IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 May 2023, accepted 21 May 2023, date of publication 25 May 2023, date of current version 1 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3280056

RESEARCH ARTICLE

Leakage-Resilient Anonymous Multi-Receiver Certificate-Based Key Encapsulation Scheme

TUNG-TSO TSAI[1], YUH-MIN TSENG[2], (Member, IEEE), AND SEN-SHAN HUANG[2]
[1]Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung 202, Taiwan
[2]Department of Mathematics, National Changhua University of Education, Changhua City 500, Taiwan

Corresponding author: Yuh-Min Tseng (ymtseng@cc.ncue.edu.tw)

This work was supported in part by the Ministry of Science and Technology, Taiwan, under Contract MOST110-2221-E-018-006-MY2 and Contract MOST110-2221-E-018-007-MY2.

ABSTRACT Key encapsulation schemes in public key system (PKS) can be used to protect sensitive or private data. Unlike traditional PKS and identity-based PKS, certificate-based PKS (CB-PKS) not only avoids the establishment of complex public key infrastructures, but also does not encounter the key escrow problem. Anonymous multi-receiver certificate-based encryption (AMR-CBE) or anonymous multi-receiver certificate-based key encapsulation (AMR-CB-KE) scheme enables a sender to use multiple receivers' public keys to perform one-time encryption process for a message and send the encrypted ciphertext to these receivers, while these receivers do not know the other receiver's identity. However, the existing AMR-CBE and AMR-CB-KE schemes cannot resist side-channel attacks. Attackers with the ability of such attacks can continuously obtain part (several bits) of the secret keys and then calculate the complete secret keys. In such a case, such attacks make a cryptographic scheme (including AMR-CBE and AMR-CB-KE schemes) insecure. Leakage-resilient cryptography is an important research topic to resist side-channel attacks. In this paper, we propose the *first* leakage-resilient anonymous multi-receiver certificate-based key encapsulation (LR-AMR-CB-KE) scheme. Based on the discrete logarithm and hash function assumptions, we demonstrate the scheme has the indistinguishability of two ciphertexts against chosen ciphertext attacks (IND-CCA) and the anonymous indistinguishability of two identities against chosen ciphertext attacks (ANON-IND-CCA) for two types of attackers in CB-PKS settings.

INDEX TERMS Leakage-resilient, side-channel attacks, certificate-based, anonymity, multi-receiver.

I. INTRODUCTION

With the rapid development of Internet, we can easily send data to the remote end through communication networks. However, most of communication networks are not secure channels so that sensitive or private data could be stolen during transmission. One potential solution to the security issues of communication networks is to use blockchain technology [1], [2], [3]. By leveraging the decentralized and immutable nature of blockchain, data can be securely transmitted over a network without the risk of interception or modification by unauthorized parties. The other solution is to encrypt the data before transmission. The remote parties can establish a session key by exchanging authenticated keys [4], [5] and use it to encrypt data during transmission. On the other hand, key encapsulation schemes [6], [7], [8] in public key systems (PKS) can be used to protect sensitive or private data. Indeed, PKS includes many different kinds, such as traditional PKS [9], [10], identity-based PKS [11], [12] and certificate-based PKS [13], [14], [15]. In traditional PKS, complex public key infrastructures (PKI) need to be built to manage each entity's certificate, which is used to connect each entity's identity with its public key. However, it takes a lot of resources to establish the PKI and manage certificates of all entities. To solve this problem, Shamir [16] proposed an identity-based PKS (ID-PKS), in which the public key of an entity is the identity of the entity. Obviously, each entity's public key is a meaningful string in the ID-PKS. This system

The associate editor coordinating the review of this manuscript and approving it for publication was SK. Hafizul Islam.

comprises two roles. One is a private key generator (PKG) who is responsible for generating private keys for entities in the system, and the other is entities who can obtain their own private keys from the PKG. We can observe that this system has the key escrow problem, namely, the PKG knows the private key of each entity and could perform signature or decryption procedures on behalf of the entity.

Certificate-based PKS (CB-PKS) [13] was proposed to avoid both the establishment of complex PKI and the key escrow problem. In this system, the identity of an entity is still viewed its public key, so there is no need to use a certificate to ensure the relationship between its identity and public key. Indeed, the use of the certificate still exists, but the certificate is viewed as one part of the private keys of the entity. The remaining part of the private keys is generated by itself, so the key escrow problem can be eliminated. Based on the CB-PKS, various cryptographic mechanisms have been studied thoroughly, such as certificate-based encryption [17], [18] and certificate-based signature [19], [20].

As mentioned earlier, we use an encryption or key encapsulation mechanism to encrypt the sensitive or private data to ensure the confidentiality. However, typical encryption mechanism uses the public key of single receiver to encrypt a message. There is a situation where a message is encrypted and send to multiple receivers. For example, in a pay-per-view television program system, there may be multiple entities subscribing to the same program. In this way, the administrator of the system must use the multiple Key entities' public keys to encrypt the same program and then send the encrypted program to the multiple entities (subscribers). However, the number of performing encryption procedures is equal to the number of multiple receivers. Obviously, it is inefficient to perform multiple encryption procedures on the same message. To settle this problem, multi-receiver encryption (MRE) schemes [21], [22] were proposed. In which, a sender can use multiple receivers' public keys to perform one-time encryption process for a message and send the same encrypted ciphertext to these receivers. When receiving the ciphertext, each receiver can use its own private key to decrypt and obtain the plaintext.

A. RELATED WORK

Since the MRE schemes under the traditional PKS [21], [22] were presented, these schemes require the establishment of PKI to maintain certificates. To avoid this problem, based on the ID-PKS, the first multi-receiver ID-based encryption (MR-IBE) scheme was proposed by Baek et al. [23]. However, the Baek et al.'s MR-IBE scheme has shortcomings in computational efficiency and ciphertext size. To remove these shortcomings, several MR-IBE schemes [24], [25], [26] were proposed. However, these existing MR-IBE schemes have a common disadvantage, namely, lack of anonymity. The ciphertext generated by these MR-IBE schemes includes the identities of all receivers. When some receiver obtains the ciphertext, he/she also knows the identities of all receivers. However, some cases do not want a receiver's identity information to be known to the other receivers. For providing the anonymity, an anonymous MR-IBE (AMR-IBE) scheme was proposed by Fan et al. [27]. Unfortunately, the proposed AMR-IBE scheme [27] was demonstrated by both Wang et al. [28] and Chien [29] to be unable to meet the anonymity. Meanwhile, an improved AMR-IBE scheme is proposed by Wang et al. [28] and Chien [29], respectively. Although two improved AMR-IBE schemes meet the anonymity, they cannot resist other attacks [30], [31]. Another new improved AMR-IBE scheme was proposed by Tseng et al. [32]. So far, to propose a novel AMR-IBE scheme [33] is still an ongoing research issue. Undoubtedly, all AMR-IBE schemes encounter the key escrow problem. Hence, Fan et al. [34] proposed an anonymous multi-receiver certificate-based encryption (AMR-CBE) scheme to avoid both the establishment of PKI and the key escrow problem.

Indeed, the security of the schemes mentioned above is based on the security of secret keys of both the system and each entity, namely, these secret keys cannot be partially disclosed to the attackers. In other words, if an attacker can continuously obtain part (several bits) of the secret keys by side-channel attacks [35], [36], it can calculate the complete secret keys. Hence, such attacks make a cryptographic scheme insecure. Fortunately, leakage-resilient cryptographic schemes can resist such attacks have been studied deeply. In the PKS, Akavia et al. [37] proposed the first encryption scheme which can resist side-channel attacks, called leakage-resilient encryption (LRE) scheme. To improve the efficiency and security, several improved LRE schemes [38], [39], [40] on the first LRE scheme [37] have been proposed. However, these schemes [37], [38], [39], [40] are secure only in bounded leakage model. Moreover, considering the unbounded leakage model, the first LRE scheme resistant to continual leakage was proposed by Kiltz and Pietrzak [41]. Later, an improved LRE scheme [42] was proposed to reduce ciphertext length and communication cost. In the ID-PKS, the first leakage-resilient ID-based encryption (LR-IBE) scheme resistant to continual leakage was proposed by Brakerski et al. [43]. However, there is a limitation in their scheme, namely, an attacker cannot obtain part of the system secret key. Yuen et al. [44] proposed an improved LR-IBE scheme to exclude the limitation. In the CB-PKS, the first leakage-resilient certificate-based encryption (LR-CBE) scheme was proposed by Yu et al. [14]. Later, a more efficient LR-CBE scheme was proposed by Guo et al. [45]. However, both LR-CBE schemes is secure only in the bounded leakage model. Furthermore, considering the unbounded leakage model, the LR-CBE scheme resistant to continual leakage respectively was proposed by Li et al. [46] and Zhou et al. [47].

B. MOTIVATION AND CONTRIBUTION

Leakage-resilient cryptography is crucial in today's digital age where the confidentiality of sensitive information is

paramount for many applications, such as financial, medical, and government institutions. However, existing cryptographic schemes may suffer from key or other confidential information leakage due to design flaws, implementation issues, or other attack vectors. Without a leakage-resilient mechanism, attackers can exploit leaked information to break the encryption system and gain access to sensitive information. Therefore, to propose a cryptographic scheme resistant to continual leakage is still an important issue at present.

As mentioned earlier, the existing AMR-CBE scheme [34] not only avoids the establishment of PKI, but also avoids the key escrow problem. However, the AMR-CBE scheme cannot resist side-channel attacks [35], [36]. In this paper, we propose the *first* leakage-resilient anonymous multi-receiver certificate-based key encapsulation (LR-AMR-CB-KE) scheme. Our specific contributions are as follows:

- We first formulate a new framework and security models for LR-AMR-CB-KE.
- Based on the new framework, a concrete LR-AMR-CB-KE scheme is proposed.
- Based on the discrete logarithm and hash function assumptions, we demonstrate the scheme has the indistinguishability of two ciphertexts against chosen ciphertext attacks (IND-CCA) and the anonymous indistinguishability of two identities against chosen ciphertext attacks (ANON-IND-CCA) for two types of attackers in CB-PKS settings.

C. ORGANIZATION

The rest of the paper is given as follows. We present some preliminaries in Section II. The framework and security models for LR-AMR-CB-KE are defined in Section III. Section IV gives a concrete LR-AMR-CB-KE scheme. We demonstrate the security of our LR-AMR-CB-KE scheme in Section V. We compare the performance with several existing schemes in Section VI. Section VII shows conclusions.

II. PRELIMINARIES

A. BILINEAR GROUPS

Let two groups G_1 and G_2 of the same prime order q be multiplicative cyclic. Let $\hat{e} : G_1 \times G_1 \rightarrow G_2$ be an associated bilinear map, and a point g be a generator of G_1. A bilinear group includes the mentioned parameters, namely, q, G_1, G_2, $\hat{e}$, g. Moreover, we say that $\hat{e}$ is a bilinear map if the following three properties hold.

- Bilinearity: the equation $\hat{e}(g^x, g^y) = \hat{e}(g, g)^{xy}$ holds, for all $x, y \in Z_q^*$.
- Non-degeneracy: $\hat{e}(g, g) \neq 1$
- Computability: for all $X, Y \in G_1$, there exists an algorithm that can compute $\hat{e}(X, Y)$ efficiently.

For more details about bilinear groups, the reader can refer to the literature [11].

B. GENERIC BILINEAR GROUP MODEL

The concept of generic bilinear group (GBG) model was proposed by Boneh et al. [48] and used in the security analysis of cryptographic schemes. The security analysis includes a known difficult problem, a security game and a concrete scheme. In the GBG model, a challenger in the security game is able to execute all operations of a bilinear group. To discuss the length of bits that secret keys can be leaked, we must express all elements in G_1 and G_2 by the form of bit-strings. Thus, we hire two injective mapping functions F_1: $Z_q^* \rightarrow \mathbb{B}G_1$ and F_2: $Z_q^* \rightarrow \mathbb{B}G_2$ to transform all elements in G_1 and G_2, where $\mathbb{B}G_1$ and $\mathbb{B}G_2$ are, respectively, the sets of transformed bit-strings of G_1 and G_2. Here, $\mathbb{B}G_1$ and $\mathbb{B}G_2$ are disjoint and satisfy $|\mathbb{B}G_1| = |\mathbb{B}G_2| = q$. Indeed, three operations of a bilinear group are the multiplications of G_1 and G_2 and a bilinear map $\hat{e}$. We present these three operations as follows.

- $OP_{G_1}(F_1(r), F_1(s)) \rightarrow F_1(r + s \ mod \ q)$.
- $OP_{G_2}(F_2(r), F_2(s)) \rightarrow F_2(r + s \ mod \ q)$.
- $OP_{\hat{e}}(F_1(r), F_1(s)) \rightarrow F_2(r \cdot s \ mod \ q)$.

Here, $r, s \in Z_q^*$, $g = F_1(1)$ and $\hat{e}(g, g) = F_2(1)$.

C. COMPLEXITY ASSUMPTIONS

According to the problem of discrete logarithm (DL) and the characteristics of hash function (HF), we give two associated assumptions as follows.

Definition 1 (DL assumption): Given the parameters q, G_1, G_2, $\hat{e}$, g of a bilinear group, the DL problem is to compute $c \in Z_q^*$ from g^c or g_2^c, where c is an unknown value and $g_2 = \hat{e}(g, g)$. Assume that $\mathcal{A}$ is a probabilistic polynomial-time (PPT) adversary. $\mathcal{A}$ has the negligible advantage $Adv_{\mathcal{A}} = \Pr[\mathcal{A}(g, g^c) = c]$ to find the solution of the DL problem.

Definition 2 (HF assumption): Given a secure HF, namely, $HF : Z_q^* \rightarrow \{0, 1\}^*$, the following three restrictions must be met.

(1) One-way property: Given a value $d \in Z_q^*$, it is difficult to find a bit-string $D \in \{0, 1\}^*$ such that $HF(D) = d$.
(2) Weak-collision resistant: Assume that there exists a bit-string $D_1 \in \{0, 1\}^*$. It is hard to find another bit-string $D_2 \in \{0, 1\}^*$ such that $HF(D_1) = HF(D_2)$.
(3) Strong-collision resistant: To find two different bit-strings D_1 and $D_2 \in \{0, 1\}^*$ such that $HF(D_1) = HF(D_2)$ is hard.

D. ENTROPY

In a leakage-resilient cryptographic scheme, the system's secret key or entity's secret key is allowed some partial information to be leaked to attackers. When attackers cannot obtain any leaked information, the uncertainty of guessing the complete secret key is high. However, if the amount of leaked information that attackers can obtain is increased, the uncertainty of guessing the complete secret key will be reduced. Indeed, we can regard the system's secret key or entity's secret key as a finite random variable. Thus, the

TABLE 1. Symbols.

Symbol	Meaning
SP	The system parameters
CPK	The CA's public key
CSK_0	The CA's initial secret key
ESK_0	The entity's initial secret key
EC_0	The entity's initial certificate
CSK_i	The CA's secret key in the i-th session
ESK_k	The entity's secret key in the k-th session
EC_k	The entity's certificate in the k-th session
EPK^{1st}	The entity's first public key
EPK^{2nd}	The entity's second public key

concept of entropy can be used to measure the security of the leakage-resilient cryptographic scheme. Assume that S is a finite random variable. Two min-entropy types are given as below.

1. The min-entropy of S is
$$H_\infty(S) = -\log_2(\max_s \Pr[S = s])$$
2. The average conditional min-entropy of S with a condition C is
$$\widetilde{H}_\infty(S|C) = -\log_2(C[\max_s \Pr[S = s|C]]).$$

Based on the two min-entropy types, the following Lemma 1 was presented by Dodis et al. [49] to measure the security of a single secret key (finite random variable) is leaked. Moreover, Galindo and Vivek [50] presented Lemma 2 to measure the security of multiple secret keys (finite multiple random variables) are leaked.

Lemma 1: Assume that S is a random variable and ψ is the maximal bit-length of leaked information. Let $f : S \rightarrow \{0, 1\}^\psi$ be a leakage function, then the inequality $\widetilde{H}_\infty(S|f(S)) \geqq H_\infty(S) - \psi$ holds.

Lemma 2: Assume that $S_1, S_2, \ldots, S_n$ are finite multiple random variables and $S \in Z_q[S_1, S_2, \ldots, S_n]$ is a polynomial with at most d-degree. Let $D_1, D_2, \ldots, D_n$ be probability distributions of $S_1 = s_1, S_2 = s_2, \ldots, S_n = s_n$ such that $H_\infty(D_i) \geqq \log q - \psi$ and $0 \leqq \psi \leqq \log q$ for $i \in [1, n]$. When $s_i \overset{D_i}{\leftarrow} Z_q$, for $i \in [1, n]$, are independent and $\psi \leqq (1-\epsilon)\log q$, the inequality $\Pr[S(S_1 = s_1, S_2 = s_2, \ldots, S_n = s_n) = 0] \leqq (d/q)2^\psi$ holds, where ϵ is a positive fraction.

E. SYMBOLS

Many symbols will be used in the LR-AMR-CB-KE. To facilitate readers' reading, we compile these symbols in Table 1.

III. FRAMEWORK AND SECURITY MODELS

A. FRAMEWORK

To resist side-channel attacks, we propose a new framework of LR-AMR-CB-KE schemes based on the framework of AMR-CBE scheme [34]. The proposed framework includes two stages. The first stage, depicted in Fig. 1, describes the system settings and each entity's public key and secret key settings. The system is executed by a certificate authority (CA) who generates the system parameters SP. In addition, the CA also has her/his initial secret key $CSK_0 = (CSK_{0,1}, CSK_{0,2})$. By the system parameter SP, each entity can set her/his own initial secret key $ESK_0 = (ESK_{0,1}, ESK_{0,2})$ and first public key EPK^{1st}. Further, the first public key EPK^{1st} is sent to the CA. In the i-th session, the CA updates the current secret key $CSK_{i-1} = (CSK_{i-1,1}, CSK_{i-1,2})$ to $CSK_i = (CSK_{i,1}, CSK_{i,2})$. Then, the CA generates the entity's second public key EPK^{2nd} and certificate EC which are sent to the entity. Immediately, the entity uses her/his own certificate EC to compute the initial certificate $EC_0 = (EC_{0,1}, EC_{0,2})$. Meanwhile, the entity sets her/his public key $EPK = (EPK^{1st}, EPK^{2nd})$.

The second stage, depicted in Fig. 2, is the procedures of multi-receiver encryption and decryption. A sender chooses a plaintext msg and (ID_1, EPK_1), (ID_2, EPK_2),..., (ID_n, EPK_n) of n entities, and the sender runs the *Multiencryption* algorithm to generate the ciphertext CT, namely $CT = ME(msg, (ID_1, EPK_1), (ID_2, EPK_2),\ldots, (ID_n, EPK_n))$. When each receiver receives the ciphertext CT, she/he can decrypt it with her/his own updated secret key $ESK_k = (ESK_{k,1}, ESK_{k,2})$ and certificate $EC_k = (EC_{k,1}, EC_{k,2})$ to obtain the plaintext msg by running the *Decryption* algorithm in the k-th session, namely $msg = D(msg, ESK_k = (ESK_{k,1}, ESK_{k,2}), EC_k = (EC_{k,1}, EC_{k,2}))$. It is worth noting that each receiver cannot know the identity information of other receivers. Now we formally define framework of LR-AMR-CB-KE schemes.

Definition 3: A LR-AMR-CB-KE scheme consists of five algorithms, namely, *Setup*, *Key generation*, *Certificate generation*, *Multiencryption* and *Decryption*, as follows.

- *Setup*: With the input of a security parameter τ, this algorithm outputs the CA's initial secret key $CSK_0 = (CSK_{0,1}, CSK_{0,2})$ and publishes the system parameters SP. Here, the system parameters SP includes the encryption $E_{sk}()$ and decryption $D_{sk}()$ algorithms from a symmetric encryption mechanism, where sk is a symmetric key.
- *Key generation*: With the input of the system parameter SP, an entity with identity ID runs this algorithm to generate the entity's initial secret key $ESK_0 = (ESK_{0,1}, ESK_{0,2})$ and first public key EPK^{1st}. The entity then sends EPK^{1st} to the CA.
- *Certificate generation*: In the i-th session of running this algorithm, with the input of the system parameter SP, the CA's current secret key $CSK_{i-1} = (CSK_{i-1,1}, CSK_{i-1,2})$, an entity's identity ID and the first public key EPK^{1st}, the CA generates the entity's second public key EPK^{2nd} and certificate EC. The CA then sends EPK^{2nd} and EC to the entity. Then, the entity respectively sets her/his initial certificate and complete public key as $EC_0 = (EC_{0,1}, EC_{0,2})$ and $EPK = (EPK^{1st}, EPK^{2nd})$.
- *Multiencryption* (*Multiencapsulation*): With the input of the system parameter SP, a plaintext msg and $(ID_1, EPK_1), (ID_2, EPK_2),\ldots, (ID_n, EPK_n)$ of n entities,

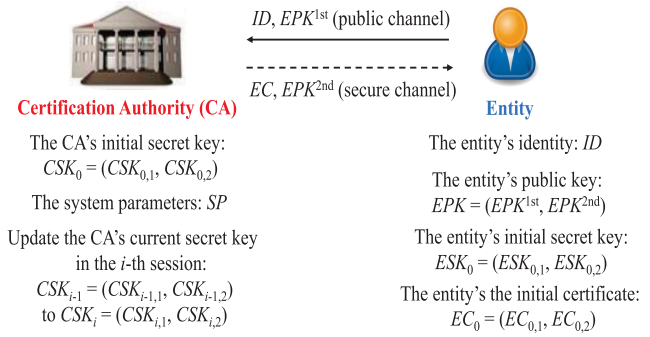


FIGURE 1. The system settings and each entity's public key and secret key settings.

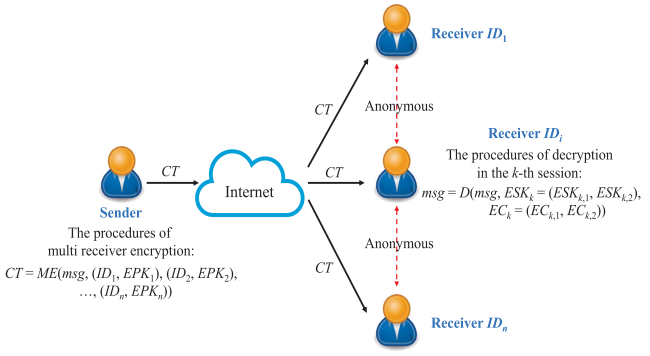


FIGURE 2. The procedures of anonymous multi-receiver encryption and decryption.

a sender runs this algorithm to generate the ciphertext CT.

- *Decryption* (*Decapsulation*): In the k-th session of running this algorithm, with the input of the system parameter SP, the ciphertext CT, the receiver's current secret key $ESK_{k-1} = (ESK_{k-1,1}, ESK_{k-1,2})$ and current certificate $EC_{k-1} = (EC_{k-1,1}, EC_{k-1,2})$, the receiver updates ESK_{k-1} and EC_{k-1} to $ESK_k = (ESK_{k,1}, ESK_{k,2})$ and $EC_k = (EC_{k,1}, EC_{k,2})$. Then, the receiver runs this algorithm to generate the plaintext msg.

B. SECURITY MODELS

We modify the security models of AMR-CBE scheme [34] and LR-CBE scheme [46], [47] to define the new security models of LR-AMR-CB-KE scheme. During each execution of the algorithms, any attacker is allowed to obtain part of the secret keys by side-channel attacks. More specifically, the attacker can obtain part of the CA's current secret key $CSK_i = (CSK_{i,1}, CSK_{i,2})$ in the i-th session of running the *Certificate generation* algorithm. And, the attacker can obtain part of the entity's current secret key $ESK_k = (ESK_{k,1}, ESK_{k,2})$ and current certificate $EC_k = (EC_{k,1}, EC_{k,2})$ in the k-th session of running the *Decryption* algorithm.

To model the ability of an attacker to obtain part of the secret key, four leaked functions $LF^I_{CG,i}$, $LF^{II}_{CG,i}$, $LF^I_{D,k}$ and $LF^{II}_{D,k}$ are employed. The first two functions $LF^I_{CG,i}$ and $LF^{II}_{CG,i}$ are used to express the attacker's ability to obtain part of the CA's current secret key $CSK_i = (CSK_{i,1}, CSK_{i,2})$ of the *Certificate generation* algorithm in the i-th session. The latter two functions $LF^I_{D,k}$ and $LF^{II}_{D,k}$ are used to express the attacker's ability to obtain part of the entity's current secret key $ESK_k = (ESK_{k,1}, ESK_{k,2})$ and current certificate $EC_k = (EC_{k,1}, EC_{k,2})$ of the *Decryption* algorithm in the k-th session. Assume that ψ is the maximal bit-length of part of the secret key. We have $|LF^I_{CG,i}|$, $|LF^{II}_{CG,i}|$, $|LF^I_{D,k}|$ and $|LF^{II}_{D,k}| \leq \psi$, where $|\cdot|$ is the output length of these leaked functions. In the following, we formally define these four leaked functions.

- $\Lambda LF^I_{CG,i} = LF^I_{CG,i}(CSK_{i,1})$.
- $\Lambda LF^{II}_{CG,i} = LF^{II}_{CG,i}(CSK_{i,2})$.
- $\Lambda LF^I_{D,k} = LF^I_{D,k}(ESK_{k,1}, EC_{k,1})$.
- $\Lambda LF^{II}_{D,k} = LF^{II}_{D,k}(ESK_{k,2}, EC_{k,2})$.

As the security models of AMR-CBE scheme [34] and LR-CBE scheme [46], [47], the security models of LR-AMR-CB-KE scheme includes the following two types of adversaries.

- Type I adversary $\mathcal{A}_I$ can run the *Key generation* algorithm to generate the entity's secret key, but $\mathcal{A}_I$ is unable to obtain the entity's certificate or the CA's secret key. However, $\mathcal{A}_I$ can obtain part of the entity's certificate and the CA's secret key.
- Type II adversary $\mathcal{A}_{II}$ possesses the CA's secret key. Hence, any entity's certificate can be generated by $\mathcal{A}_{II}$. However, $\mathcal{A}_{II}$ is unable to obtain the entity's secret key. But, $\mathcal{A}_{II}$ can obtain part of the entity's secret key.

Next, we give two security models of LR-AMR-CB-KE scheme. The first is to simulate the security of the indistinguishability of two ciphertexts against chosen ciphertext attacks (IND-CCA) in the continual leakage model. The other is to simulate the security of the anonymous indistinguishability of two identities against chosen ciphertext attacks (ANON-IND-CCA) in the continual leakage model.

Definition 4: A LR-AMR-CB-KE scheme is secure for the indistinguishability of two ciphertexts against chosen ciphertext attacks (IND-CCA) in the continual leakage model if no adversary $\mathcal{A}$ (including Type I adversary $\mathcal{A}_I$ and Type II adversary $\mathcal{A}_{II}$), who possesses a non-negligible advantage, can win the following security game $G^{IND-CCA}_{LR-AMR-CB-KE}$ with a challenger $\mathcal{C}$ in a probabilistic polynomial time.

- *Setup phase*: With the input of a security parameter τ, the challenger $\mathcal{C}$ runs the *Setup* algorithm to output the CA's initial secret key $(CSK_{0,1}, CSK_{0,2})$ and the system parameters SP. For the adversary $\mathcal{A}_I$, $\mathcal{C}$ keeps the CA's initial secret key $(CSK_{0,1}, CSK_{0,2})$ by itself. Otherwise, $\mathcal{C}$ sends the CA's initial secret key $(CSK_{0,1}, CSK_{0,2})$ to the adversary $\mathcal{A}_{II}$. The system parameter SP is received by both $\mathcal{A}_I$ and $\mathcal{A}_{II}$.
- *Query phase*: The following different queries can be adaptively asked by the adversary $\mathcal{A}$.

- *Key generation query* (ID): $\mathcal{A}$ sends out the query with an identity ID. Then, $\mathcal{C}$ runs the *Key generation* algorithm with ID to generate the entity's secret key ESK and first public key EPK^{1st}. The challenger $\mathcal{C}$ utilizes ESK to set the entity's initial secret key ($ESK_{0,1}$, $ESK_{0,2}$). Finally, ($ESK_{0,1}$, $ESK_{0,2}$) is transmitted to $\mathcal{A}$ if the *Public key replace query* (ID) has never been sent out by $\mathcal{A}$.
- *Certificate generation query* (ID, EPK^{1st}): $\mathcal{A}$ sends out the query in the i-th session with an identity ID and the associated first public key EPK^{1st}. Then, $\mathcal{C}$ runs the *Certificate generation* algorithm with ID, EPK^{1st} and the CA's current secret key ($CSK_{i-1,1}$, $CSK_{i-1,2}$) to generate the entity's second public key EPK^{2nd} and certificate EC. The challenger $\mathcal{C}$ utilizes EC to set the entity's initial certificate ($EC_{0,1}$, $EC_{0,2}$). Both entity's initial certificate ($EC_{0,1}$, $EC_{0,2}$) and second public key EPK^{2nd} are transmitted to $\mathcal{A}$.
- *Certificate generation leak query* (i, $LF^{I}_{CG,i}$, $LF^{II}_{CG,i}$): $\mathcal{A}$ sends out the query in the i-th session with two leaked functions $LF^{I}_{CG,i}$ and $LF^{II}_{CG,i}$. Then, $\mathcal{C}$ returns the two leaked information $\Lambda LF^{I}_{CG,i}$ and $\Lambda LF^{II}_{CG,i}$ to $\mathcal{A}$. Here, the two leaked information $\Lambda LF^{I}_{CG,i}$ and $\Lambda LF^{II}_{CG,i}$ are used to indicate the leakage length of the CA's current secret key ($CSK_{i-1,1}$, $CSK_{i-1,2}$).
- *Public key retrieve query* (ID): $\mathcal{A}$ sends out the query with an identity ID. Then, $\mathcal{C}$ returns the associated public key $EPK = (EPK^{1st}, EPK^{2nd})$ to $\mathcal{A}$.
- *Public key replace query* (ID, ($rEPK^{1st}$, $rEPK^{2nd}$)): $\mathcal{A}$ sends out the query with an identity ID and the replaced public key ($rEPK^{1st}$, $rEPK^{2nd}$). Then, $\mathcal{C}$ records the public key replacement information of the identity ID.
- *Decryption (Decapsulation) query* (ID, CT): $\mathcal{A}$ sends out the query in the k-th session with an identity ID and the associated ciphertext CT. Then, $\mathcal{C}$ updates the associated current secret key $ESK_{k-1} = (ESK_{k-1,1}, ESK_{k-1,2})$ and current certificate $EC_{k-1} = (EC_{k-1,1}, EC_{k-1,2})$ to $ESK_k = (ESK_{k,1}, ESK_{k,2})$ and $EC_k = (EC_{k,1}, EC_{k,2})$. Finally, the challenger $\mathcal{C}$ returns the plaintext msg by running the *Decryption* algorithm with ESK_k and EC_k.
- *Decryption* (*Decapsulation*) *leak query* (ID, k, $LF^{I}_{D,k}$, $LF^{II}_{D,k}$): $\mathcal{A}$ sends out the query in the k-th session with two leaked functions $LF^{I}_{D,k}$ and $LF^{II}_{D,k}$. Then, $\mathcal{C}$ returns the two leaked information $\Lambda LF^{I}_{D,k}$ and $\Lambda LF^{II}_{D,k}$ to $\mathcal{A}$. Here, the two leaked information $\Lambda LF^{I}_{D,k}$ and $\Lambda LF^{II}_{D,k}$ are used to indicate the leakage length of the entity's current certificate ($EC_{k-1,1}$, $EC_{k-1,2}$) if the adversary is $\mathcal{A}_I$. The two leaked information $\Lambda LF^{I}_{D,k}$ and $\Lambda LF^{II}_{D,k}$ are used to indicate the leakage length of the entity's current secret key ($ESK_{k-1,1}$, $ESK_{k-1,2}$) if the adversary is $\mathcal{A}_{II}$.

- *Challenge phase*: $\mathcal{A}$ sends out two target plaintexts msg_0, msg_1 and n identities of entities, namely ID^*_1, ID^*_2,. . ., ID^*_n. Then, $\mathcal{C}$ picks a random bit $b \in \{0, 1\}$, and returns the target ciphertext CT^* by running the *Multiencryption* algorithm with msg_b, (ID^*_1, EPK^*_1), (ID^*_2, EPK^*_2),. . ., (ID^*_n, EPK^*_n). Since there are two types of adversaries, we must consider the following two situations.
  - If the adversary is $\mathcal{A}_I$, the *Certificate generation query* with ID^*_1, ID^*_2,. . ., ID^*_n is not allowed to be asked.
  - If the adversary is $\mathcal{A}_{II}$, the *Key generation query* with ID^*_1, ID^*_2,. . ., ID^*_n is not allowed to be asked.
- *Guess phase*: $\mathcal{A}$ sends out a guess $b' \in \{0, 1\}$. If the guess $b' = b$, $\mathcal{A}$ wins this the game. We define the advantage of winning this game as $Adv(\mathcal{A})$ = |Pr[$b' = b$] - 1/2|.

Definition 5: A LR-AMR-CB-KE scheme is secure for the anonymous indistinguishability of two identities against chosen ciphertext attacks (ANON-IND-CCA) in the continual leakage model if no adversary $\mathcal{A}$, who possesses a non-negligible advantage, can win the following security game $G^{ANON-IND-CCA}_{LR-AMR-CB-KE}$ with a challenger $\mathcal{C}$ in a probabilistic polynomial time.

- *Setup phase*: The challenger $\mathcal{C}$ performs the same procedure as *Setup phase* in the definition 4.
- *Query phase*: The adversary $\mathcal{A}$ performs the same procedure as *Query phase* in the definition 4.
- *Challenge phase*: $\mathcal{A}$ sends out a target plaintext msg and n identities of entities, namely ID^*_1, ID^*_2,. . ., ID^*_n. Then, $\mathcal{C}$ picks a random bit $b \in \{0, 1\}$, and returns the target ciphertext CT^* by running the *Multiencryption* algorithm with msg, (ID^*_b, EPK^*_1), (ID^*_3, EPK^*_2), (ID^*_4, EPK^*_4),. . ., (ID^*_n, EPK^*_n). Since there are two types of adversaries, we must consider the following two situations.
  - If the adversary is $\mathcal{A}_I$, the *Certificate generation query* with ID^*_1 and ID^*_2 is not allowed to be asked.
  - If the adversary is $\mathcal{A}_{II}$, the *Key generation query* with ID^*_1 and ID^*_2 is not allowed to be asked.
- *Guess phase*: $\mathcal{A}$ sends out a guess $b' \in \{0, 1\}$. If the guess $b' = b$, $\mathcal{A}$ wins this the game. We define the advantage of winning this game as $Adv(\mathcal{A})$ = |Pr[$b' = b$] - 1/2|.

IV. THE PROPOSED LR-AMR-CB-KE SCHEME

In this section, we present the first LR-AMR-CB-KE scheme which includes the following five algorithms, namely, *Setup*, *Key generation*, *Certificate generation*, *Multiencryption* and *Decryption*.

- *Setup*: With the input of a security parameter τ, this algorithm outputs the CA's initial secret key $CSK_0 = (CSK_{0,1}, CSK_{0,2})$ and publishes the system parameter SP, which can be obtained by the following work.

- Choose two multiplicative cyclic groups G_1 and G_2 of the same prime order q. Let $\hat{e} : G_1 \times G_1 \rightarrow G_2$ be an associated bilinear map.
- Pick a point g as a generator of G_1 and compute the CA's secret key $CSK = g^s$, where s is a random value in Z_q^*. Then, set the CA's public key $CPK = \hat{e}(g^s, g)$.
- Randomly choose a refreshing value $ra \in Z_q^*$ to set the CA's initial secret key $CSK_0 = (CSK_{0,1}, CSK_{0,2}) = (g^{ra}, g^{-ra} \cdot CSK)$.
- Pick two random values $m, n \in Z_q^*$ to compute $M = g^m$ and $N = g^n$.
- Select encryption $E_{sk}()$ and decryption $D_{sk}()$ algorithms from a symmetric encryption mechanism, where sk is a symmetric key.
- Choose five hash functions, $H_0 : G_2 \times G_1 \rightarrow \{0,1\}^\lambda$, $H_1, H_2, H_3 : \{0,1\}^\lambda \rightarrow \{0,1\}^\lambda$ and $H_4 : \{0,1\}^* \times G_1 \rightarrow \{0,1\}^\lambda$, where λ is a fixed length.
- Set the system parameters $SP = \{p, G_1, G_2, \hat{e}, g, CPK, M, N, E, D, H_0, H_1, H_2, H_3, H_4\}$.

- *Key generation*: With the input of the system parameter SP, an entity with identity ID runs this algorithm to generate the entity's secret key ESK and first public key EPK^{1st}, which can be obtained by the following work.
  - Randomly choose a value $\alpha \in Z_q^*$ and set $ESK = g^\alpha$.
  - Set $EPK^{1st} = \hat{e}(ESK, g) = \hat{e}(g^\alpha, g)$.

  Finally, the entity randomly picks a refreshing value $rb \in Z_q^*$ to set the entity's initial secret key $ESK_0 = (ESK_{0,1}, ESK_{0,2}) = (g^{rb}, g^{-rb} \cdot ESK)$.
- *Certificate generation*: In the i-th session of running this algorithm, with the input of the system parameter SP, the CA's current secret key $CSK_{i-1} = (CSK_{i-1,1}, CSK_{i-1,2})$, an entity's identity ID and the first public key EPK^{1st}, the CA generates the entity's second public key EPK^{2nd} and certificate EC, which can be obtained by the following work.
  - Randomly choose a refreshing value $rc \in Z_q^*$ to update the CA's current secret key $CSK_i = (CSK_{i,1}, CSK_{i,2}) = (g^{rc} \cdot CSK_{i-1,1}, g^{-rc} \cdot CSK_{i-1,2})$.
  - Set $E = ID||EPK^{1st}$ and randomly pick a value $\beta \in Z_q^*$. Compute the entity's second public key $EPK^{2nd} = g^\beta$, temporary key $EC_{tmp} = CSK_{i,1} \cdot (M \cdot N^E)^\beta$ and certificate $EC = CSK_{i,2} \cdot EC_{tmp}$.

  The CA then transmits the entity's second public key EPK^{2nd} and certificate EC to the entity. Afterwards, the entity randomly chooses a refreshing value $rd \in Z_q^*$ to set her/his initial certificate $EC_0 = (EC_{0,1}, EC_{0,2}) = (g^{rd}, g^{-rd} \cdot EC)$. Finally, the entity sets her/his public key $EPK = (EPK^{1st}, EPK^{2nd})$.
- *Multiencryption* (*Multiencapsulation*): With the input of the system parameter SP, a plaintext msg and $(ID_1, EPK_1), (ID_2, EPK_2), \ldots, (ID_n, EPK_n)$ of n entities, a sender runs this algorithm to generate the ciphertext CT, which can be obtained by the following work.
  - Pick a random value $r \in Z_q^*$, and compute $R = g^r$, $U_i = (EPK_i^{1st})^r$, $V_i = (CPK \cdot \hat{e}(EPK_i^{2nd}, M \cdot N^{E_i}))^r$ and $K_i = H_0(V_i, U_i)$, where $E_i = ID_i||EPK_i^{1st}$ for $i = 1, 2, \ldots, n$.
  - Randomly choose $\omega \in \{0,1\}^\lambda$, and compute $W_i = H_1(K_i)||(H_2(K_i) \oplus \omega)$ for $i = 1, 2, \ldots, n$.
  - Set a symmetric key $sk = H_3(\omega)$, and generate $T = E_{sk}(msg)$ and $\sigma = H_4(msg, \omega, W_1, W_2, \ldots, W_n, R, T)$.
  - Set the ciphertext $CT = \{(W_1, W_2, \ldots, W_n), R, T, \sigma\}$.
- *Decryption* (*Decapsulation*): In the k-th session of running this algorithm, with the input of the system parameter SP, the ciphertext CT and the receiver's current secret key $ESK_{k-1} = (ESK_{k-1,1}, ESK_{k-1,2})$ and current certificate $EC_{k-1} = (EC_{k-1,1}, EC_{k-1,2})$, the receiver runs this algorithm to generate the plaintext msg, which can be obtained by the following work.
  - Randomly choose two refreshing values re and $rf \in Z_q^*$ to update the receiver's current secret key $ESK_k = (ESK_{k,1}, ESK_{k,2}) = (g^{re} \cdot ESK_{k-1,1}, g^{-re} \cdot ESK_{k-1,2})$ and current certificate $EC_k = (EC_{k,1}, EC_{k,2}) = (g^{rf} \cdot EC_{k-1,1}, g^{-rf} \cdot EC_{k-1,2})$, respectively.
  - Compute $U_{tmp} = \hat{e}(R, ESK_{k,1})$, and set $U = U_{tmp} \cdot \hat{e}(R, ESK_{k,2})$.
  - Compute $V_{tmp} = \hat{e}(R, EC_{k,1})$, and set $V = V_{tmp} \cdot \hat{e}(R, EC_{k,2})$.
  - Compute $K = H_0(V, U)$ and $H_1(K)$.
  - Use the ciphertext CT and $H_1(K)$ to find the associated W_i, for $i \in [1, n]$. Then, a result $\phi = H_2(K) \oplus \omega$ can be obtained due to $W = H_1(K)||(H_2(K) \oplus \omega)$.
  - Recover $\omega' = \phi \oplus H_2(K)$.
  - Set a symmetric key $sk' = H_3(\omega')$, and generate $msg' = E'_{sk}(T)$.
  - Compute $\sigma' = H_4(msg', \omega', W_1, W_2, \ldots, W_n, R, T)$. If $\sigma' = \sigma$, return msg'; otherwise return a symbol "$\perp$".

Below, we show that the value of K calculated during the *Decryption* process is equal to the value of K_i for $i \in [1, n]$ used during the *Multiencryption* process.

$$\begin{aligned}
K &= H_0(V, U) \\
&= H_0(\hat{e}(R, EC_k), \hat{e}(R, ESK_k)) \\
&= H_0(\hat{e}(g^r, CSK_i \cdot (M \cdot N^E)^\beta), \hat{e}(g^r, g^\alpha)) \\
&= H_0(\hat{e}(g^r, g^s \cdot (M \cdot N^E)^\beta), \hat{e}(g^r, g^\alpha)) \\
&= H_0(\hat{e}(g^r, g^s) \cdot \hat{e}(g^r, (M \cdot N^E)^\beta), \hat{e}(g^r, g^\alpha)) \\
&= H_0((\hat{e}(g^s, g) \cdot \hat{e}(g^\beta, M \cdot N^{E_i}))^r, \hat{e}(g^\alpha, g)^r) \\
&= H_0((CPK \cdot \hat{e}(EPK_i^{2nd}, M \cdot N^{E_i}))^r, (EPK_i^{1st})^r) \\
&= H_0(V_i, U_i) = K_i
\end{aligned}$$

V. SECURITY ANALYSIS

In this section, we use Theorems 1 and 2 to demonstrate that the proposed LR-AMR-CB-KE scheme is secure against the adversary $\mathcal{A}_I$ and $\mathcal{A}_{II}$ for the indistinguishability of two

ciphertexts against chosen ciphertext attacks (IND-CCA). Moreover, Theorems 3 and 4 are used to demonstrate that the proposed LR-AMR-CB-KE scheme is secure against the adversary $\mathcal{A}_I$ and $\mathcal{A}_{II}$ for the anonymous indistinguishability of two identities against chosen ciphertext attacks (ANON-IND-CCA).

Theorem 1: Based on the DL and HF assumptions, the proposed LR-AMR-CB-KE scheme is secure under the GBG model against the adversary $\mathcal{A}_I$ for the indistinguishability of two ciphertexts against chosen ciphertext attacks (IND-CCA) in the security game $G^{IND-CCA}_{LR-AMR-CB-KE}$.

Proof. Assume that there exists an adversary $\mathcal{A}_I$ who plays the following security game $G^{IND-CCA}_{LR-AMR-CB-KE}$ with a challenger $\mathcal{C}$.

- *Setup phase*: With the input of a security parameter τ, the challenger $\mathcal{C}$ runs the *Setup* algorithm to output the CA's initial secret key $CSK_0 = (CSK_{0,1}, CSK_{0,2})$, generated by the CA's secret key CSK, and the system parameter $SP = \{p, G_1, G_2, \hat{e}, g, CPK, M, N, E, D, H_0, H_1, H_2, H_3, H_4\}$. Since the adversary is $\mathcal{A}_I$, C keeps the CA's initial secret key $CSK_0 = (CSK_{0,1}, CSK_{0,2})$ by itself. Additionally, five lists $L_1, L_2, L_{ESK}, L_{EC}$ and L_{ME}, empty at the beginning, are created by the challenger $\mathcal{C}$ as below.
  - List L_1 is created to record all elements of G_1. We first define three symbols μ, v and s as the type μ, the v-th query and s-th element. Each element in the list L_1 is rendered in the format $(\mathbb{P}G_{1,\mu,v,s}, \mathbb{B}G_{1,\mu,v,s})$, where $\mathbb{P}G_{1,\mu,v,s}$ is multivariate polynomial and $\mathbb{B}G_{1,\mu,v,s}$ is the corresponding bit-string. Due to the CA's secret key CSK and the system parameter SP, we add four elements $(\mathbb{P}CSK, \mathbb{B}G_{1,I,0,1})$, $(\mathbb{P}g, \mathbb{B}G_{1,I,0,2})$, $(\mathbb{P}M, \mathbb{B}G_{1,I,0,3})$ and $(\mathbb{P}N, \mathbb{B}G_{1,I,0,4})$ into the list L_1.
  - List L_2 is created to record all elements of G_2. The definitions of three symbols μ, v and s are identical to the above. Each element in the list L_2 is rendered in the format $(\mathbb{P}G_{2,\mu,v,s}, \mathbb{B}G_{2,\mu,v,s})$, where $\mathbb{P}G_{2,\mu,v,s}$ is multivariate polynomial and $\mathbb{B}G_{2,\mu,v,s}$ is the corresponding bit-string. Due to the system parameter SP, we add one element $(\mathbb{P}CPK, \mathbb{B}G_{1,I,0,1})$ into the list L_2.
    It can be observed that in lists L_1 and L_2, each element is represented as multivariate polynomial and the corresponding bit-string. Therefore, we give two conversion formulas, namely CF-1 and CF-2, to describe the conversion between multivariate polynomial and the corresponding bit-string.
    - ✓ CF-1: When the challenger $\mathcal{C}$ receives $\mathbb{P}G_{1,\mu,v,s}$ / $\mathbb{P}G_{2,\mu,v,s}$, she/he can use it to search the list L_1/L_2 and find the corresponding $\mathbb{B}G_{1,\mu,v,s}$ / $\mathbb{B}G_{2,\mu,v,s}$ if $\mathbb{P}G_{1,\mu,v,s}$ / $\mathbb{P}G_{2,\mu,v,s}$ exists in the list L_1/L_2. Then, $\mathcal{C}$ returns $\mathbb{B}G_{1,\mu,v,s}$ / $\mathbb{B}G_{2,\mu,v,s}$. If $\mathbb{P}G_{1,\mu,v,s}$ / $\mathbb{P}G_{2,\mu,v,s}$ does not exist in the list L_1/L_2, $\mathcal{C}$ generates a corresponding bit-string $\mathbb{B}G_{1,\mu,v,s}$ / $\mathbb{B}G_{2,\mu,v,s}$, which will be returned and added to the L_1/L_2.
    - ✓ CF-2: When the challenger $\mathcal{C}$ receives $\mathbb{B}G_{1,\mu,v,s}$ / $\mathbb{B}G_{2,\mu,v,s}$, she/he can use it to search the list L_1/L_2 and find the corresponding $\mathbb{P}G_{1,\mu,v,s}$ / $\mathbb{P}G_{2,\mu,v,s}$ if $\mathbb{B}G_{1,\mu,v,s}$ / $\mathbb{B}G_{2,\mu,v,s}$ exists in the list L_1/L_2. Then, $\mathcal{C}$ returns $\mathbb{P}G_{1,\mu,v,s}$ / $\mathbb{P}G_{2,\mu,v,s}$. If $\mathbb{B}G_{1,\mu,v,s}$ / $\mathbb{B}G_{2,\mu,v,s}$ does not exist in the list L_1/L_2, $\mathcal{C}$ terminates the game.
  - List L_{ESK} is created to record each entity's secret key ESK and first public key EPK^{1st}. The identity ID, ESK and EPK^{1st} of each entity is rendered in the format $(ID, \mathbb{P}ESK, \mathbb{P}EPK^{1st})$.
  - List L_{EC} is created to record each entity's certificate EC and second public key EPK^{2nd}. The identity ID, EC and EPK^{2nd} of each entity is rendered in the format $(ID, \mathbb{P}EC, \mathbb{P}EPK^{2nd})$.
  - List L_{ME} is created to record the values used in the *Multiencryption* algorithm. The values are rendered in the format $(\mathbb{P}U, \mathbb{P}V, K, H_1(K), H_2(K), \omega, H_3(\omega) = sk)$, where K, $H_1(K)$, $H_2(K)$, ω and $H_3(\omega) = sk$ are bit-strings.
- *Query phase*: The following different queries can be adaptively asked by the adversary $\mathcal{A}_I$ at most p times. Additionally, three group operation queries, namely OP_{G_1}, OP_{G_2} and $OP_{\hat{e}}$, can also be asked by the adversary $\mathcal{A}_I$.
  - OP_{G_1} *query* $(\mathbb{B}G_{1,o,r,i}, \mathbb{B}G_{1,o,r,j}, Cmd)$: $\mathcal{A}_I$ sends out the r-th query with $\mathbb{B}G_{1,o,r,i}$, $\mathbb{B}G_{1,o,r,j}$ and Cmd. The challenger $\mathcal{C}$ returns $\mathbb{B}G_{1,o,r,k}$, which can be obtained by the following work.
    (1) According to CF-2, convert $\mathbb{B}G_{1,o,r,i}$ and $\mathbb{B}G_{1,o,r,j}$ to $\mathbb{P}G_{1,o,r,i}$ and $\mathbb{P}G_{1,o,r,j}$.
    (2) Compute $\mathbb{P}G_{1,o,r,k} = \mathbb{P}G_{1,o,r,i} + \mathbb{P}G_{1,o,r,j}$ if $Cmd =$ "*multiplication*". Compute $\mathbb{P}G_{1,o,r,k} = \mathbb{P}G_{1,o,r,i}$ - $\mathbb{P}G_{1,o,r,j}$ if $Cmd =$ "*division*".
    (3) According to CF-1, convert $\mathbb{P}G_{1,o,r,k}$ to $\mathbb{B}G_{1,o,r,k}$.
  - OP_{G_2} *query* $(\mathbb{B}G_{2,o,r,i}, \mathbb{B}G_{2,o,r,j}, Cmd)$: $\mathcal{A}_I$ sends out the r-th query with $\mathbb{B}G_{2,o,r,i}$, $\mathbb{B}G_{2,o,r,j}$ and Cmd. The challenger $\mathcal{C}$ returns $\mathbb{B}G_{2,o,r,k}$, which can be obtained by the following work.
    (1) According to CF-2, convert $\mathbb{B}G_{2,o,r,i}$ and $\mathbb{B}G_{2,o,r,j}$ to $\mathbb{P}G_{2,o,r,i}$ and $\mathbb{P}G_{2,o,r,j}$.
    (2) Compute $\mathbb{P}G_{2,o,r,k} = \mathbb{P}G_{2,o,r,i} + \mathbb{P}G_{2,o,r,j}$ if $Cmd =$ "*multiplication*". Compute $\mathbb{P}G_{2,o,r,k} = \mathbb{P}G_{2,o,r,i}$ $\mathbb{P}G_{2,o,r,j}$ if $Cmd =$ "*division*".
    (3) According to CF-1, convert $\mathbb{P}G_{2,o,r,k}$ to $\mathbb{B}G_{2,o,r,k}$.
  - $OP_{\hat{e}}$ *query* $(\mathbb{B}G_{1,\hat{e},r,i}, \mathbb{B}G_{1,\hat{e},r,j})$: $\mathcal{A}_I$ sends out the r-th query with $\mathbb{B}G_{1,\hat{e},r,i}$ and $\mathbb{B}G_{1,\hat{e},r,j}$. The chal-

lenger $\mathcal{C}$ returns $\mathbb{B}G_{1,\hat{e},r,k}$, which can be obtained by the following work.

(1) According to CF-2, convert $\mathbb{B}G_{1,\hat{e},r,i}$ and $\mathbb{B}G_{1,\hat{e},r,j}$ to $\mathbb{P}G_{1,\hat{e},r,i}$ and $\mathbb{P}G_{1,\hat{e},r,j}$.
(2) Compute $\mathbb{P}G_{1,\hat{e},r,k} = \mathbb{P}G_{1,\hat{e},r,i} \cdot \mathbb{P}G_{1,\hat{e},r,j}$.
(3) According to CF-1, convert $\mathbb{P}G_{1,\hat{e},r,k}$ to $\mathbb{B}G_{1,\hat{e},r,k}$.

- *Key generation query* (*ID*): $\mathcal{A}_I$ sends out the query with an identity *ID*. Then, $\mathcal{C}$ can find $\mathbb{P}EC$ and $\mathbb{P}EPK^{1st}$ in the list L_{ESK} with respect to *ID*. According to CF-1, convert $\mathbb{P}EC$ and $\mathbb{P}EPK^{1st}$ to $\mathbb{B}EC$ and $\mathbb{B}EPK^{1st}$ which are the response.
- *Certificate generation query* (*ID*, EPK^{1st}): $\mathcal{A}_I$ sends out the query in the *i*-th session with an identity *ID* and the associated first public key EPK^{1st}. Then, $\mathcal{C}$ can find $\mathbb{P}EC$ and $\mathbb{P}EPK^{2nd}$ in the list L_{EC} with respect to *ID* and EPK^{1st}. If $\mathbb{P}EC$ and $\mathbb{P}EPK^{2nd}$ does not exist in the list L_{EC}, $\mathcal{C}$ converts $\mathbb{P}EC$ and $\mathbb{P}EPK^{2nd}$ to $\mathbb{B}EC$ and $\mathbb{B}EPK^{2nd}$, and takes them as the response. Otherwise, $\mathcal{C}$ does the following work.

(1) Choose a new variate $\mathbb{P}BG_{CG,i,1}$ in G_1, and set $\mathbb{P}EPK^{2nd} = \mathbb{P}BG_{CG,i,1}$.
(2) Compute $\mathbb{P}EC = \mathbb{P}CSK + \mathbb{P}BG_{CG,i,1} \cdot (\mathbb{P}M + \mathbb{P}E \cdot \mathbb{P}N)$, where $\mathbb{P}E = ID||EPK^{1st}$.
(3) Add $\mathbb{P}EC$ and $\mathbb{P}EPK^{2nd}$ into the list L_{EC}.
(4) Convert $\mathbb{P}EC$ and $\mathbb{P}EPK^{2nd}$ to $\mathbb{B}EC$ and $\mathbb{B}EPK^{2nd}$ which are the response.

- *Certificate generation leak query* (*i*, $LF^I_{CG,i}$, $LF^{II}_{CG,i}$): $\mathcal{A}_I$ sends out the query in the *i*-th session with two leaked functions $LF^I_{CG,i}$ and $LF^{II}_{CG,i}$. Then, $\mathcal{C}$ returns the two leaked information $\Lambda LF^I_{CG,i} = LF^I_{CG,i}(CSK_{i-1,1})$ and $\Lambda LF^{II}_{CG,i} = LF^{II}_{CG,i}(CSK_{i-1,2})$ to $\mathcal{A}_I$.
- *Public key retrieve query* (*ID*): $\mathcal{A}_I$ sends out the query with an identity *ID*. Then, $\mathcal{C}$ can respectively find $\mathbb{P}EPK^{1st}$ and $\mathbb{P}EPK^{2nd}$ in the lists L_{ESK} and L_{EC} with respect to *ID*. According to CF-1, convert $\mathbb{P}EPK^{1st}$ and $\mathbb{P}EPK^{2nd}$ to $\mathbb{B}EPK^{1st}$ and $\mathbb{B}EPK^{2nd}$ which are the response.
- *Public key replace query* (*ID*, ($\mathbb{B}rEPK^{1st}$, $\mathbb{B}rEPK^{2nd}$)): $\mathcal{A}_I$ sends out the query with an identity *ID* and the replaced public key ($\mathbb{B}rEPK^{1st}$, $\mathbb{B}rEPK^{2nd}$). According to CF-2, $\mathcal{C}$ converts $\mathbb{B}rEPK^{1st}$ and $\mathbb{B}rEPK^{2nd}$ to $\mathbb{P}rEPK^{1st}$ and $\mathbb{P}rEPK^{2nd}$. Then, $\mathcal{C}$ respectively records (*ID*, $-$, $\mathbb{P}rEPK^{1st}$) and (*ID*, $-$, $\mathbb{P}rEPK^{2nd}$) in the lists L_{ESK} and L_{EC}.
- *Decryption* (*Decapsulation*) *query* (*ID*, *CT*): $\mathcal{A}_I$ sends out the query in the *k*-th session with an identity *ID* and the associated ciphertext $CT = \{(W_1, W_2, \ldots, W_n), R, T, \sigma\}$. The challenger $\mathcal{C}$ returns the plaintext *msg* by doing the following work.

(1) With respect to *ID*, find the associated secret key $\mathbb{P}ESK$ in the list L_{ESK} and the associated certificate $\mathbb{P}EC$ in the list L_{EC}.
(2) According to CF-2, convert $\mathbb{B}R$ and $\mathbb{P}R$ in the list L_1. Compute $\mathbb{P}U = \mathbb{P}R \cdot \mathbb{P}ESK$ and $\mathbb{P}V = \mathbb{P}R \cdot \mathbb{P}EC$.
(3) Use $\mathbb{P}U$ and $\mathbb{P}V$ to find $H_3(\omega) = sk$ from ($\mathbb{P}U$, $\mathbb{P}V$, K, $H_1(K)$, $H_2(K)$, ω, $H_3(\omega) = sk$) in the list L_{ME}. Obtain the plaintext *msg* from computing $D_{sk}(T)$.

- *Decryption* (*Decapsulation*) *leak query* (*ID*, *k*, $LF^I_{D,k}$, $LF^{II}_{D,k}$): $\mathcal{A}_I$ sends out the query in the *k*-th session with two leaked functions $LF^I_{D,k}$ and $LF^{II}_{D,k}$. Then, $\mathcal{C}$ returns the two leaked information $\Lambda LF^I_{D,k} = LF^I_{D,k}(EC_{k-1,1})$ and $\Lambda LF^{II}_{D,k} = LF^{II}_{D,k}(EC_{k-1,2})$ to $\mathcal{A}_I$.

- *Challenge phase*: $\mathcal{A}_I$ sends out two target plaintexts msg_0, msg_1 and n identities of entities, namely ID^*_1, ID^*_2,..., ID^*_n. Since the adversary is $\mathcal{A}_I$, the *Certificate generation query* with ID^*_1, ID^*_2,..., ID^*_n is not allowed to be asked. The challenger $\mathcal{C}$ picks a random bit $b \in \{0, 1\}$, and returns the target ciphertext CT^* by running the *Multiencryption* algorithm with msg_b, (ID^*_1, EPK^*_1), (ID^*_2, EPK^*_2),..., (ID^*_n, EPK^*_n).
- *Guess phase*: $\mathcal{A}_I$ sends out a guess $b' \in \{0, 1\}$. If the guess $b' = b$, $\mathcal{A}_I$ wins this the game. We define the advantage of winning this game as $Adv(\mathcal{A}_I) = |\Pr[b' = b] - 1/2|$.

In the following, we estimate the advantage $Adv(\mathcal{A}_I)$ which is divided into two situations, namely $Adv_{\mathcal{A}_I-wl}$ and $Adv_{\mathcal{A}_I-l}$. Here, the advantage $Adv_{\mathcal{A}_I-wl}$ states that $\mathcal{A}_I$ wins the security game without issuing the *Certificate generation leak query* or *Decryption* (*Decapsulation*) *leak query*, while the advantage $Adv_{\mathcal{A}_I-l}$ states that $\mathcal{A}_I$ wins the security game with issuing the *Certificate generation leak query* and *Decryption* (*Decapsulation*) *leak query*.

- The advantage $Adv_{\mathcal{A}_I-wl}$: When one event *EventA* that simulates a collision in L_1 or L_2 occurs and the other event *EventB* that simulates a guess $b' = b$ in *Guess phase* occurs, the adversary $\mathcal{A}_I$ wins the security game.
  - *EventA*: We fist consider the collision in the list L_1. Assume that k is the number of variables in the list L_1. Then, k variables, namely v_1, v_2,..., v_k, are randomly selected. We can define $\mathbb{P}G_{1,d}(v_1, v_2, \ldots, v_k) = \mathbb{P}G_{1,i}, \mathbb{P}G_{1,j}$, where $\mathbb{P}G_{1,i}$ and $\mathbb{P}G_{1,j}$ are two different polynomials in the list L_1. Obviously, we can say that the collision happened if $\mathbb{P}G_{1,d}(v_1, v_2, \ldots, v_k) = 0$. According to Lemma 2, we obtain that the probability of $\mathbb{P}G_{1,d}(v_1, v_2, \ldots, v_k) = 0$ is at most $3/q$ since the highest degree of all polynomials in the list L_1 is 3. In addition, we must consider the case of picking out two elements from the list L_1. Hence, we obtain that the probability of collision in the list L_1 is $(3/q)\binom{|L_1|}{2}$. Similarly, the probability of collision in

the list L_2 is $(6/q)\binom{|L_1|}{2}$. Since $|L_1| + |L_2| \leqq 3p$, we have

$$\begin{aligned} Pr[EventA] &\leq q(3/q)\binom{|L_1|}{2} + (6/q)\binom{|L_2|}{2} \\ &\leq q(6/q)(|L_1| + |L_2|)^2 \\ &\leq q54p^2/q. \end{aligned}$$

- *EventB*: Since the guess $b' = b$, we have Pr[*EventB*] $\leqq$ 1/2.

By *EventA* and *EventB*, the advantage $Adv_{\mathcal{A}_I-wl}$ can be presented as below. $Adv_{\mathcal{A}_I-wl}$ = |Pr[*EventA*] + Pr[*EventB*] - 1/2| $\leqq$ |$54p^2/q$ + 1/2 - 1/2| = $54p^2/q$ = $O(p^2/q)$. Hence, $Adv_{\mathcal{A}_I-wl}$ is negligible if the situation of $p = poly(\log q)$ occurs.

- The advantage $Adv_{\mathcal{A}_I-l}$: In addition to the above advantage $Adv_{\mathcal{A}_I-wl}$, the adversary $\mathcal{A}_I$ can also add additional advantage $Adv_{\mathcal{A}_I-l}$ by issuing the *Certificate generation leak query* and *Decryption* (*Decapsulation*) *leak query*.
  - When sending out the *Certificate generation leak query* (i, $LF^I_{CG,i}$, $LF^{II}_{CG,i}$), $\mathcal{A}_I$ can receive the two leaked information $\Lambda LF^I_{CG,i} = LF^I_{CG,i}(CSK_{i-1,1})$ and $\Lambda LF^{II}_{CG,i} = LF^{II}_{CG,i}(CSK_{i-1,2})$, where $|LF^I_{CG,i}|$, $|LF^{II}_{CG,i}| \leqq \psi$. Since the CA's secret key CSK can be obtained by computing $CSK_{0,1} \cdot CSK_{0,2} = CSK_{1,1} \cdot CSK_{1,2} = \ldots = CSK_{i-1,1} \cdot CSK_{i-1,2}$, the leaked information we gain from $LF^I_{CG,i}(CSK_{i-1,1})$ and $LF^{II}_{CG,i}(CSK_{i-1,2})$ is independent of the leaked we gain from $LF^I_{CG,i}(CSK_{i,1})$ and $LF^I_{CG,i}(CSK_{i,2})$ due to the techniques about multiplicative blinding and key update. Hence, $\mathcal{A}_I$ can gain at most 2ψ bits of CSK.
  - When sending out the *Decryption* (*Decapsulation*) *leak query* (ID, k, $LF^I_{D,k}$, $LF^{II}_{D,k}$), $\mathcal{A}_I$ can receive the two leaked information $\Lambda LF^I_{D,k} = LF^I_{D,k}(EC_{k-1,1})$ and $\Lambda LF^{II}_{D,k} = LF^{II}_{D,k}(EC_{k-1,2})$, where $|LF^I_{D,k}|$, $|LF^{II}_{D,k}| \leqq \psi$. Since the entity's certificate EC can be obtained by computing $EC_{0,1} \cdot EC_{0,2} = EC_{1,1} \cdot EC_{1,2} = \ldots = EC_{k-1,1} \cdot EC_{k-1,2}$, the leaked information we gain from $LF^I_{D,k}(EC_{k-1,1})$ and $LF^{II}_{D,k}(EC_{k-1,2})$ is independent of the leaked we gain from $LF^I_{D,k}(EC_{k,1})$ and $LF^{II}_{D,k}(EC_{k,2})$ due to the techniques about multiplicative blinding and key update. Hence, $\mathcal{A}_I$ can gain at most 2ψ bits of EC.

In the following, we analysis $\mathcal{A}_I$'s advantage of winning the security game $G^{IND-CCA}_{LR-AMR-CB-KE}$. The target ciphertext CT^* can be responded correctly by the adversary with the CA's secret key CSK, the target entity's certificate EC or a correct guess. Hence, we have to discuss the following three events.

(1) Event E_{CSK}: The CA's secret key CSK can be gained by $\mathcal{A}_I$ from $\Lambda LF^I_{CG,i}$ and $\Lambda LF^{II}_{D,k}$. Meanwhile, we denote $\overline{E_{CSK}}$ as the complement event of E_{CSK}.

(2) Event E_{EC}: The target entity's certificate EC can be gained by $\mathcal{A}_I$ from $\Lambda LF^I_{D,k}$ and $\Lambda LF^{II}_{D,k}$. Meanwhile, we denote $\overline{E_{EC}}$ as the complement event of E_{EC}.

(3) Event E_{CG}: A correct guess can be outputted by $\mathcal{A}_I$.

According to the above events, we have $\mathcal{A}_I$'s probability Pr[$\mathcal{A}_I$] of winning the security game as below.

$$\begin{aligned} Pr[\mathcal{A}_I] &= Pr[E_{CG}] \\ &= Pr[E_{CG} \wedge (E_{CSK} \vee E_{EC})] \\ &\quad + Pr[E_{CG} \wedge (\overline{E_{CSK}} \wedge \overline{E_{EC}})] \\ &\leq qPr[E_{CSK} \vee E_{EC}] \\ &\quad + Pr[E_{CG} \wedge (\overline{E_{CSK}} \wedge \overline{E_{EC}})]. \end{aligned}$$

Next, we consider the probability Pr[$E_{CG} \wedge (\overline{E_{CSK}} \wedge \overline{E_{EC}})$]. The event $(\overline{E_{CSK}} \wedge \overline{E_{EC}})$ states that $\mathcal{A}_I$ cannot obtain useful information to output the correct response. Thus, Pr[$E_{CG} \wedge (\overline{E_{CSK}} \wedge \overline{E_{EC}})$] is 1/2 due to Pr[$E_{CG}$] = 1/2. We then have Pr[$\mathcal{A}_I$] $\leq$ Pr[$E_{CSK} \vee E_{EC}$] + 1/2 and $Adv_{\mathcal{A}_I-l}$ = |Pr[$\mathcal{A}_I$] - 1/2| $\leq$ Pr[$E_{CSK} \vee E_{EC}$]. At most 2ψ bits of CSK and EC can be respectively gained by $\mathcal{A}_I$ from *Certificate generation leak query* and *Decryption* (*Decapsulation*) *leak query*. With the earlier result, namely $Adv_{\mathcal{A}_I-wl} \leq O(p^2/q)$, we have $Adv_{\mathcal{A}_I-l} \leq Adv_{\mathcal{A}_I-wl} \cdot 2^{2\psi} \leq O((p^2/q) \cdot 2^{2\psi})$. Based on Lemma 2, $Adv_{\mathcal{A}_I-l}$ is negligible if the situation of $\psi < (1-\epsilon)\log q$ occurs.

Theorem 2: Based on the DL and HF assumptions, the proposed LR-AMR-CB-KE scheme is secure under the GBG model against the adversary $\mathcal{A}_{II}$ for the indistinguishability of two ciphertexts against chosen ciphertext attacks (IND-CCA) in the security game $G^{IND-CCA}_{LR-AMR-CB-KE}$.

Proof. Assume that there exists an adversary $\mathcal{A}_{II}$ who plays the following security game $G^{IND-CCA}_{LR-AMR-CB-KE}$ with a challenger $\mathcal{C}$.

- *Setup phase*: The content of this phase is the same as that of *Setup phase* in Theorem 1. Furthermore, the adversary is $\mathcal{A}_{II}$ who can gain the CA's initial secret key ($CSK_{0,1}$, $CSK_{0,2}$).
- *Query phase*: The content of this phase is the same as that of *Query phase* in Theorem 1. Since the adversary is $\mathcal{A}_{II}$ who holds the CA's initial secret key ($CSK_{0,1}$, $CSK_{0,2}$), any entity's certificate EC can be computed by herself/himself. However, $\mathcal{A}_{II}$ is unable gain an entity's secret key ESK. However, the two leaked information $\Lambda LF^I_{D,k} = LF^I_{D,k}(EC_{k-1,1})$ and $\Lambda LF^{II}_{D,k} = LF^{II}_{D,k}(EC_{k-1,2})$ can be gained by $\mathcal{A}_{II}$ in *Decryption* (*Decapsulation*) *leak query*.
- *Challenge phase*: $\mathcal{A}_{II}$ sends out two target plaintexts msg_0, msg_1 and n identities of entities, namely ID^*_1, ID^*_2,..., ID^*_n. Since the adversary is $\mathcal{A}_{II}$, both the *Public key replace query* (ID^*_i, ($\mathbb{B}rEPK^{1st}$, $\mathbb{B}rEPK^{2nd}$)) and *Key generation query* (ID^*_i), for $i = 1, 2, \ldots, n$, are not allowed to be asked. The challenger $\mathcal{C}$ picks a random bit $b \in \{0, 1\}$, and returns the target ciphertext CT^* by

running the *Multiencryption* algorithm with msg_b, (ID_1^*, EPK_1^*), (ID_2^*, EPK_2^*),. . ., (ID_n^*, EPK_n^*).
- *Guess phase*: $\mathcal{A}_{II}$ sends out a guess $b' \in \{0, 1\}$. If the guess $b' = b$, $\mathcal{A}_{II}$ wins this the game. We define the advantage of winning this game as $Adv(\mathcal{A}_{II}) = |\Pr[b' = b] - 1/2|$.

In the following, we estimate the advantage $Adv(\mathcal{A}_{II})$ which is divided into two situations, namely $Adv_{\mathcal{A}_{II}-wl}$ and $Adv_{\mathcal{A}_{II}-l}$. Here, the advantage $Adv_{\mathcal{A}_{II}-wl}$ states that $\mathcal{A}_{II}$ wins the security game without issuing *Decryption* (*Decapsulation*) *leak query*, while the advantage $Adv_{\mathcal{A}_{II}-l}$ states that $\mathcal{A}_{II}$ wins the security game with issuing *Decryption* (*Decapsulation*) *leak query*. Due to the similar inference of Theorem 1, we have $Adv_{\mathcal{A}_{II}-wl} = O(p^2/q)$. The other advantage $Adv_{\mathcal{A}_{II}-l}$ is discussed as below.

- The advantage $Adv_{\mathcal{A}_{II}-l}$: In addition to the above advantage $Adv_{\mathcal{A}_{II}-wl}$, the adversary $\mathcal{A}_{II}$ can also add additional advantage $Adv_{\mathcal{A}_{II}-l}$ by issuing the *Decryption* (*Decapsulation*) *leak query*. When sending out the *Decryption* (*Decapsulation*) *leak query* (ID, k, $LF^I_{D,k}$, $LF^{II}_{D,k}$), $\mathcal{A}_{II}$ can receive the two leaked information $\Lambda LF^I_{D,k} = LF^I_{D,k}(ESK_{k-1,1})$ and $\Lambda LF^{II}_{D,k} = LF^{II}_{D,k}(ESK_{k-1,2})$, where $|LF^I_{D,k}|, |LF^{II}_{D,k}| \leqq \psi$. Since the entity's secret key ESK can be obtained by computing $ESK_{0,1} \cdot ESK_{0,2} = ESK_{1,1} \cdot ESK_{1,2} = \ldots = ESK_{k-1,1} \cdot ESK_{k-1,2}$, the leaked information we gain from $LF^I_{D,k}(ESK_{k-1,1})$ and $LF^{II}_{D,k}(ESK_{k-1,2})$ is independent of the leaked we gain from $LF^I_{D,k}(ESK_{k,1})$ and $LF^{II}_{D,k}(ESK_{k,2})$ due to the techniques about multiplicative blinding and key update. Hence, $\mathcal{A}_{II}$ can gain at most 2ψ bits of ESK.

In the following, we analysis $\mathcal{A}_{II}$'s advantage of winning the security game $G^{IND-CCA}_{LR-AMR-CB-KE}$. The target ciphertext CT^* can be responded correctly by the adversary with the target entity's secret key ESK or a correct guess. Hence, we have to discuss the following two events.

(1) Event E_{ESK}: The target entity's secret key ESK can be gained by $\mathcal{A}_{II}$ from $\Lambda LF^I_{D,k}$ and $\Lambda LF^{II}_{D,k}$. Meanwhile, we denote $\overline{E_{ESK}}$ as the complement event of E_{ESK}.
(2) Event E_{CG}: A correct guess can be outputted by $\mathcal{A}_{II}$.

According to the above events, we have $\mathcal{A}_{II}$'s probability $\Pr[\mathcal{A}_{II}]$ of winning the security game as below.

$$\begin{aligned} Pr[\mathcal{A}_{II}] &= Pr[E_{CG}] \\ &= Pr[E_{CG} \wedge E_{ESK}] + Pr[E_{CG} \wedge \overline{E_{ESK}}] \\ &\leq qPr[E_{ESK}] + Pr[E_{CG} \wedge \overline{E_{ESK}}]. \end{aligned}$$

Next, we consider the probability $\Pr[E_{CG} \wedge \overline{E_{ESK}}]$. The event $\overline{E_{ESK}}$ states that $\mathcal{A}_{II}$ cannot obtain useful information to output the correct response. Thus, $\Pr[E_{CG} \wedge \overline{E_{ESK}}]$ is $1/2$ due to $\Pr[E_{CG}] = 1/2$. We then have $\Pr[\mathcal{A}_{II}] \leqq \Pr[E_{ESK}] + 1/2$ and $Adv_{\mathcal{A}_{II}-l} = |\Pr[\mathcal{A}_{II}] - 1/2| \leq \Pr[E_{ESK}]$. At most 2ψ bits of ESK can be gained by $\mathcal{A}_{II}$ from the *Decryption* (*Decapsulation*) *leak query*. With the earlier result, namely $Adv_{\mathcal{A}_{II}-wl} \leq O(p^2/q)$, we have $Adv_{\mathcal{A}_{II}-l} \leq Adv_{\mathcal{A}_{II}-wl} \cdot 2^{2\psi} \leq O((p^2/q) \cdot 2^{2\psi})$. Based on Lemma 2, $Adv_{\mathcal{A}_{II}} - l$ is negligible if the situation of $\psi < (1 - \epsilon)\log q$ occurs.

Theorem 3: Based on the DL and HF assumptions, the proposed LR-AMR-CB-KE scheme is secure under the GBG model against the adversary $\mathcal{A}_I$ for the anonymous indistinguishability of two identities against chosen ciphertext attacks (ANON-IND-CCA) in the security game $G^{ANON-IND-CCA}_{LR-AMR-CB-KE}$.

Proof. Assume that there exists an adversary $\mathcal{A}_I$ who plays the following security game $G^{ANON-IND-CCA}_{LR-AMR-CB-KE}$ with a challenger $\mathcal{C}$.

- *Setup phase*: The content of this phase is the same as that of *Setup phase* in Theorem 1.
- *Query phase*: The content of this phase is the same as that of *Query phase* in Theorem 1.
- *Challenge phase*: $\mathcal{A}_I$ sends out a target plaintexts msg and n identities of entities, namely ID_1^*, ID_2^*,. . ., ID_n^*. Since the adversary is $\mathcal{A}_I$, the *Certificate generation query* with ID_1^* and ID_2^* is not allowed to be asked. The challenger $\mathcal{C}$ picks a random bit $b \in \{0, 1\}$, and returns the target ciphertext CT^* by running the *Multiencryption* algorithm with msg, (ID_b^*, EPK_1^*), (ID_3^*, EPK_3^*), (ID_4^*, EPK_4^*),. . ., (ID_n^*, EPK_n^*).
- *Guess phase*: $\mathcal{A}_I$ sends out a guess $b' \in \{0, 1\}$. If the guess $b' = b$, $\mathcal{A}_I$ wins this the game. We define the advantage of winning this game as $Adv(\mathcal{A}_I) = |\Pr[b' = b] - 1/2|$.

Due to the similar inference of Theorem 1, we have $Adv_{\mathcal{A}_I-wl} = O(p^2/q)$ and the other advantage $Adv_{\mathcal{A}_I-l} \leq Adv_{\mathcal{A}_I-wl} \cdot 2^{2\psi} \leq O((p^2/q) \cdot 2^{2\psi})$.

Theorem 4: Based on the DL and HF assumptions, the proposed LR-AMR-CB-KE scheme is secure under the GBG model against the adversary $\mathcal{A}_{II}$ for the anonymous indistinguishability of two identities against chosen ciphertext attacks (ANON-IND-CCA) in the security game $G^{ANON-IND-CCA}_{LR-AMR-CB-KE}$.

Proof. Assume that there exists an adversary $\mathcal{A}_{II}$ who plays the following security game $G^{ANON-IND-CCA}_{LR-AMR-CB-KE}$ with a challenger $\mathcal{C}$.

- *Setup phase*: The content of this phase is the same as that of *Setup phase* in Theorem 1. Furthermore, the adversary is $\mathcal{A}_{II}$ who can gain the CA's initial secret key ($CSK_{0,1}$, $CSK_{0,2}$).
- *Query phase*: The content of this phase is the same as that of *Query phase* in Theorem 1. Since the adversary is $\mathcal{A}_{II}$ who holds the CA's initial secret key ($CSK_{0,1}$, $CSK_{0,2}$), any entity's certificate EC can be computed by herself/himself. However, $\mathcal{A}_{II}$ is unable gain an entity's secret key ESK. However, the two leaked information $\Lambda LF^I_{D,k} = LF^I_{D,k}(EC_{k-1,1})$ and $\Lambda LF^{II}_{D,k} =$

TABLE 2. Comparison of our LR-AMR-CB-KE scheme with existing AMR-IBE scheme, AMR-CBE scheme and LR-CBE scheme.

	Tseng and Fan's AMR-IBE scheme [33]	Fan *et al.*'s AMR-CBE scheme [34]	Zhou *et al.*'s LR-CBE scheme [47]	Our LR-AMR-CB-KE scheme
Without the key escrow problem	No	Yes	Yes	Yes
Providing multi-receiver encryption	Yes	Yes	No	Yes
With anonymity	Yes	Yes	No	Yes
Resisting side-channel attacks	No	No	Yes	Yes

$LF^{II}_{D,k}(EC_{k-1,2})$ can be gained by $\mathcal{A}_{II}$ in *Decryption* (*Decapsulation*) *leak query*.

- *Challenge phase*: $\mathcal{A}_{II}$ sends out a target plaintexts *msg* and n identities of entities, namely ID^*_1, ID^*_2,. . ., ID^*_n. Since the adversary is $\mathcal{A}_{II}$, both the Public key replace query (ID^*_i, ($\mathbb{B}rEPK^{1st}$, $\mathbb{B}rEPK^{2nd}$)) and *Key generation query* (ID^*_i), for i = 1 and 2, are not allowed to be asked. The challenger $\mathcal{C}$ picks a random bit $b \in \{0, 1\}$, and returns the target ciphertext CT^* by running the *Multiencryption* algorithm with *msg*, (ID^*_b, EPK^*_1), (ID^*_3, EPK^*_3), (ID^*_4, EPK^*_4),. . ., (ID^*_n, EPK^*_n).
- *Guess phase*: $\mathcal{A}_{II}$ sends out a guess $b' \in \{0, 1\}$. If the guess $b' = b$, $\mathcal{A}_{II}$ wins this the game. We define the advantage of winning this game as $Adv(\mathcal{A}_{II})$ = |Pr[$b' = b$] - 1/2|.

Due to the similar inference of Theorems 1 and 2, we have $Adv_{\mathcal{A}_{II}-wl} = O(p^2/q)$ and the other advantage $Adv_{\mathcal{A}_{II}-l} \leq Adv_{\mathcal{A}_{II}-wl} \cdot 2^{2\psi} \leq O((p^2/q) \cdot 2^{2\psi})$.

VI. COMPARISONS AND PERFORMANCE ANALYSIS

We compare the properties between our LR-AMR-CB-KE scheme, AMR-IBE scheme [33], the AMR-CBE scheme [34] and the LR-CBE scheme [47] in terms of the key escrow problem, multi-receiver encryption, anonymity and side-channel attacks in Table 2. Obviously, Tseng and Fan's AMR-IBE scheme [33] inherit the key escrow problem and does not have the ability to resist side-channel attacks, although they scheme has the function of anonymous multi-receiver for encrypting a message. Although Fan et al.'s AMR-CBE scheme [34] avoids the key escrow problem and has the function of anonymous multi-receiver encryption, the scheme cannot resist side-channel attacks. Zhou et al.'s LR-CBE scheme [47] can resist side-channel attacks and remove the key escrow problem, but the scheme lacks the function of anonymous multi-receiver encryption. Obviously, our LR-AMR-CB-KE scheme not only resists side-channel attacks, but also has the function of anonymous multi-receiver encryption and avoids the key escrow problem.

Now, we present two symbols that can be utilized to assess the computational burden of *Multiencryption* and *Decryption* algorithms of the LR-AMR-CB-KE scheme.

- CB_{pair}: the computational burden of a bilinear pairing $\hat{e} : G_1 \times G_1 \rightarrow G_2$.
- CB_{exp}: the computational burden of an exponentiation in G_1 or G_2.

TABLE 3. Computational burden of a bilinear pairing and an exponentiation.

Operations	CB_{pair}	CB_{exp}
Computational burden	7.83 ms	0.47 ms

TABLE 4. The computational burden for *n* receivers.

Our LR-AMR-CB-KE	*Multiencryption*	*Decryption*
Computational burden	$n \cdot CB_{pair} + (3n + 1) \cdot CB_{exp}$	$4 \cdot CB_{pair} + 4 \cdot CB_{exp}$
n = 1	9.71 ms	33.20 ms
n = 5	46.67 ms	33.20 ms
n = 50	462.47 ms	33.20 ms
n = 100	924.47 ms	33.20 ms
n = 150	1,386.47 ms	33.20 ms

Table 3 from [51] displays the outcomes of a simulation that determined CB_{pair} and CB_{exp} to be 7.83 ms and 0.47 ms, respectively. The experiment was conducted using an Intel Core i7-8550U CPU 1.80 GHz processor and input parameters of finite field F_p, G_1, and G_2. The values of p, G_1, and G_2 were chosen such that p is a prime number with 256 bits, and G_1 and G_2 are groups with a prime order of 224 bits over the finite field F_p.

Moreover, with reference to Table 3, we examine the computational burden of the *Multiencryption* and *Decryption* algorithms in the LR-AMR-CB-KE scheme for different numbers of receivers. Table 4 illustrates the computational burden for n receivers, where n takes values of 1, 5, 50, 100, and 150.

VII. CONCLUSION

In this paper, we proposed the *first* LR-AMR-CB-KE scheme. We defined the framework of LR-AMR-CB-KE, and employed the concept of the leakage resilient properties to present new security models of LR-AMR-CB-KE. Based on the discrete logarithm and hash function assumptions, the proposed scheme was formally proven to be secure for the IND-CCA and ANON-IND-CCA. As compared with the previous AMR-IBE, AMR-CBE and LR-CBE schemes, our LR-AMR-CB-KE scheme removing the key escrow problem not only provides an anonymous multi-receiver mechanism, but also has the abilities to resist side-channel attacks.

REFERENCES

[1] X. Chen, F. Wang, Z. Huang, and Y.-C. Chen, "Smart grid aggregation billing scheme based on blockchain," *J. Netw. Intell.*, vol. 7, no. 4, pp. 878–893, Nov. 2022.

[2] Z. Wang, C. Zhang, and X. Mu, "Decentralized solution for cold chain logistics combining IoT and blockchain technology," *J. Netw. Intell.*, vol. 8, no. 1, pp. 47–61, Feb. 2023.

[3] J. Chen, H. Xiao, M. Hu, and C.-M. Chen, "A blockchain-based signature exchange protocol for metaverse," *Future Gener. Comput. Syst.*, vol. 142, pp. 237–247, May 2023.

[4] C.-M. Chen, S. Liu, X. Li, S. H. Islam, and A. K. Das, "A provably-secure authenticated key agreement protocol for remote patient monitoring IoMT," *J. Syst. Archit.*, vol. 136, Mar. 2023, Art. no. 102831.

[5] S. Khasim and S. S. Basha, "An improved fast and secure CAMEL based authenticated key in smart health care system," *Cloud Comput. Data Sci.*, vol. 3, no. 2, pp. 77–91, May 2022.

[6] T. Saito, K. Xagawa, and T. Yamakawa, "Tightly-secure key-encapsulation mechanism in the quantum random Oracle model," in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 10822. Berlin, Germany: Springer, 2018, pp. 520–551.

[7] J. Wu, Y. Tseng, S. Huang, and T. Tsai, "Leakage-resilient certificate-based key encapsulation scheme resistant to continual leakage," *IEEE Open J. Comput. Soc.*, vol. 1, pp. 131–144, 2020.

[8] S. H. Park, S. Kim, D. H. Lee, and J. H. Park, "Improved ring LWR-based key encapsulation mechanism using cyclotomic trinomials," *IEEE Access*, vol. 8, pp. 112585–112597, 2020.

[9] D. Hofheinz and T. Jager, "Tightly secure signatures and public-key encryption," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 7417. Berlin, Germany: Springer, 2012, pp. 590–607.

[10] R. Chen, Y. Mu, G. Yang, F. Guo, and X. Wang, "Dual-server public-key encryption with keyword search for secure cloud storage," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 4, pp. 789–798, Apr. 2016.

[11] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 2139. Berlin, Germany: Springer, 2001, pp. 213–229.

[12] H. Deng, Z. Qin, Q. Wu, Z. Guan, R. H. Deng, Y. Wang, and Y. Zhou, "Identity-based encryption transformation for flexible sharing of encrypted data in public cloud," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3168–3180, 2020.

[13] C. Gentry, "Certificate-based encryption and the certificate revocation problem," in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer), vol. 2656. Berlin, Germany: Springer, 2003, pp. 272–293.

[14] Q. Yu, J. Li, Y. Zhang, W. Wu, X. Huang, and Y. Xiang, "Certificate-based encryption resilient to key leakage," *J. Syst. Softw.*, vol. 116, pp. 101–112, Jun. 2016.

[15] S. A. Chaudhry, H. Alhakami, A. Baz, and F. Al-Turjman, "Securing demand response management: A certificate-based access control in smart grid edge computing infrastructure," *IEEE Access*, vol. 8, pp. 101235–101243, 2020.

[16] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 196. Heidelberg, Germany: Springer-Verlag, 1984, pp. 47–53.

[17] D. Galindo, P. Morillo, and C. Rafols, "Improved certificate-based encryption in the standard model," *J. Syst. Softw.*, vol. 81, no. 7, pp. 1218–1226, Jul. 2008.

[18] Y. Lu and J. Li, "Efficient certificate-based encryption scheme secure against key replacement attacks in the standard model," *J. Inf. Sci. Eng.*, vol. 30, no. 5, pp. 1553–1568, Sep. 2014.

[19] J. Li, Z. Wang, and Y. Zhang, "Provably secure certificate-based signature scheme without pairings," *Inf. Sci.*, vol. 233, pp. 313–320, Jun. 2013.

[20] Y.-H. Hung, S.-S. Huang, and Y. Tseng, "A short certificate-based signature scheme with provable security," *Inf. Technol. Control*, vol. 45, no. 3, pp. 243–253, Sep. 2016.

[21] K. Kurosawa, "Multi-recepient public-key encryption with shortened ciphertext," in *Public Key Cryptography* (Lecture Notes in Computer Science), vol. 2274. Berlin, Germany: Springer, 2002, pp. 48–63.

[22] M. Bellare, A. Boldyreva, and D. Pointcheval, "Multi-recepient encryption schemes: Security notions and randomness re-use," in *Public Key Cryptography* (Lecture Notes in Computer Science), vol. 2567. Berlin, Germany: Springer, 2003, pp. 85–99.

[23] J. Baek, R. Safavi-Naini, and W. Susilo, "Efficient multi-receiver identity-based encryption and its application to broadcast encryption," in *Public Key Cryptography* (Lecture Notes in Computer Science), vol. 3396. Berlin, Germany: Springer, 2005, pp. 380–397.

[24] S. Chatterjee and P. Sarkar, "Multi-receiver identity-based key encapsulation with shortened ciphertext," in *Progress in Cryptology—INDOCRYPT* (Lecture Notes in Computer Science), vol. 4329. Berlin, Germany: Springer, 2006, pp. 394–408.

[25] L. Lu and L. Hu, "Pairing-based multi-recipient public key encryption," in *Proc. Int. Conf. Secur. Manag.*, 2006, pp. 159–165.

[26] J.-H. Park, K.-T. Kim, and D.-H. Lee, "Cryptanalysis and improvement of a multi-receiver identity-based key encapsulation at Indocrypt'06," in *Proc. ASIACCS*, 2008, pp. 373–380.

[27] C.-I. Fan, L.-Y. Huang, and P.-H. Ho, "Anonymous multireceiver identity-based encryption," *IEEE Trans. Comput.*, vol. 59, no. 9, pp. 1239–1249, Sep. 2010.

[28] H. Wang, Y. Zhang, H. Xiong, and B. Qin, "Cryptanalysis and improvements of an anonymous multi-receiver identity-based encryption scheme," *IET Inf. Secur.*, vol. 6, no. 1, pp. 20–27, Mar. 2012.

[29] H. Chien, "Improved anonymous multi-receiver identity-based encryption," *Comput. J.*, vol. 55, no. 4, pp. 439–446, Apr. 2012.

[30] J.-H. Zhang and Y.-B. Cui, "Comment an anonymous multi-receiver identity-based encryption scheme," Cryptol. ePrint Arch., IACR, Pittsburgh, PA, USA, Rep. 201, 2012.

[31] H. Wang, "Insecurity of 'improved anonymous multi-receiver identity-based encryption,'" *Comput. J.*, vol. 57, no. 4, pp. 636–638, Apr. 2014.

[32] Y.-M. Tseng, Y.-H. Huang, and H.-J. Chang, "Privacy-preserving multireceiver ID-based encryption with provable security," *Int. J. Commun. Syst.*, vol. 27, no. 7, pp. 1034–1050, Jul. 2014.

[33] Y.-F. Tseng and C.-I. Fan, "Anonymous multireceiver identity-based encryption against chosen-ciphertext attacks with tight reduction in the standard model," *Secur. Commun. Netw.*, vol. 2021, pp. 1–11, Jun. 2021.

[34] C.-I. Fan, P.-J. Tsai, J.-J. Huang, and W.-T. Chen, "Anonymous multi-receiver certificate-based encryption," in *Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discovery*, Oct. 2013, pp. 19–26.

[35] T. Kubota, K. Yoshida, M. Shiozaki, and T. Fujino, "Deep learning side-channel attack against hardware implementations of AES," *Microprocess. Microsyst.*, vol. 87, Nov. 2021, Art. no. 103383.

[36] K. Ngo, E. Dubrova, Q. Guo, and T. Johansson, "A side-channel attack on a masked IND-CCA secure saber KEM implementation," *IACR Trans. Cryptograph. Hardw. Embedded Syst.*, vol. 2021, pp. 676–707, Aug. 2021.

[37] A. Akavia, S. Goldwasser, and V. Vaikuntanathan, "Simultaneous hardcore bits and cryptography against memory attacks," in *Proc. Theory Cryptogr. Conf.* (Lecture Notes in Computer Science), vol. 5444. Berlin, Germany: Springer, 2009, pp. 474–495.

[38] M. Naor and G. Segev, "Public-key cryptosystems resilient to key leakage," *SIAM J. Comput.*, vol. 41, no. 4, pp. 772–814, Jan. 2012.

[39] S. Li, F. Zhang, Y. Sun, and L. Shen, "Efficient leakage-resilient public key encryption from DDH assumption," *Cluster Comput.*, vol. 16, no. 4, pp. 797–806, Dec. 2013.

[40] S. Liu, J. Weng, and Y. Zhao, "Efficient public key cryptosystem resilient to key leakage chosen ciphertext attacks," in *CTRSA* (Lecture Notes in Computer Science), vol. 7779. Berlin, Germany: Springer, 2013, pp. 84–100.

[41] E. Kiltz and K. Pietrzak, "Leakage resilient Elgamal encryption," in *Advances in Cryptology—ASIACRYPT*, (Lecture Notes in Computer Science), vol. 6477. Berlin, Germany: Springer, 2010, pp. 595–612.

[42] D. Galindo, J. Großschädl, Z. Liu, P. K. Vadnala, and S. Vivek, "Implementation of a leakage-resilient ElGamal key encapsulation mechanism," *J. Cryptograph. Eng.*, vol. 6, no. 3, pp. 229–238, Feb. 2016.

[43] Z. Brakerski, Y. T. Kalai, J. Katz, and V. Vaikuntanathan, "Cryptography resilient to continual memory leakage," in *Proc. 51st Annu. IEEE Symp. Found. Comput. Sci.*, Feb. 2010, pp. 501–510.

[44] T.-H. Yuen, S. S. M. Chow, Y. Zhang, and S.-M. Yiu, "Identity-based encryption resilient to continual auxiliary leakage," in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 7237. Berlin, Germany: Springer, 2012, pp. 117–134.

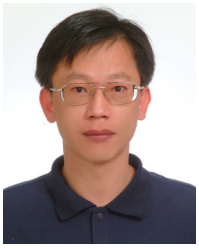
[45] Y. Guo, J. Li, Y. Lu, Y. Zhang, and F. Zhang, "Provably secure certificate-based encryption with leakage resilience," *Theor. Comput. Sci.*, vol. 711, pp. 1–10, Feb. 2018.

[46] J. Li, Y. Guo, Q. Yu, Y. Lu, Y. Zhang, and F. Zhang, "Continuous leakage-resilient certificate-based encryption," *Inf. Sci.*, vols. 355–356, pp. 1–14, Aug. 2016.

[47] Y. Zhou, B. Yang, T. Wang, Z. Xia, and H. Hou, "Continuous leakage-resilient certificate-based encryption scheme without bilinear pairings," *Comput. J.*, vol. 63, no. 1, pp. 508–524, Jan. 2020.

[48] D. Boneh, X. Boyen, and E. J. Goh, "Hierarchical identity-based encryption with constant size ciphertext," in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 3494. Berlin, Germany: Springer, 2005, pp. 440–456.
[49] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM J. Comput.*, vol. 38, no. 1, pp. 97–139, Jan. 2008.
[50] D. Galindo and S. Virek, "A practical leakage-resilient signature scheme in the generic group model," in *Selected Areas in Cryptography* (Lecture Notes in Computer Science), vol. 7707. Berlin, Germany: Springer, 2013, pp. 50–65.
[51] Y. Li, Q. Cheng, X. Liu, and X. Li, "A secure anonymous identity-based scheme in new authentication architecture for mobile edge computing," *IEEE Syst. J.*, vol. 15, no. 1, pp. 935–946, Mar. 2021.

TUNG-TSO TSAI received the Ph.D. degree from the Department of Mathematics, National Changhua University of Education, Taiwan, in 2014, under the supervision of Prof. Yuh-Min Tseng. He is currently an Assistant Professor with the Department of Computer Science and Engineering, National Taiwan Ocean University, Taiwan. His research interests include applied cryptography, pairing-based cryptography, and leakage-resilient cryptography.



YUH-MIN TSENG (Member, IEEE) is currently the Vice President and a Professor with the Department of Mathematics, National Changhua University of Education, Taiwan. He has published over 100 scientific journal articles on various research areas of cryptography, security, and computer networks. His research interests include cryptography, network security, computer networks, and leakage-resilient cryptography. He is a member of the IEEE Computer Society, the IEEE Communications Society, and the Chinese Cryptology and Information Security Association (CCISA). He serves as an editor for several international journals.

SEN-SHAN HUANG received the Ph.D. degree from the University of Illinois at Urbana–Champaign, in 1997, under the supervision of Prof. Bruce C. Berndt. He is currently a Professor with the Department of Mathematics, National Changhua University of Education, Taiwan. His research interests include number theory, cryptography, and leakage-resilient cryptography.