \to G_2$ be a bilinear mapping.

Definition 2 (The Bilinear Diffie-Hellman (BDH) Problem)**:** Given (P, aP, bP, cP) for some random $a, b, c \in \mathbb{Z}_q^*$, compute $e(P, P)^{abc}$.

Definition 3 (The Decisional Bilinear Diffie-Hellman (DBDH) Problem)**:** Given (P, aP, bP, cP, Z) for some random $a, b, c \in \mathbb{Z}_q^*$, and $Z \in_R \{e(P, P)^{abc}, Y \in_R G_2 \setminus \{e(P, P)^{abc}\}\}$, decide if $Z = e(P, P)^{abc}$.

Definition 4 (The DBDH Assumption [3])**:** Define that an algorithm $\mathcal{A}$ with output $\beta \in \{0, 1\}$ has advantage ε in solving the *DBDH* problem if

$$|Pr[\mathcal{A}(P, aP, bP, cP, e(P, P)^{abc}) = 1] - Pr[\mathcal{A}(P, aP, bP, cP, Z) = 1]| \geq \varepsilon$$

where $a, b, c \in_R \mathbb{Z}_q^*$ and $Z \in_R \{e(P, P)^{abc}, Y \in_R G_2 \setminus \{e(P, P)^{abc}\}\}$. We say that the *DBDH* assumption holds if no polynomial-time algorithm has non-negligible advantage in solving the *DBDH* problem.

Definition 5 (The Modified Decisional Bilinear Diffie-Hellman (*M-DBDH*) Problem [10])**:** Given (P, aP, bP, cP, $e(P, P)^{b^2c}$, Z) for some random $a, b, c \in \mathbb{Z}_q^*$, and $Z \in_R \{e(P, P)^{abc}, Y \in_R G_2 \setminus \{e(P, P)^{abc}\}\}$, decide if $Z = e(P, P)^{abc}$. Define that an algorithm $\mathcal{A}$ with output $\beta \in \{0, 1\}$ has advantage ε in solving the *M-DBDH* problem if

$$\begin{aligned}&|Pr[\mathcal{A}(P, aP, bP, cP, e(P, P)^{b^2c}, e(P, P)^{abc}) = 1]\\ &- Pr[\mathcal{A}(P, aP, bP, cP, e(P, P)^{b^2c}, Z) = 1]| \geq \varepsilon\end{aligned}$$

where $a, b, c \in_R \mathbb{Z}_q^*$ and $Z \in_R \{e(P, P)^{abc}, Y \in_R G_2 \setminus \{e(P, P)^{abc}\}\}$.

Definition 6 (The *M-DBDH* Assumption [10])**:** We say that the *M-DBDH* assumption holds if no polynomial-time algorithm has non-negligible advantage in solving the *M-DBDH* problem.

3. RELATED WORKS

In 2014, Islam *et al*. proposed the first AMRCLE scheme in order to deal with the key escrow problem [14]. They also claimed that their scheme is provably secure under the CDH assumption. In 2015, Hung *et al*. proposed a new AMRCLE scheme with security proofs and better efficiency compared to [14]. In this section, we briefly review Islam *et al*.'s scheme and Hung *et al*.'s scheme.

3.1 Islam *et al*.'s Scheme

3.1.1 Scheme description

- **Setup**

PKG performs the following operations:

1. Generate a group G with prime order p and generator P.
2. Choose an integer $s\in\mathbb{Z}_p^*$ randomly as the master secret key, and set $P_0 = sP$.
3. Choose four cryptographic one-way hash functions, H_0, H_1, H_2, H_3: $\{0, 1\}^* \rightarrow \mathbb{Z}_p^*$.
4. Select secure symmetric encryption/decryption function $E_x(\cdot)/D_x(\cdot)$, where x is the symmetric key.
5. Publish the system parameters $params = \{G, p, P, P_0, E_x(\cdot), D_x(\cdot), H_0, H_1, H_2, H_3\}$ and keep the master key s secret.

- **SetSecretValue**

A user i with identity ID_i chooses $x_i\in\mathbb{Z}_p^*$ as his secret key and $P_i = x_iP$ as the corresponding public key.

- **PartialKeyExtract**

User i sends the tuple (ID_i, P_i) to PKG. Then the PKG executes the following.
1. Choose a number $t_i\in\mathbb{Z}_p^*$ and calculate $T_i = t_iP$.
2. Calculate $l_i = H_0(ID_i, T_i, P_i)$ and $d_i = (t_i + sl_i) \bmod p$.
3. Send (d_i, T_i) to user ID_i through secure channel.

- **SetPrivateKey**

ID_i keeps $sk_i = (d_i, x_i)$ as his full private key.

- **SetPublicKey**

ID_i keeps $pk_i = (P_i, T_i)$ as his full public key.

- **Encrypt**

A sender produces the ciphertext of a message by performing the following steps:
1. Choose a message m and select a set of t receivers, whose identities are $\{ID_1, \ldots, ID_t\}$.
2. Choose $r\in\mathbb{Z}_p^*$ and compute $\xi = rP$.
3. For $i = 1$ to t, compute $l_i = H_0(ID_i, T_i, P_i)$, $u_i = r(T_i + l_iP_0 + P_i)$, and $\alpha_i = H_1(\xi, u_i)$.
4. Select a $\theta\in\mathbb{Z}_p^*$ and compute $\psi(x) = \prod_{i=1}^{t}(x-\alpha_i)+\theta = \sum_{i=0}^{t-1}c_ix^i + x^t \bmod p$.
5. Compute $\beta = H_2(\xi, \theta)$, $\delta = E_\beta(m)$ and $\gamma = H_3(c_0, c_1, \ldots, c_{t-1}, m, \theta, \xi, \delta)$.
6. Set the ciphertext $(c_0, c_1, \ldots, c_{t-1}, \xi, \delta, \gamma)$.

- **Decrypt**

After receiving the ciphertext $(c_0, c_1, ..., c_{t-1}, \xi, \delta, \gamma)$, a selected receiver, say ID_i, can decrypt the ciphertext as follows.

1. Compute $u_i = (d_i + x_i)\xi$ and $\alpha_i = H_1(\xi, u_i)$.
2. Compute $\theta = \sum_{j=0}^{t-1} c_j(\alpha_i)^j + (\alpha_i)^t$.
3. Compute $\beta = H_2(\xi, \theta)$, $m = D_\beta(\delta)$, and $\gamma' = H_3(c_0, c_1, ..., c_{t-1}, m, \theta, \xi, \delta)$.
4. Accept m if $\gamma = \gamma'$

3.1.2 Discussion on the simulation of the CCA game for confidentiality against Type II adversary

In the *Setup* phase, the simulator $\mathcal{S}$ sets $P_0 = aP$. When the adversary $\mathcal{A}$ queries for ID_i's partial private key, $\mathcal{S}$ first chooses $d_i, l_i \in \mathbb{Z}_p^*$, and computes $T_i = d_iP - l_iP_0$, and sets $H_0(ID_i, T_i, P_i) = l_i$, where P_i is the public key of ID_i. And then $\mathcal{S}$ sets d_i as the partial private key of ID_i. In the *Challenge* phase, $\mathcal{S}$ needs to compute $u_j = b(T_j + P_j)$ for the target user ID_j, where $P_j = x_jP$. However, $b(T_j + P_j) = b(d_jP - l_jP_0 + x_jP) = (d_j + x_j)(bP) - l_j(abP)$. Since computing abP is hard, $\mathcal{S}$ cannot successfully generate the challenge ciphertext and fails the simulation. Same flaws also happen in the other proofs. Even if we use the DDH assumption as the underlying assumption, one can break the assumption through a bilinear mapping function.

3.2 Hung *et al.*'s Scheme

3.2.1 Scheme description

- **Setup**

PKG performs the following operations:

1. Generate two cyclic group G_1, G_2 with prime order q, bilinear mapping function e: $G_1 \times G_1 \to G_2$, and a generator P of G_1.
2. Choose an integer $s \in \mathbb{Z}_q^*$ randomly as the master secret key, and set $P_{pub} = sP$.
3. Choose six cryptographic one-way hash functions, H_0: $\{0, 1\}^* \to G_1$, H_1: $G_2 \times G_1 \to \{0, 1\}^w$, H_2, H_3, H_4: $\{0, 1\}^w$, and H_5: $\{0, 1\}^* \times G_1 \to Z_q^*$ for some integer w.
4. Select secure symmetric encryption/decryption function $E_{sk}(\cdot)/D_{sk}(\cdot)$, where sk is the symmetric key.
5. Publish the system parameters $params = \{G_1, G_2, e, q, P, P_{pub}, E_{sk}(\cdot), D_{sk}(\cdot), H_0, H_1, H_2, H_3, H_4, H_5\}$ and keep the master key s secret.

- **PartialKeyExtract**

A user sends his identity ID to PKG. Then the PKG computes the partial private key $DID = s \cdot QID = sH_0(ID)$.

- **SetSecretValue**

The user with identity ID chooses $xid \in \mathbb{Z}_q^*$ as his secret value.

- **SetPublicKey**

The user with identity ID compute his public key as $PID = xid \cdot P$.

- **SetPrivateKey**
  The user with identity ID keeps $SID = (DID, xid)$ as his private key.

- **Multiencrypt**
  A sender produces the ciphertext of a message by performing the following steps:
  1. Choose a message m and select a set of t receivers with public key (ID_1, PID_1), …, (ID_t, PID_t).
  2. Choose $r \in \mathbb{Z}_q^*$ and compute $U = rP$ and $F_i = rPID_i$ for $i = 1, ..., t$.
  3. For $i = 1$ to t, compute $QID_i = H_0(ID_i)$, $K_i = e(P_{pub}, QID_i)r$, and $T_i = H_1(K_i, F_i)$.
  4. Pick an ephemeral value $\sigma \in \{0, 1\}^w$ at random and compute $C_i = H_2(T_i)||(H_3(T_i)\oplus\sigma)$ for $i = 1, ..., t$.
  5. Compute $V = E_{H_4(\sigma)}(m)$ and $\Lambda = H_5(m, \sigma, C_1, C_2, ..., C_t, V, U)$.
  6. Set the ciphertext $CT = (C_0, C_1, ..., C_t, V, U, \Lambda)$.

- **Decrypt**
  After receiving the ciphertext $CT = (C_0, C_1, ..., C_t, V, U, \Lambda)$, a selected receiver, say ID, with full private key (DID, xid), can decrypt the ciphertext as follows.
  1. Compute $K = e(U, DID)$, $F = xid \cdot U$, $T = H_1(K, F)$, and $H_2(T)$.
  2. Use $H_2(T)$ to find its associated C_i by the relation $C_i = H_2(T)||W$, where W denotes the remaining strings by removing $H_2(T)$ from C_i.
  3. Compute $\sigma' = W\oplus H_3(T)$ and $m' = D_{H_4(\sigma')}(V)$.
  4. Accept m if $\Lambda = H_5(m', \sigma', C_1, ..., C_t, V, U)$.

3.2.2 Discussion on the simulation of the CCA game for confidentiality against Type I adversary

In the proof of Theorem 1, the confidentiality against Type I adversary, the authors made an assumption that if the adversary $\mathcal{A}$ wins the game, then must have queried to the oracle H_1 with some specific inputs (K, F), such that $BDDH(P, QID_i, P_{pub}, U^*, K) = 1$. With the specific inputs from $\mathcal{A}$, the challenger is able to solve the Gap-BDH problem by computing $e(P, P)^{abc} = K^{u_i^{-1}}$. Since U^* is set to be cP in the challenge phase, if $\mathcal{A}$ is able to compute K, it implies $\mathcal{A}$ is able to computes the symmetry key sk used to encrypt m_β, and thus $\mathcal{A}$ wins the game. However, the proof only aims at the attackers who are capable of getting the key, sk, before winning the CCA game. The authors have not considered the attackers who can win the game without getting the key. As a result, their proof does not cover all possible attackers. Besides, the same problem exists in the proof for Theorems 2, 3 and 4, too.

4. AN ANONYMOUS MULTI-RECEIVER CERTIFICATELESS AUTHENTICATED ENCRYPTION SCHEME

In this section, we propose an AMRCLAE scheme with provable CCA security in both confidentiality and anonymity against Type I and Type II attackers. The notations are shown in Table 1.

The proposed AMRCLAE scheme is described as follows.

Table 1. The notations.

Notation	Meaning
G_1	a cyclic additive group of prime order q
G_2	a cyclic multiplicative group of prime order q e
e	a bilinear mapping; e: $G_1 \times G_1 \rightarrow G_2$
P	a generator of G_1
KGC	the key generation center
P_{pub}	the public key of KGC
M	a message
ID_i	the identity of user i
Q_i	the hashed value of ID_i
d_i	the partial private key of ID_i
P_i	the public key of ID_i
S_i	the private key of ID_i

- **Setup**

KGC performs the following operations:

1. Choose an integer $\alpha \in \mathbb{Z}_q^*$ randomly as the master secret key, and set $P_{pub} = \alpha P$.
2. Choose three cryptographic one-way hash functions, H: $\{0, 1\}^* \rightarrow G_1$, H_1: $G_2 \times G_2 \rightarrow \mathbb{Z}_q^*$, and H_2: $G_2 \times \mathbb{Z}_q^* \rightarrow \mathbb{Z}_q^*$.
3. Compute $\Omega = e(P, P)$.
4. Publish the system parameters $params = \{G_1, G_2, e, q, P, P_{pub}, H, H_1, H_2, \Omega\}$ and keep the master key α secret.

- **PartialKeyExtract**

When user i joins the system, KGC will compute $Q_i = H(ID_i)$ and the partial private key $d_i = \alpha Q_i$ of the user, and then KGC will send d_i to user i in a secure manner.

- **SetSecretValue**

A user with identity ID_i randomly chooses $x_i \in \mathbb{Z}_q^*$ as his secret value.

- **SetPrivateKey**

A user with identity ID_i set his private key $S_i = (d_i, x_i)$.

- **SetPublicKey**

A user with identity ID_i computes $P_i = x_i P$ as his public key.

- **Encrypt**

A sender, whose identity is ID_s, produces the ciphertext of a message by performing the following steps:

1. Choose a message $M \in G_2$ and select a set of t receivers, whose identities are $\{ID_1, \ldots, ID_t\}$.
2. Choose $k \in \mathbb{Z}_q^*$ at random and compute $r = H_2(M, k)$.
3. For $i = 1$ to t, compute $Q_i = H(ID_i)$ and $v_i = H_1(e(rQ_i, d_s), e(rQ_i, x_s P_i))$, where P_i is the public key of the receiver with identity ID_i, and (d_s, x_s) is the private key of the sender.
4. Compute $f(x) = \prod_{i=1}^{t}(x - v_i) + k = \sum_{i=0}^{t-1} c_i x^i + x^t \bmod q$.
5. Compute $U = rP$, $U_1 = rP_s$, $V = rQ_s$, and $W = M \cdot \Omega^k$, where P_s is the public key of the sender.
6. Set the ciphertext $C = (c_0, c_1, \ldots, c_{t-1}, U, U_1, V, W, ID_s)$.

• **Decrypt**

After receiving the ciphertext $C = (c_0, c_1, ..., c_{t\text{-}1}, U, U_1, V, W, ID_s)$, a selected receiver, say ID_i, can decrypt C as follows,

1. Compute $v_i' = H_1(e(V, d_i), e(x_iQ_i, U_1))$.
2. Compute $k' = f(v_i') = \sum_{j=0}^{t-1} c_j (v_i')^j + (v_i')^t \bmod q$.
3. Compute $M' = W/\Omega^{k'}$.
4. Accept M' if $U = H_2(M', k')P$. Otherwise, output $\perp$. To authenticate the sender, the decryptor can verify if $e(U, H(ID_s)) = e(V, P)$ and $e(U, P_s) = e(U_1, P)$.

The correctness of encryption and decryption is demonstrated as follows.

$$\begin{aligned} v_i' &= H_1(e(V, d_i), e(x_iQ_i, U_1)) \\ &= H_1(e(rQ_s, \alpha Q_i), e(x_iQ_i, rx_sP)) \\ &= H_1(e(rQ_i, \alpha Q_s), e(rQ_i, x_sx_iP)) \\ &= H_1(e(rQ_i, d_s), e(rQ_i, x_sP_i)) \\ &= v_i, \end{aligned}$$

and

$$k' = f(v_i') = f(v_i) = k.$$

Thus, the selected receiver ID_i can successfully recover the message by computing $M' = W/\Omega^{k'} = W/\Omega^k = M$, and he will accept the message due to $U = H_2(M, k)P = H_2(M', k')P$. After successfully recovering the message, it follows that $e(V, d_i) = e(rH(ID'), \alpha H(ID_i)) = e(rH(ID_i), d')$ for some identity ID'. The receiver can be convinced that the ciphertext is encrypted with the private key of some valid user ID', where $V = rH(ID')$. If $e(U, H(ID_s)) = e(V, P)$, $V = rH(ID_s)$, which implies $ID' = ID_s$. Similarly, $e(x_iQ_i, U_1) = e(rQ_i, x'P_i)$ for some user's secret value x' such that $U_1 = r(x'P)$. Once $e(U, P_s) = e(U_1, P)$, $U_1 = rP_s = r(x_sP)$, which implies $x' = x_s$. Thus, the sender is authenticated.

5. SECURITY MODELS AND PROOFS

In this section, we will define the security models and the security notions for AMRCLAE with sender authentication. The security notions are "Indistinguishability of encryptions under selective multi-ID, chosen-ciphertext attacks" (IND-sMID-CCA) and "Anonymous indistinguishability of encryptions under selective multi-ID, chosen-ciphertext attacks" (Anon-sMID-CCA). There are two types of adversary in the definition of our security model. A Type I adversary is able to replace the public key of any user, but not able to access the master secret key *msk*. A Type II adversary has the ability to access the master secret key *msk* but is not allowed to replace the public keys of users. The Type II adversary models security against a malicious KGC and the Type I adversary models security against a malicious user. We will prove that our proposed scheme is CCA secure in confidentiality and anonymity against Types I and II adversaries.

5.1 Confidentiality

Definition 7 (The IND-sMID-CCA Game I)**:** Let $\mathcal{A}$ be a polynomial-time Type I attacker. $\mathcal{A}$ interacts with a simulator $\mathcal{S}$ in the following game.

Initialization: $\mathcal{A}$ chooses a set of identities $ID^* = \{ID^*_1, ID^*_2, ..., ID^*_t\}$ and sends ID^* to $\mathcal{S}$.

Setup: $\mathcal{S}$ runs the Setup algorithm to generate *params* and *msk*. $\mathcal{S}$ then sends *params* to $\mathcal{A}$.

Phase 1: $\mathcal{A}$ issues the following queries.

– Hash query: $\mathcal{S}$ operates hash functions on the inputs given by $\mathcal{A}$ and returns the hashed values.
– PartialKeyExtract(ID_i): $\mathcal{A}$ sends an identity ID_i to $\mathcal{S}$ and $\mathcal{S}$ returns the partial private key of ID_i where *PartialKeyExtract*(ID_j) cannot be queried if $ID_j \in ID^*$.
– *SecretValue*(ID_i): $\mathcal{A}$ sends an identity ID_i to $\mathcal{S}$ and $\mathcal{S}$ returns the secret value x_i of ID_i.
– *PublicKey*(ID_i): $\mathcal{A}$ sends an identity ID_i to $\mathcal{S}$ and $\mathcal{S}$ returns the public key P_i of ID_i.
– *KeyReplacement*(ID_i, P'_i, x'): $\mathcal{A}$ sends an identity ID_i, the public key, and the secret value to $\mathcal{S}$. $\mathcal{S}$ then replaces the public key and the secret value of ID_i with P'_i *and* x'_i, respectively.
– *Encrypt*(ID_s, ID_1, ..., ID_u, M): $\mathcal{A}$ sends a sender's identity ID_s, a receiver set $\{ID_1, ..., ID_u\}$, and a message M to $\mathcal{S}$. $\mathcal{S}$ returns a ciphertext C to $\mathcal{A}$.
– *Decrypt*(C, ID_i): $\mathcal{A}$ sends an identity ID_i and a ciphertext C to $\mathcal{S}$ and $\mathcal{S}$ returns the result of the decryption.

Challenge: $\mathcal{A}$ submits a sender's identity ID_s and (M_0, M_1) to $\mathcal{S}$ where M_0, M_1 are two distinct messages of the same length and $ID_s \notin ID^*$. $\mathcal{S}$ then randomly chooses $\beta \in \{0, 1\}$ and generates $C^* = Encrypt(ID_s, ID^*_1, ..., ID^*_t, M_\beta)$. Finally, $\mathcal{S}$ sends C^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ issues the queries defined in Phase 1 except for issuing Decrypt queries with $C = C^*$ and $ID_i \in ID^*$.

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

The advantage of $\mathcal{A}$ winning the game is defined as

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

An AMRCLAE scheme is said to be IND-sMID-CCA secure against Type I adversary if there exists no polynomial-time Type I attacker that can win the IND-sMID-CCA game I with non-negligible advantage.

Definition 8 (The IND-sMID-CCA Game II)**:** Let $\mathcal{A}$ be a polynomial-time Type II attacker. $\mathcal{A}$ interacts with a simulator $\mathcal{S}$ in the following game.

Initialization: $\mathcal{A}$ chooses a set of identities $ID^* = (ID^*_1, ID^*_2, ..., ID^*_t)$ and sends ID^* to $\mathcal{S}$.

Setup: $\mathcal{S}$ runs the Setup algorithm to generate *params* and *msk*. $\mathcal{S}$ then sends *params* and *msk* to $\mathcal{A}$.

Phase 1: $\mathcal{A}$ issues the following queries.

− *Hash query*: $\mathcal{S}$ operates hash functions on the inputs given by $\mathcal{A}$ and returns the hashed values.

− *SecretValue*(ID_i): $\mathcal{A}$ sends an identity ID_i to $\mathcal{S}$ and $\mathcal{S}$ returns the secret value x_i of ID_i where *SecretValue*(ID_j) cannot be queried if $ID_j \in ID^*$.

− *PublicKey*(ID_i): $\mathcal{A}$ sends an identity ID_i to $\mathcal{S}$ and $\mathcal{S}$ returns the public key P_i of ID_i.

− *KeyReplacement*(ID_i, P'_i, x'_i): $\mathcal{A}$ sends an identity ID_i, where $ID_i \notin ID^*$, the public key, and the secret value to $\mathcal{S}$. $\mathcal{S}$ then replaces the public key and the secret value of ID_i with P' and x', respectively.

− *Encrypt*(ID_s, ID_1, ..., ID_u, M): $\mathcal{A}$ sends a sender's identity ID_s, a receiver set $\{ID_1, \ldots, ID_u\}$, and a message M to $\mathcal{S}$. $\mathcal{S}$ returns a ciphertext C to $\mathcal{A}$.

− *Decrypt*(C, ID_i): $\mathcal{A}$ sends an identity ID_i and a ciphertext C to $\mathcal{S}$ and $\mathcal{S}$ returns the result of the decryption.

Challenge: $\mathcal{A}$ submits a sender's identity ID_s and (M_0, M_1) to $\mathcal{S}$ where M_0, M_1 are two distinct messages of the same length and $ID_s \notin ID^*$. $\mathcal{S}$ then randomly chooses $\beta \in \{0, 1\}$ and generates $C^* = Encrypt(ID_s, ID_1^*, ..., ID_t^*, M_\beta)$. Finally, $\mathcal{S}$ sends C^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ issues the queries defined in Phase 1 except for issuing Decrypt queries with $C = C^*$ and $ID_i \in ID^*$.

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

The advantage of $\mathcal{A}$ winning the game is defined as

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

It is not necessary to simulate *PartialKeyExtract* oracle of $\mathcal{A}$ since $\mathcal{A}$ has *msk*. An AMRCLAE scheme is said to be IND-sMID-CCA secure against Type II adversary if there exists no polynomial-time Type II attacker that can win the IND-sMID-CCA game II with non-negligible advantage.

Theorem 1: (Confidentiality) The proposed AMRCLAE scheme is IND-sMID-CCA secure against Type I adversary in the random oracle model if the M-DBDH assumption holds.

Proof: The main idea of the proof is proof by contradiction. Assume that the proposed scheme is not IND-sMID-CCA secure against Type I adversary, *i.e.*, there exists a polynomial-time Type I adversary $\mathcal{A}$ that wins the IND-sMID-CCA game I with non-negligible advantage. Then we will construct a polynomial-time algorithm $\mathcal{S}$ that has non-negligible advantage in solving the M-DBDH problem.

First, $\mathcal{S}$ is given $\langle q, G_1, G_2, e, P, aP, bP, cP, e(P, P)^{b^2c}, Z\rangle$ which is an instance of the M-DBDH problem. $\mathcal{S}$ simulates the game for $\mathcal{A}$ as follows:

Initialization: $\mathcal{A}$ outputs a target identity set $ID^* = \{ID_1^*, ..., ID_t^*\}$.

Setup: $\mathcal{S}$ sets $P_{pub} = cP$, computes $\Omega = e(P, P)$, and outputs $\{G_1, G_2, e, q, P, P_{pub}, H, H_1, H_2, \Omega\}$ as the public parameters where H, H_1, and H_2 are three random oracles controlled by $\mathcal{S}$.

Phase 1: $\mathcal{S}$ maintains H-list, H_1-list, and H_2-list to store the results of querying H, H_1, and H_2, respectively. In this phase $\mathcal{A}$ can issue the following queries:

– H-query:
This oracle takes an identity $ID_j \in \{0, 1\}^*$ as input. If there exists a record (ID_j, Q_j, q_j) in H-list, return Q_j. Otherwise, do the following:
1. Randomly select $q_j \in \mathbb{Z}_q^*$.
2. If $ID_j \in ID^*$, compute $Q_j = q_j(bP)$; else $Q_j = q_jP$.
3. Return Q_j and add (ID_j, Q_j, q_j) into H-list.

– H_1-query:
This oracle takes (X_j, Y_j) as input, where $X_j, Y_j \in G_2$. If there exists a record (X_j, Y_j, v_j) in H_1-list, return v_j. Otherwise, do the following:
1. Randomly choose $v_j \in \mathbb{Z}_q^*$.
2. Add (X_j, Y_j, v_j) to H_1-list.
3. Return v_j.

– H_2-query:
This oracle takes $M_j \in G_2$ and an integer $k_j \in \mathbb{Z}_q^*$ as input. If there exists a record (M_j, k_j, r_j, U_j) in H_2-list, return r_j. Otherwise, do the following:
1. Randomly choose $r_j \in \mathbb{Z}_q^*$ and compute $U_j = r_jP$.
2. Add (M_j, k_j, r_j, U_j) to H_2-list.
3. Return r_j.

– PartialKeyExtract:
This oracle takes an identity ID_j as input. Call $H(ID_j)$ and retrieve q_j form H-list.
Then, $\mathcal{S}$ does the following:
– If $ID_j \in ID^*$, return "reject".
– Otherwise, compute $d_j = q_j\,(cP)$ and return d_j.

– PublicKey:
This oracle takes an identity ID_j as input. If there exists (ID_j, P_j, x_j) in pk-list, return P_j.
Otherwise, do the following:
1. Choose $x_j \in \mathbb{Z}_q^*$.
2. Compute $P_j = x_jP$.
3. Add (ID_j, P_j, x_j) to pk-list.
4. Return P_j.

– SecretValue:
This oracle takes an identity ID_j as input. If there exists (ID_j, P_j, x_j) in pk-list, return x_j. Otherwise, do the following:
1. Choose $x_j \in \mathbb{Z}_q^*$.
2. Compute $P_j = x_jP$.
3. Add (ID_j, P_j, x_j) to pk-list
4. Return x_j.

– KeyReplacement:
$\mathcal{A}$ may issue this query with input (ID_j, P'_j, x'_i). $\mathcal{S}$ then replaces the record ID_j, P_j, x_j in pk-list with (ID_j, P'_j, x'_i).

– Encrypt:
This oracle takes $u + 1$ identities $(ID_s, ID_1, ..., ID_u)$ and a message M as input. Upon receiving an Encrypt query, $\mathcal{S}$ does the following:
1. Choose $k, r \in \mathbb{Z}_q^*$ at random and set $H_2 (M, k) = r$.
2. For $i = 1$ to u,
   – if $ID_s \notin ID^*$, compute $v_i = H_1(e(Q_i, d_s)^r, e(Q_i, P_i)^{rx_s})$, where (d_s, x_s) is the private key of the sender ID_s;
   – if $ID_s \in ID^*$ and $ID_i \notin ID^*$, compute $v_i = H_1(e(d_i, Q_s)^r, e(Q_i, P_s)^{rx_i})$, where (d_i, x_i) is the private key of the receiver ID_i and P_s is the public key of ID_s;
   – if $ID_s, ID_i \notin ID^*$, set $T = e(P, P)^{b^2c}$, compute $v_i = H_1(T^{rq_sq_i}, e(Q_i, P_i)^{rx_s})$.
3. Compute $f(x) = \prod_{i=1}^{u}(x - v_i) + k = \sum_{i=0}^{u-1} c_i x^i + x^u \mod q$.
4. Compute $U = rP$, $U_1 = rP_s$, $V = rQ_s$, and $W = M \cdot \Omega^k$.
5. Set the ciphertext $C = (c_0, c_1, ..., c_{u\text{-}1}, U, U_1, V, W, ID_s)$ and return C.

– Decrypt:
This oracle takes an identity ID_j and a ciphertext C as input. Upon receiving a Decrypt query, denoted by Decrypt(C, ID_j) where $C = (c_0, ..., c_{u\text{-}1}, U, U_1, V, W, ID_s)$, $\mathcal{S}$ does the following:
1. Search H_2-list to get (M_i, k_i, r_i, U_i) with $U_i = U$. If not found, return "reject".
2. Search H-list to get (ID_s, Q_s, q_s) with $e(U, Q_s) = e(P, V)$. If not found, return "reject".
3. This step can be separated into three cases:
   – if $ID_s \notin ID^*$, compute $v_j = H_1(e(Q_i, d_s)^{r_i}, e(Q_i, P_i)^{r_ix_s})$;
   – if $ID_s \in ID^*$ and $ID_j \notin ID^*$, compute $v_j = H_1(e(d_j, Q_s)^{r_i}, e(Q_j, P_s)^{r_ix_j})$;
   – if $ID_s, ID_j \notin ID^*$, set $T = e(P, P)^{b^2c}$ and compute $v_j = H_1(T^{r_iq_sq_j}, e(Q_j, P_j)^{rx_s})$.
4. Compute $k = c_0 + c_1v_j + \ldots + c_{u\text{-}1}v_j^{u\text{-}1} + v_j^u \mod q$.
5. Check whether $k_i = k$ and $M_i = W/\Omega^k$ or not. If not, return "reject". Otherwise, return M_i.

Challenge: $\mathcal{A}$ sends (M_0, M_1) and ID_s to $\mathcal{S}$ where $M_0, M_1 \in G_2$ are two distinct messages with the same length and $ID_s \notin ID^*$. $\mathcal{S}$ performs the following operations:
1. Choose $\beta \in \{0, 1\}$ randomly.
2. For $i = 1$ to t, call $H(ID_i^*)$ and retrieve q_i^* from H-list.

3. Call $H(ID_s)$ and retrieve q_s from H-list.
4. Search (P_i^*, x_i^*) for $i = 1, ..., t$ and (P_s, x_s) in the pk-list.
5. Choose $k \in \mathbb{Z}_q^*$ and set $U^* = aP$, $U_1^* = x_s(aP)$, and $V^* = q_s(aP)$.
6. For $i = 1$ to t, compute $v_i = H_1(Z^{q_i^* q_s}, e(q_i^*(bP), x_s x_i^*(aP)))$.
7. Compute $f(x) = \prod_{i=1}^{t}(x - v_i) + k = \sum_{i=0}^{t-1} c_i x^i + x^t \bmod q$ and $W^* = M_\beta \cdot \Omega^k$.
8. Set the ciphertext $C^* = (c_0, c_1, ..., c_{t-1}, U^*, U_1^*, V^*, W^*, ID_s)$ and send C^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ makes queries as those in Phase 1. However, if $\mathcal{A}$ issues a Decrypt query with input $C = C^*$ and $ID_i \in ID^*$, $\mathcal{S}$ will return "reject."

Guess: Finally, $\mathcal{A}$ outputs $\beta' \in \{0, 1\}$. If $\beta' = \beta$, then $\mathcal{S}$ outputs 1. Otherwise, $\mathcal{S}$ randomly chooses $\bar{\beta} \in \{0, 1\}$ and outputs $\bar{\beta}$.

If $Z = e(P, P)^{abc}$, then $Z^{q_i^* q_s} = e(P, P)^{abcq_i^* q_s} = e(q_i^*(bP), q_s(cP))^a = e(Q_i^*, d_s)^a$ for $i = 1$ to t. Therefore, C^* is a correct ciphertext. Otherwise, Z is an element randomly chosen in G_2. As the construction above, $\mathcal{S}$ correctly simulates the IND-sMID-CCA game I. If $\mathcal{A}$ wins the IND-sMID-CCA game with non-negligible advantage at least ε, $|Pr[\beta' = \beta] - ½| \geq \varepsilon$ under a correct simulation of the game, *i.e.*, $|Pr[\mathcal{A}(\Pi) = \beta' = \beta] - ½| \geq \varepsilon$, where Π is a correct AMRCLAE scheme. Thus, we have that

$$\begin{aligned}
&Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, e(P,P)^{abc}) = 1] \\
&= Pr[\mathcal{A}(\Pi) = \beta] + \frac{1}{2}(1 - \Pr[A(\Pi) = \beta]) \\
&= \frac{1}{2} Pr[\mathcal{A}(\Pi) = \beta] + \frac{1}{2}
\end{aligned}$$

and

$$\begin{aligned}
&Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, Z) = 1] \\
&= \frac{1}{2} Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, e(P,P)^{abc}) = 1] \\
&= \frac{1}{2} Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, X \in_R G_2 \setminus \{e(P,P)^{abc}\}) = 1] \\
&= \frac{1}{2}(\frac{1}{2} Pr[\mathcal{A}(\Pi) = \beta] + \frac{1}{2}) + \frac{1}{2}(\frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2}) \\
&= \frac{1}{4} Pr[\mathcal{A}(\Pi) = \beta] + \frac{5}{8}.
\end{aligned}$$

We can obtain

$$\begin{aligned}
&| Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, e(P,P)^{abc}) = 1] \\
&- Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, Z) = 1] | \\
&= | \frac{1}{4} Pr[\mathcal{A}(\Pi) = \beta] - \frac{1}{8} | = \frac{1}{4} | \Pr[A(\Pi) = \beta] - \frac{1}{2} | \geq \frac{\varepsilon}{4}.
\end{aligned}$$

Therefore, $\mathcal{S}$ solves the M-DBDH problem with non-negligible advantage $\frac{\varepsilon}{4}$ within polynomial time. ❑

Theorem 1 ensures the CCA security of confidentiality against Type I adversary. In confidentiality, there do not exist inside attackers (selected receivers) because every selected receiver can decrypt the ciphertext.

Theorem 2: (Confidentiality) The proposed AMRCLAE scheme is IND-sMID-CCA secure against Type II adversary in the random oracle model if the M-DBDH assumption holds.

***Proof*:** First, $\mathcal{S}$ is given $\langle q, G_1, G_2, e, P, aP, bP, cP, e(P, P)^{b^2c}, Z\rangle$. $\mathcal{S}$ simulates the game for an adversary $\mathcal{A}$ as follows.

Initialization: $\mathcal{A}$ outputs a target identity set $ID^* = \{ID_1^*, ..., ID_t^*\}$.

Setup: $\mathcal{S}$ sets $P_{pub} = \alpha P$, where $\alpha \in \mathbb{Z}_q^*$, and sends α to $\mathcal{A}$. The rest of the step is the same as that of the proof of Theorem 1.

Phase 1: $\mathcal{S}$ maintains H-list, H_1-list, and H_2-list and $\mathcal{A}$ can issue the following queries:

– H-query: This oracle is identical to that of the proof of Theorem 1 if we replace "$q_j(bP)$" by "$q_j(cP)$".
– H_1-query: This oracle is identical to that of the proof of Theorem 1.
– H_2-query: This oracle is identical to that of the proof of Theorem 1.
– PublicKey: This oracle is identical to that of the proof of Theorem 1 except step 2) shown below.

2) If $ID_j \in ID^*$, compute $P_j = x_j(bP)$; else compute $P_j = x_jP$.

– SecretValue:
This oracle takes an identity ID_j as input. If $ID_j \in ID^*$, $\mathcal{S}$ returns "reject". If (ID_j, P_j, x_j) is in pk-list, return x_j. Otherwise, $\mathcal{S}$ simulates this oracle as that of the proof of Theorem 1.
– KeyReplacement:
$\mathcal{A}$ may issue this query with input (ID_j, P_j', x_i'). If $ID_j \in ID^*$, $\mathcal{S}$ return "reject". $\mathcal{S}$ then replaces the record ID_j, P_j, x_j in pk-list with (ID_j, P_j', x_i').
– Encrypt:
This oracle is the same as that of the proof of Theorem 1 except the following.
– if $ID_s, ID_i \in ID^*$, set $T = e(P, P)^{b^2c}$ and compute $v_i = H_1(e(Q_i, d_s)^r, T^{rx_sx_iq_i})$.
– Decrypt: This oracle is the same as that of the proof of Theorem 1 except the following.
– if $ID_s, ID_i \in ID^*$, set $T = e(P, P)^{b^2c}$ and compute $v_j = H_1(e(Q_j, d_s)^{r_i}, T^{r_ix_sx_jq_j})$.

Challenge: This step is the same as that of the proof of Theorem 1 except the following.

6) For $i = 1$ to t, compute $v_i = H_1(e(Q_i^*, \alpha q_s(aP)), Z^{q_i^*x_i^*x_s})$.

Phase 2 and Guess: The two steps are identical to those of the proof of Theorem 1.

If $Z = e(P, P)^{abc}$, then C^* is a correct ciphertext and $\mathcal{S}$ correctly simulates the IND-sMID-CCA game II. If $\mathcal{A}$ wins the IND-sMID-CCA game II with non-negligible advantage ε, $|Pr[\mathcal{A}(\Omega) = \beta' = \beta] - ½| \geq \varepsilon$, where Ω is a correct AMRCLAE scheme. Thus, we have that

$$\begin{aligned}
&| Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, e(P,P)^{abc}) = 1] \\
&-Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, Z) = 1] | \\
&=| \frac{1}{4} Pr[\mathcal{A}(\Omega) = \beta] - \frac{1}{8} |= \frac{1}{4} | Pr[\mathcal{A}(\Omega) = \beta] - \frac{1}{2} | \geq \frac{\varepsilon}{4}.
\end{aligned}$$

Thus, $\mathcal{S}$ solves the M-DBDH problem with non-negligible advantage $\frac{\varepsilon}{4}$ within polynomial time. ❑

Theorem 2 ensures the CCA security of confidentiality against Type II adversary.

5.2 Anonymity

Definition 9 (The Anon-sMID-CCA Game I)**:** Let $\mathcal{A}$ be a polynomial-time Type I attacker. $\mathcal{A}$ interacts with a simulator $\mathcal{S}$ in the following game.

Initialization: $\mathcal{A}$ chooses two identities $ID^* = \{ID^*_0, ID^*_1\}$ and sends ID^* to $\mathcal{S}$.

Setup and Phase 1: These two phases are the same as those of Definition 7.

Challenge: $\mathcal{A}$ submits a sender's identity ID_s, a message M, and a set of identities $\{ID_2, ID_3, ..., ID_t\}$ to $\mathcal{S}$, with restrictions that $ID_s \notin ID^*$ and *PartialKeyExtract*(ID_s) has not been queried before. $\mathcal{S}$ then randomly chooses $\beta \in \{0, 1\}$ and generates $C^* = Encrypt(ID_s, ID^*_\beta, ID_2, ..., ID_t, M)$. Finally, $\mathcal{S}$ sends C^* *to* $\mathcal{A}$.

Phase 2: $\mathcal{A}$ issues the queries defined in Phase 1 except for issuing *Decrypt*(C^*, ID^*_0), *Decrypt*(C^*, ID^*_1), and *PartialKeyExtract*(ID_s).

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

The advantage of $\mathcal{A}$ winning the game is defined as

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

An AMRCLAE scheme is said to be Anon-sMID-CCA secure against Type I adversary if there exists no polynomial-time Type I attacker that can win the Anon-sMID-CCA game I with non-negligible advantage.

Definition 10 (The Anon-sMID-CCA Game II)**:** Let $\mathcal{A}$ be a polynomial-time Type II attacker. $\mathcal{A}$ interacts with a simulator $\mathcal{S}$ in the following game.

Initialization: $\mathcal{A}$ chooses two identities $ID^* = \{ID_0^*, ID_1^*\}$ and sends ID^* to $\mathcal{S}$.

Setup and Phase 1: These two phases are the same as those of Definition 8.

Challenge: $\mathcal{A}$ submits a sender's identity ID_s, a message M, and a set of identities $\{ID_2, ID_3, ..., ID_t\}$ to $\mathcal{S}$, with restrictions that $ID_s \notin ID^*$ and $SecretValue(ID_s)$ has not been queried before. $\mathcal{S}$ then randomly chooses $\beta \in \{0, 1\}$ and generates $C^* = Encrypt(ID_s, ID_\beta^*, ID_2, ..., ID_t, M)$. Finally, $\mathcal{S}$ sends C^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ issues the queries defined in Phase 1 except for issuing $Decrypt(C^*, ID_0^*)$, $Decrypt(C^*, ID_1^*)$, and $SecretValue(ID_s)$.

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

The advantage of $\mathcal{A}$ winning the game is defined as

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

It is not necessary to simulate *PartialKeyExtract* oracle of $\mathcal{A}$ since $\mathcal{A}$ has *msk*. An AMRCLAE scheme is said to be Anon-sMID-CCA secure against Type II adversary if there exists no polynomial-time Type II attacker that can win the Anon-sMID-CCA game II with non-negligible advantage.

Note that there are some restrictions about the choice of ID_s in the challenge phase. In the Anon-sMID-CCA Game I and the Anon-sMID-CCA Game II, we have not modelled the sender as an attacker. The anonymity will be meaningless if the sender is the attacker since the receivers are chosen by him.

Theorem 3: (Anonymity) The proposed AMRCLAE scheme is Anon-sMID-CCA secure against Type I adversary in the random oracle model if the M-DBDH assumption holds.

***Proof*:** $\mathcal{S}$ is given $\langle q, G_1, G_2, e, P, aP, bP, cP, e(P, P)^{b^2c}, Z\rangle$ and then simulates the game for $\mathcal{A}$ as follows:

Initialization: $\mathcal{A}$ outputs a target identity set $ID^* = \{ID_0^*, ID_1^*\}$.

Setup and Phase 1: These two phases are the same as those in the proof of Theorem 1.

Challenge: $\mathcal{A}$ sends a message M, $t - 1$ receivers' identities $\{ID_2, ID_3, ..., ID_t\}$, and a sender's identity ID_s to $\mathcal{S}$ with restrictions that $ID_s \notin ID^*$ and PartialKeyExtract(ID_s) has not been queried. $\mathcal{S}$ does the following:

1. Choose $\beta \in \{0, 1\}$ randomly.

2. For $i = 2$ to t, call $H(ID_i)$ and retrieve q_i from H-list.
3. Call $H(ID^*_\beta)$ and retrieve q^*_β from H-list.
4. Call $H(ID_s)$ and retrieve q_s from H-list.
5. Get P_i and x_i for $i = 2, ..., t$ from the pk-list.
6. Get P^*_β, x^*_β, P_s, and x_s from the pk-list.
7. Choose $k \in \mathbb{Z}^*_q$, and set $U^* = aP$, $U^* = x_s(aP)$, and $V^* = q_s(aP)$.
8. For $i = 2$ to t, compute $v_i = H_1(e(q_iU^*, q_s(cP)), e(q_iU^*, x_sx_iP))$.
9. Compute $v_\beta = H_1(Z^{q^*_\beta q_s}, e(q^*_\beta(bP), x_sx^*_\beta(aP)))$.
10. Compute $f(x) = (x - v_\beta)\prod_{i=2}^{t}(x - v_i) + k = \sum_{i=0}^{t-1} c_ix^i + x^t \bmod q$ and $W^* = M \cdot \Omega^k$.
11. Set the ciphertext $C^* = (c_0, c_1, ..., c_{t-1}, U^*, U^*_1, V^*, W^*, ID_s)$ and send C^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ makes queries as those in Phase 1. However, if $\mathcal{A}$ issues Decrypt(C^*, $ID_i \in ID^*$) or PartialKeyExtract(ID_s), $\mathcal{S}$ will return "reject".

Guess: $\mathcal{A}$ outputs $\beta' \in \{0, 1\}$. If $\beta' = \beta$, then $\mathcal{S}$ outputs 1. Otherwise, $\mathcal{S}$ outputs a random bit $\bar{\beta}$.

If $Z = e(P, P)^{abc}$ then $Z^{q^*_\beta q_s} = e(P, P)^{abcq^*_\beta q_s} = e(q^*_\beta(bP), q_s(cP))^a = e(Q^*_\beta, d_s)^a$ for $\beta \in \{0, 1\}$ and C^* is a correct ciphertext. If $\mathcal{A}$ wins the game with non-negligible advantage ε, $|Pr[\mathcal{A}(\Omega) = \beta' = \beta] - ½| \geq \varepsilon$, where $\mathbf{\Omega}$ is a correct AMRCLAE scheme. Thus, we have that

$$\begin{aligned}
&| Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, e(P,P)^{abc}) = 1 \\
&-Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, Z) = 1]| \\
&=|(\frac{1}{2}Pr[\mathcal{A}(\Omega) = \beta] + \frac{1}{2}) - (\frac{1}{4}Pr[\mathcal{A}(\Omega) = \beta] + \frac{5}{8})| \\
&=|\frac{1}{4}Pr[\mathcal{A}(\Omega) = \beta] + \frac{1}{8}| = \frac{1}{4}|Pr[\mathcal{A}(\Omega) = \beta] - \frac{1}{2}| \geq \frac{\varepsilon}{4}.
\end{aligned}$$

Therefore, $\mathcal{S}$ solves the M-DBDH problem with non-negligible advantage $\frac{\varepsilon}{4}$ within polynomial time. ❑

Theorem 3 guarantees the CCA security of anonymity against Type I adversary.

Theorem 4: (Anonymity) The proposed AMRCLAE scheme is Anon-sMID-CCA secure against Type II adversary in the random oracle model if the M-DBDH assumption holds.

***Proof*:** $\mathcal{S}$ is given $\langle q, G_1, G_2, e, P, aP, bP, cP, e(P, P)^{b^2c}, Z\rangle$ and then simulates the game for an adversary $\mathcal{A}$ as follows

Initialization: $\mathcal{A}$ outputs a target identity set $ID^* = \{ID^*_0, ID^*_1\}$.

Setup and **Phase 1:** These two phases are the same as those in the proof of Theorem 2.

Challenge: $\mathcal{A}$ sends a message M, $t-1$ receivers' identities $\{ID_2, ID_3, ..., ID_t\}$, and a sender's identity ID_s to $\mathcal{S}$, with restrictions that $ID_s \notin ID^*$ and SecretValue(ID_s) has not been queried before. $\mathcal{S}$ does the same works as those of the proof of Theorem 3 except the following.

8) For $i = 2$ to t, compute $v_i = H_1(e(q_iU^*, \alpha q_sP), e(q_iU^*, x_sx_iP))$.
9) Compute $v_\beta = H_1(e(q_\beta^*(cP), \alpha q_s(aP), Z^{q_\beta^* x_\beta^* x_s})$.

Phase 2: $\mathcal{A}$ makes queries as those in Phase 1. However, if $\mathcal{A}$ issues Decrypt(C^*, $ID_i \in ID^*$) or SecretValue(ID_s), $\mathcal{S}$ will return "reject".

Guess: $\mathcal{A}$ outputs $\beta' \in \{0, 1\}$. If $\beta' = \beta$, then $\mathcal{S}$ outputs 1. Otherwise, $\mathcal{S}$ outputs a random bit $\bar{\beta}$.

If $Z = e(P, P)^{abc}$, $\mathcal{S}$ correct simulates the Anon-sMID-CCA game II. If $\mathcal{A}$ wins the game with non-negligible advantage ε, $|\Pr[\mathcal{A}(\Omega) = \beta' = \beta] - ½| \geq \varepsilon$, where Ω is a correct AMRCLAE scheme. Thus, we have that

$$| \Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, e(P,P)^{abc}) = 1$$
$$- \Pr[\mathcal{S}(P, aP, bP, cP, e(P,P)^{b^2c}, Z) = 1] | \geq \frac{\varepsilon}{4}.$$

Therefore, $\mathcal{S}$ solves the M-DBDH problem with non-negligible advantage $\frac{\varepsilon}{4}$ within polynomial time. ❑

Theorem 4 guarantees the CCA security of anonymity against Type II adversary.

5.3 Sender Authentication

Definition 11 (The Sender Authentication Game I)**:** Let $\mathcal{A}$ be a polynomial-time Type I attacker. $\mathcal{A}$ interacts with a simulator $\mathcal{S}$ in the following game.

Initialization: $\mathcal{A}$ chooses a target sender identity ID_s^* and target receiver identity ID_R^*. Then $\mathcal{A}$ sends $ID^* = \{ID_s^*, ID_R^*\}$ to $\mathcal{S}$.

Setup and **Phase 1:** These two phases are the same as those of Definition 7.

Forge: Finally, $\mathcal{A}$ outputs a ciphertext C^* with restrictions that the sender is ID_s^*, ID_R^* is one of the receivers, and C^* was not outputted by querying the Encrypt oracle. $\mathcal{A}$ wins the game if C^* is a correct ciphertext.

The advantage of $\mathcal{A}$ winning the game is defined as

$$\mathbf{Adv}^{\text{SA-I}}(\mathcal{A}) = Pr[Decrypt(C, ID_R^*) \neq \perp].$$

An AMRCLAE scheme is said to satisfy sender authentication against Type I adversary if there exists no polynomial-time Type I attacker that can win the Sender Authentication game I with non-negligible advantage.

Definition 12 (The Sender Authentication Game II)**:** Let $\mathcal{A}$ be a polynomial-time Type II attacker. $\mathcal{A}$ interacts with a simulator $\mathcal{S}$ in the following game.

Initialization: $\mathcal{A}$ chooses a target sender identity ID_s^* and target receiver identity ID_R^*. Then $\mathcal{A}$ sends $ID^* = \{ID_s^*, ID_R^*\}$ to $\mathcal{S}$.

Setup and **Phase 1:** These two phases are the same as those of Definition 8.

Forge: Finally, $\mathcal{A}$ outputs a ciphertext C^* with restrictions that the sender is ID_s^*, ID_R^* is one of the receivers, and C^* was not outputted by querying the Encrypt oracle. $\mathcal{A}$ wins the game if C^* is a correct ciphertext.

The advantage of $\mathcal{A}$ winning the game is defined as

$$\mathbf{Adv}^{\text{SA-II}}(\mathcal{A}) = Pr[Decrypt(C, ID_R^*) \neq \perp].$$

An AMRCLAE scheme is said to satisfy sender authentication against Type II adversary if there exists no polynomial-time Type II attacker that can win the Sender Authentication game II with non-negligible advantage.

Theorem 5: (Sender Authentication) The proposed AMRCLAE scheme satisfies sender authentication against Type I adversary in the random oracle model if the 1-wDBDHI assumption holds.

***Proof*:** Assume that there exists a polynomial-time Type I adversary $\mathcal{A}$ that wins the Sender Authentication game I with non-negligible advantage. Then we will construct a polynomial-time algorithm $\mathcal{S}$ that has non-negligible advantage in solving the 1-wDBDHI problem.

First, $\mathcal{S}$ is given $\langle q, G_1, G_2, e, P, bP, cP, Z\rangle$ which is an instance of the 1-wDBDHI problem. $\mathcal{S}$ simulates the game for A as follows:

Initialization: $\mathcal{A}$ outputs an identity set $ID^* = \{ID_s^*, ID_R^*\}$.

Setup and **Phase 1:** If we set $T = Z$, then these two phases will be the same as those in the proof of Theorem 1.

Forge: Finally, $\mathcal{A}$ outputs $C^* = (c_0, c_1, ..., c_{t\text{-}1}, U^*, U_1^*, V^*, W^*, ID_s^*)$, where C^* was not outputted by querying the Encrypt oracle. Then $\mathcal{S}$ performs the followings.

1. Search H_2-list to get (M_i, k_i, r_i, U_i) with $U_i = U^*$.
2. Computer $v^* = H_1(Z^{q_s^* q_R^* r_i}, e(H(ID_R^*), x_R^* U_1^*))$.
3. Computer $k^* = c_0 + c_1 v^* + \ldots + c_{t\text{-}1}(v^*)^{t\text{-}1} + (v^*)^t \bmod q$.

4. Verify if $r_iH(ID_s^*) = V^*$, $k_i = k^*$, $M_i = W/\Omega^{k^*}$, and $e(U^*, H(ID_s^*)) = e(V^*, P)$. If not, $\mathcal{S}$ outputs 0. Otherwise, $\mathcal{S}$ outputs 1.

In the *Encrypt* oracle, if $T = Z = e(P, P)^{b^2c}$, then $T^{rq_sq_i} = e(P, P)^{b^2crq_sq_i} = e(q_i(bP), q_s(bcP))^r = e(Q_i, d_s)^r$. Similarly, in the *Decrypt* oracle, if $T = Z = e(P, P)^{b^2c}$, then $T^{r_iq_sq_j} = e(P, P)^{b^2cr_iq_sq_j} = e(q_j(bP), q_s(bcP))^{r_i} = e(Q_j, d_s)^{r_i}$. As the construction above, $\mathcal{S}$ correctly simulates the game if $Z = e(P, P)^{b^2c}$. Assume that $\mathcal{A}$ wins the game with non-negligible advantage at least ε under a correct simulation. To analysis the advantage of solving the 1-wDBDHI problem, we define the following events.

E_1: The game has been correctly simulated.
E_2: $\mathcal{A}$ wins the game.

Then we have that

$$Pr[\mathcal{S}(P, bP, cP, e(P,P)^{b^2c}) = 1] = Pr[E_1 \wedge E_2]$$
$$Pr[E_1]Pr[E_2 \mid E_1] \geq 1 \cdot \varepsilon = \varepsilon.$$

and

$$|\Pr[\mathcal{S}(P, bP, cP, e(P,P)^{b^2c}) = 1 - \Pr[\mathcal{S}(P, bP, cP, Z) = 1]]|$$
$$= |\tfrac{1}{2}\Pr[\mathcal{S}(P, bP, cP, e(P,P)^{b^2c}) = 1| \geq \tfrac{\varepsilon}{2}.$$

Therefore, $\mathcal{S}$ solves the 1-WDBDHI problem with non-negligible advantage $\frac{\varepsilon}{2}$ within polynomial time. ❑

Theorem 6: (Sender Authentication) The proposed AMRCLAE scheme satisfies sender authentication against Type II adversary in the random oracle model if the 1-wDBDHI assumption holds.

Proof: $\mathcal{S}$ is given $\langle q, G_1, G_2, e, P, bP, cP, Z\rangle$ and then simulates the game for an adversary $\mathcal{A}$ as follows:

Initialization: $\mathcal{A}$ outputs an identity set $ID^* = \{ID_s^*, ID_R^*\}$.

Setup and **Phase 1:** If we set $T = Z$, these two phases will be the same as those in the proof of Theorem 2.

Forge: Finally, $\mathcal{A}$ outputs $C^* = (c_0, c_1, ..., c_{t-1}, U_1^*, U^*, V^*, W^*, ID_s^*)$, where C^* was not outputted by querying the Encrypt oracle. $\mathcal{S}$ performs the followings.

1. Search H_2-list to get (M_i, k_i, r_i, U_i) with $U_i = U^*$.
2. Compute $v^* = H_1(e(V^*, \alpha H(ID_R^*)), Z^{r_i q_R^* x_s^* x_R^*})$.
3. Compute $k^* = c_0 + c_1v^* + \ldots + c_{t-1}(v^*)^{t-1} + (v^*)^t \bmod q$.
4. Verify if $r_iP_s^* = U_1^*$, $k_i = k^*$, $M_i = W/\Omega^{k^*}$, and $e(U^*, P_s^*) = e(U_1^*, P)$. If not, $\mathcal{S}$ outputs 0. Otherwise, $\mathcal{S}$ outputs 1.

$\mathcal{S}$ correctly simulates the game if $Z = e(P, P)^{b^2c}$. Assume that $\mathcal{A}$ wins the game with non-negligible advantage at least ε under a correct simulation. Then we have that

$$| Pr[\mathcal{S}(P, bP, cP, e(P,P)^{b^2c}) = 1$$
$$-Pr[\mathcal{S}(P, bP, cP, Z) = 1] || \geq \tfrac{\varepsilon}{2}.$$

That is, $\mathcal{S}$ solves the 1-wDBDHI problem with non-negligible advantage $\frac{\varepsilon}{2}$ within polynomial time. ❑

Theorems 5 and 6 guarantees that the proposed scheme satisfies Sender Authentication. In other words, even if an adversary compromises with any $t - 1$ of t receivers, the adversary cannot impersonate the sender to generate a correct ciphertext for the t receivers.

6. COMPARISONS

In this section, we compare the proposed scheme with [10] and the existing AM-RCLEs [12, 14] in performance and security. According to [15, 17, 21, 34], we can obtain that $T_p \approx 5T_e$, $T_s \approx 29T_m$, $T_e \approx 240T_m$, $T_h \approx 23T_m$, and $T_a \approx 0.12T_m$ shown in Table 2, which summarizes the computation costs of encryption/decryption and the ciphertext length for multiple receivers.

Table 2. Performance comparison.

	Encryption cost	Decryption cost	Ciphertext Length
[14]	$(2t+2)T_h+(2t+2)T_s+2tT_a+T_{poly}$ $\approx(104t+104)T_m+T_{poly}$	$3T_h+tT_m+T_s$ $\approx(t+98)T_m$	$(t + 1)\|q\| + u + w$
[12]	$(4t+2)T_h+(2t+1)T_s+tT_p$ $\approx(1350t+75)T_m$	$5T_h+T_p+T_s$ $\approx(1344)T_m$	$\|q\| + u + (2t + 1)\, w$
[10]	$(2t+1)T_h+4T_s+tT_p+T_{poly}$ $\approx(1246t+139)T_m+T_{poly}$	$2T_h+tT_m+T_p+T_s$ $\approx(t+1275)T_m$	$t\|q\| + 2u + v + \|ID\|$
Ours	$(t+1)T_h+(3t+4)T_s+T_a+2tT_p+T_{poly}$ $\approx(2510t+139)T_m+T_{poly}$	$2T_h+2T_s+tT_m+2T_p$ $\approx(t+2504)T_m$	$t\|q\| + 3u + v + \|ID\|$

* T_p: the cost of a pairing operation
* T_h: the cost of a hash operation
* T_m: the cost of a modular multiplication in Z_q
* T_e: the cost of a modular exponentiation in Z_q
* T_s: the cost of a scalar multiplication in an additive group or an exponentiation in a multiplicative group
* T_a: the cost of an addition in an additive group or a multiplication in a multiplicative group
* T_{poly}: the cost of constructing polynomial
* T_{CRT}: the cost of using Chinese Remainder Theorem
* t: the number of receivers
* $|ID|$: the bit-length of an identity
* q: a big prime
* u: the bit-length of an element in an additive group
* v: the bit-length of an element in a multiplicative group
* w: the bit-length of a symmetric encryption key

The security comparison is shown in Table 3. There are two kinds of Type I adversaries in the security model of anonymity, who are insiders and outsiders. An insider is a selected receiver, while an outsider is not. An insider can get information more than an outsider since an insider can decrypt the ciphertext. The proposed scheme owns CCA security against both Type I and Type II attackers simultaneously in not only confidentiality but also anonymity. Especially, it also is the first AMRCLE scheme that achieves sender authentication. All the properties of our scheme have been formally proved in Section 5.

Table 3. Security comparison.

	Confidentiality		Anonymity			Security Model	Sender Authentication
	Outsider	KGC	Outsider	Insider	KGC		
[14]	CCA*	CCA*	CCA*	CCA*	CCA*	ROM	No
[12]	CCA*	CCA*	CCA*	CCA*	CCA*	ROM	No
[10]	CCA	–	CCA	CCA	–	ROM	Yes
Ours	CCA	CCA	CCA	CCA	CCA	ROM	Yes

* There exist some problems in their security proofs, shown in Sections 3.1 and 3.2.

7. CONCLUSION

This paper has presented the first anonymous multi-receiver certificateless authenticated encryption scheme. The proposed scheme achieves provable CCA security in both confidentiality and anonymity against Type I and Type II attackers, and it also achieves sender authentication. The security of our scheme is guaranteed based on the M-DBDH assumption and the 1-wDBDHI assumption under the random oracle model. An open problem in this field is to find a secure anonymous multi-receiver certificateless authenticated encryption scheme under the standard model, *i.e.*, without random oracles.

ACKNOWLEDGMENT

This work was partially supported by the Ministry of Science and Technology of Taiwan under grants MOST 105-2923-E-110-001-MY3 and MOST 107-2218-E-110-014, and financially supported by the Information Security Research Center at National Sun Yat-sen University in Taiwan and the Intelligent Electronic Commerce Research Center from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan.

REFERENCES

1. J. Baek, R. Safavi-Naini, and W. Susilo, "Efficient multi-receiver identity-based encryption and its application to broadcast encryption," in *Proceedings of the 8th International Conference on Theory and Practice in Public Key Cryptography*, 2005, pp. 380-397.

2. M. Bellare and P. Rogaway, "Random oracles are practical: a paradigm for designing efficient protocols," in *Proceedings of the 1st ACM Conference on Computer and Communications Security*, 1993, pp. 62-73.
3. D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Proceedings of Advances in Cryptology*, 2001, pp. 213-229.
4. Z. Chen, S. Li, C. Wang, and Y. Shen, "Two constructions of multireceiver encryption supporting constant keys, short ciphertexts, and identity privacy," *International Journal of Network Security*, Vol. 14, 2012, pp. 270-279.
5. Z. Chen, S. Li, C. Wang, and M. Zhang, "Comments on FHH anonymous multireceiver encryption," *International Journal of Network Security*, Vol. 16, 2014, pp. 285-288.
6. H. Y. Chien, "Improved anonymous multi-receiver identity-based encryption," *The Computer Journal*, Vol. 55, 2012, pp. 439-446.
7. H. Cui, Y. Mu, and F. Guo, "Server-aided identity-based anonymous broadcast encryption," *International Journal of Security and Networks*, Vol. 8, 2013, pp. 29-39.
8. X. Du, Y. Wang, J. Ge, and Y. Wang, "An ID-based broadcast encryption scheme for key distribution," *IEEE Transactions on Broadcasting*, Vol. 51, 2005, pp. 264-266.
9. C. I. Fan, L. Y. Huang, and P. H. Ho, "Anonymous multireciever identity-based encryption," *IEEE Transactions on Computers*, Vol. 59, 2010, pp. 1239-1249.
10. C. I. Fan and Y. F. Tseng, "Anonymous multi-receiver identity-based authenticated encryption with CCA security," *Symmetry*, Vol. 7, 2015, pp. 1856-1881.
11. L. Harn, C. C. Chang, and H. L. Wu, "An anonymous multi-receiver encryption based on RSA," *International Journal of Network Security*, Vol. 15, 2013, pp. 307-312.
12. Y. H. Hung, S. S. Huang, Y. M. Tseng, and T. T. Tsai, "Efficient anonymous multi-receiver certificateless encryption," *IEEE Systems Journal*, Vol. 99, 2015, pp. 1-12.
13. J. Hur, C. Park, and S. O. Hwang, "Privacy-preserving identity-based broadcast encryption," *Information Fusion*, Vol. 13, 2012, pp. 296-303.
14. S. K. Islam, M. K. Khan, and A. M. Al-Khouri, "Anonymous and provably secure certificateless multireceiver encryption without bilinear pairing," *Security and Communication Networks*, Vol. 8, pp. 2214-2231.
15. N. Koblitz, A. Menezes, and S. Vanstone, "The state of elliptic curve cryptography," *Designs*, *Codes and Cryptography*, Vol. 19, 2000, pp. 173-193.
16. H. Li and L. Pang, "Cryptanalysis of Wang *et al.*'s improved anonymous multi-receiver identity-based encryption scheme," *IET Information Security*, Vol. 8, 2013, pp. 8-11.
17. A. J. Menezes, S. A. Vanstone, and P. C. V. Oorschot, *Handbook of Applied Cryptography*, CRC Press, Inc. Boca Raton, 2001.
18. A. Muthulakshmi, R. Anitha, S. Rohini, and K. Princy, "Identity based privacy preserving dynamic broadcast encryption for multi-privileged group," *Recent Trends in Computer Networks and Distributed Systems Security*, Vol. 335, 2012, pp. 272-282.
19. L. Pang, L. Guo, Q. Pei, J. Gui, and Y. Wang, "A new ID-based multi-recipient public-key encryption scheme," *Chinese Journal of Electronics*, Vol. 22, 2013, pp. 89-92.

20. Y. Ren, Z. Niu, and X. Zhang, "Fully anonymous identity-based broadcast encryption without random oracles," *International Journal of Network Security*, Vol. 16, 2014, pp. 256-264.
21. M. Scott, "Implementing cryptographic pairings," in *Proceedings of the the First international conference on Pairing-Based Cryptography*, 2007, pp. 177-196.
22. Y. M. Tseng, Y. H. Huang, and H. J. Chang, "Privacy-preserving multireceiver ID-based encryption with provable security," *International Journal of Communication Systems*, Vol. 27, 2012, pp. 1034-1050.
23. Y. M. Tseng, Y. H. Huang, and H. J. Chang, "CCA-secure anonymous multi-receiver ID-based encryption," in *Proceedings of the 26th International Conference on Advanced Information Networking and Applications Workshops*, 2012, pp. 177-182.
24. Y. M. Tseng, T. T. Tsai, S. S. Huang, and H. Y. Chien, "Efficient anonymous multi-receiver ID-based encryption with constant decryption cost," in *Proceedings of International Conference on Information Science*, *Electronics and Electrical Engineering*, 2014, pp. 131-137.
25. H. Wang, "Insecurity of 'Improved anonymous multi-receiver identity-based encryption'," *The Computer Journal*, Vol. 57, 2014, pp. 636-638.
26. L. Wang and C.-K. Wu, "Efficient identity-based multicast scheme from bilinear pairing," *IEE Proceedings of Communications*, Vol. 152, 2005, pp. 877-882.
27. H. Wang, Y. Zhang, H. Xiong, and B. Qing, "Crytanalysis and improvements of an anonymous multi-receiver identity-based encryption scheme," *IET Information Security*, Vol. 6, 2012, pp. 20-27.
28. H. Wang, "Provably Secure anonymous multi-receiver identity-based encryption with shorter ciphertext," in *Proceedings of IEEE 12th International Conference on Dependable, Autonomic and Secure Computing*, 2014, pp. 85-90.
29. B. Zhang, T. Sun, and D. Yu, "ID-based anonymous multi-receiver key encapsulation mechanism with sender authentication," *Algorithm and Architectures for Parallel Processing*, LNCS, Vol. 8631, 2014, pp. 645-658.
30. J. Zhang and J. Mao, "An improved anonymous multi-receiver identity-based encryption scheme," *International Journal of Communication Systems*, Vol. 28, 2015, pp. 645-658.
31. J. Zhang and Y. Xu, "Comment on anonymous multi-receiver identity-based encryption scheme," in *Proceedings of the 4th International Conference on Intelligent Networking and Collaborative Systems*, 2012, pp. 473-476.
32. J. Zhang, Y. Xu, and J. Zou, "Comment on Wang *et al.*'s anonymous multi-receiver ID-based encryption scheme and its improved schemes," *International Journal of Intelligent Information and Database Systems*, Vol. 7, 2013, pp. 400-413.
33. M. Zhang and T. Takagi, "Efficient constructions of anonymous multireceiver encryption protocol and their deployment in group e-mail systems with privacy preservation," *IEEE Systems Journal*, Vol. 7, 2013, pp. 410-419.
34. Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Securing mobile Ad hoc networks with certificateless public keys," *IEEE Transactions on Dependable and Secure Computing*, Vol. 3, 2006, pp. 386-399.

Yi-Fan Tseng (曾一凡) was born in Kaohsiung, Taiwan. He received the B.S. degree, the M.S. degree, and the Ph.D. degree in Computer Science and Engineering from National Sun Yat-sen University, Taiwan, in 2012, 2014, and 2018, respectively. His research interests include cloud computing and security, network and communication security, information security, cryptographic protocols, and applied cryptography.

Chun-I Fan (范俊逸) received the M.S. degree in Computer Science and Information Engineering from the National Chiao Tung University, Hsinchu, Taiwan, in 1993, and the Ph.D. degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan, in 1998. From 1999 to 2003, he was an Associate Researcher and a Project Leader with Telecommunication Laboratories, Chunghwa Telecom Company, Ltd., Taoyuan, Taiwan. In 2003, he joined the faculty of the Department of Computer Science and Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan, where has been a Full Professor since 2010. His current research interests include applied cryptology, cryptographic protocols, and information and communication security. Prof. Fan is the Chairman of the Chinese Cryptology and Information Security Association, and was the Chief Executive Officer (CEO) of "Aim for the Top University Plan" Office at National Sun Yat-sen University. He was the recipient of the Best Student Paper Awards from the National Conference on Information Security in 1998, the Dragon Ph.D. Thesis Award from Acer Foundation, the Best Ph.D. Thesis Award from the Institute of Information and Computing Machinery in 1999, and the Engineering Professors Award from Chinese Institute of Engineers – Kaohsiung Chapter in 2016. Prof. Fan is also an Outstanding Faculty in Academic Research in National Sun Yat-sen University.