IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 6, 2019, accepted January 27, 2019, date of publication January 31, 2019, date of current version February 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2896773

Leakage-Resilient Certificate-Based Signature Resistant to Side-Channel Attacks

JUI-DI WU[1], YUH-MIN TSENG[1], SEN-SHAN HUANG[1], AND TUNG-TSO TSAI[2]
[1]Department of Mathematics, National Changhua University of Education, Changhua 500, Taiwan
[2]Department of Research, Foxconn, Taipei 114, 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 MOST106-2221-E-018-007-MY2.

ABSTRACT Certificate-based cryptography is an attractive public-key setting, and it not only simplifies certificate management in the traditional public-key cryptography but also eliminates the key escrow problem inherent in the identity-based cryptography. Recently, leakage-resilient cryptography resistant to side-channel attacks has received significant attention from cryptographic researchers. By side-channel attacks, adversaries could obtain partial information of secret and private keys involved in cryptographic algorithms by perceiving execution time or energy consumptions of each algorithm invocation. The certificate-based signature (CBS) is a class of important public-key signature. Up to date, there exists no leakage-resilient CBS (LR-CBS) scheme resistant to side-channel attacks. In this paper, the *first* LR-CBS scheme is proposed and it possesses overall unbounded leakage property, namely, it permits adversaries to continuously obtain partial information of secret or private keys involved in the associated algorithm invocations. The security analysis is given to prove that the proposed LR-CBS scheme is existential unforgeability against adaptive chosen-message attacks for adversaries in the generic bilinear group model.

INDEX TERMS Side-channel attacks, leakage resilience, certificate-based signature, generic bilinear, group model.

I. INTRODUCTION

In the traditional public-key cryptography [1], [2], a user usually selects her/his secret key and then computes the corresponding public key. Hence, each user requires a certificate to provide a trusted binding between the user's public key and identity information. Meanwhile, a public-key infrastructure (PKI) has to be created to manage certificates of all users. The concept of identity (ID)-based cryptography [3], [4] was introduced to remove certificate management. In an ID-based public-key setting, a user's identity is viewed as the user's public key so that no certificate is required.

In addition, a private key generator (PKG) with a system secret key is responsible to generate the user's private keys according to the user's identity information. In such a case, it incurs the key escrow problem. It means that the PKG may decrypt any cipher-texts sent to arbitrary user, and sign any messages on behalf of arbitrary user. In 2003, the concept of certificateless cryptography [5] was presented to resolve the key escrow problem. In a certificateless public-key setting, a user's private key consists of two components, namely, one is a secret key chosen by the user himself/herself and the other is a partial private key generated by a trusted key generation center (KGC). Since the KGC does not know the user's secret key, the key escrow problem is resolved. It is worth mentioning that the certificateless public-key setting do not require certificate to validate the user's public key so that it must provide additional mechanisms to revoke misbehaving users [6], [7].

The notion of certificate-based cryptography was introduced by Gentry [8] to simplify certificate management in the traditional public-key cryptography and eliminate the key escrow problem inherent in the ID-based cryptography. As compared with the certificateless cryptography, the certificate-based cryptography does not require additional revocation mechanisms. In a certificate-based public-key setting, a user first sets her/his secret/public key pair while sending the public key to a trusted certificate authority (CA). By the user's public key, identity information and validity period, the CA generates the user's associated certificate, where the certificate is viewed as a part of the user's private key. Hence, for certificate-based signature (CBS) and

The associate editor coordinating the review of this manuscript and approving it for publication was Sedat Akleylek.

encryption (CBE) schemes, a user must use both her/his up-to-date certificate and secret key to sign a message or decrypt a cipher-text.

The adversary models of those public-key settings mentioned above (namely, traditional, ID- based, certificateless and certificated-based public-key settings) have a nature assumption that secret and private keys involved in cryptographic algorithms must be entirely hidden to adversaries. Recently, a new type of threat, called ''side-channel attack'', endangers the security of cryptographic schemes based on these public key settings. By side-channel attacks, adversaries could obtain partial information of secret and private keys involved in cryptographic algorithms by perceiving execution time or energy consumptions [9]–[12] of each algorithm invocation. Recently, leakage-resilient cryptography resistant to side-channel attacks has received significant attention from cryptographic researchers. Based on various public-key settings, numerous leakage-resilient cryptographic primitives (encryption and signature schemes) have been proposed to address side-channel attacks, such as leakage-resilient encryption schemes [13], [14], leakage-resilient signature schemes [15], [16], leakage-resilient ID-based encryption schemes [17], [18], leakage-resilient ID-based signature schemes [19], [20], leakage-resilient certificateless encryption schemes [21], [22] and leakage-resilient certificateless signature schemes [23].

Based on certificate-based public-key settings, several leakage-resilient certificate-based encryption (LR-CBE) [24]–[26] have been proposed, but there exists no leakage-resilient CBS (LR-CBS) scheme resistant to side-channel attacks. In this paper, we aim at the design of the first LR-CBS scheme with overall unbounded leakage property in the sense that it permits adversaries to continuously obtain partial information of the secret or private keys involved in the associated cryptographic algorithms.

A. RELATED WORK

In the section, we first briefly review the related leakage-resilient signature schemes with overall unbounded leakage property under various kinds of public-key settings. In addition, the related work of the previously proposed CBS schemes is also recalled.

Generally, a cryptographic scheme composes of several phases or algorithms. A leakage-resilient cryptographic scheme is still secure even if partial information of secret and private keys involved in cryptographic algorithms is leaked to adversaries. For leakage information amount during the life time of the cryptographic scheme, there are two kinds of leakage models. One is bounded leakage model [27] in the sense that the total leakage amount musts be bounded to a fixed bit-length. The other is continuous leakage model [19], [22] that allows adversaries to continuously obtain partial information of secret and private keys while the total leakage amount is unbounded. Obviously, the continuous leakage model with overall unbounded leakage property is more practical than the bounded leakage model.

Under traditional public-key settings, two leakage-resilient signature schemes with overall unbounded leakage property were proposed. In 2013, Galindo and Vivek [15] proposed a secure leakage-resilient signature scheme overall unbounded leakage property in the generic bilinear group (GBG) model [28]. To improve performance, Tang *et al*. [16] proposed an improvement on Galindo and Vivek's scheme. In 2016, Wu *et al*. [19] defined an adversary model of leakage-resilient ID-based signature (LR-IBS) schemes under the continual leakage model and proposed the first LR-IBS scheme based on an ID-based public-key setting. Under the continual leakage model, Wu *et al*.'s scheme allows adversaries to leak partial information of the PKG's system secret key in the key extract phase and users' private keys in the signing phase for each algorithm invocation. In 2018, based on a certificateless public-key setting, Wu *et al*. [23] also proposed leakage-resilient certificateless signature (LR-CLS) scheme with overall unbounded leakage property. In the generic bilinear group model, Wu *et al*. formally proved that both LR-IBS scheme and LR-CLS scheme are existential unforgeability against adaptive chosen-message attacks of adversaries.

In the following, we briefly review the related work of certificate-based signature (CBS) scheme. In 2004, Kang *et al*. [29] presented the first CBS scheme based on bilinear pairings [4]. Afterward, Li *et al*. [30] defined a new adversary model (security notion) of CBS schemes and introduced a new attack, the key replacement attack on the certificate-based public-key setting. In addition, Li *et al*. also demonstrated that Kang *et al*.'s scheme suffers from the key replacement attack while presenting an improved CBS scheme. Based on Li *et al*.'s adversary model, Liu *et al*. [31] presented two CBS schemes, namely, a CBS scheme without pair operation in the random oracle model and a CBS scheme under the standard model (without using random oracles). In 2009, Zhang *et al*. [32] demonstrated that Liu *et al*.'s CBS scheme without pair operation was insecure and presented an improvement. Meanwhile, Wu *et al*. [33] also proposed the other improved CBS scheme on Liu *et al*.'s CBS scheme without pair operation in the random oracle model. The signature lengths of these CBS schemes [29]–[33] are at least two group elements. For reducing the signature length, the first short certificate-based signature (SCBS) scheme was proposed by Liu *et al*. [34]. However, Cheng *et al*. [35] demonstrated that Liu *et al*.'s SCBS scheme cannot resist the attacks of Type I adversary under the accredited adversary model [30], [32], [33]. In 2012, Li *et al*. [36] also proposed an improved SCBS scheme. In 2016, Hung *et al*. [37] demonstrated that Li *et al*.'s SCBS scheme is still insecure against Type I adversary and proposed a provably secure and novel SCBS scheme.

The security of these CBS and SCBS schemes mentioned above is under the accredited adversary model [30], [32], [33], [37] which includes two kinds of adversaries, namely, Type I (uncertified entity) and Type II adversary (honest-but-curious CA). In 2016, Liu and Li [38] defined an enhanced

adversary model of the CBS schemes. In the enhanced adversary model, a Type II adversary is changed from an honest-but-curious CA to a malicious-but-passive CA. Under the new adversary model, Liu and Li demonstrated that the previous CBS schemes suffer from malicious-but-passive certificate authority attack, namely, the CA may forge a new signature (ID, m, σ') from an existing signature (ID, m, σ), where ID and m are the same identity and message, respectively. Zhou and Cui [39] also proposed a new CBS scheme under the enhanced adversary model. Nevertheless, the malicious-but-passive CA cannot forge a signature on an arbitrary message m if a signer with identity ID did not generate a signature on the message m. Indeed, the adversary model defined in [30], [32], [33], and [37] is enough to model the abilities of adversaries.

B. CONTRIBUTION AND ORGANIZATION

Up to date, no leakage-resilient CBS (LR-CBS) scheme resistant to side-channel attacks is proposed. In this paper, we first define a new adversary model of LR-CBS schemes resistant to side-channel attacks under the continual leakage model. The adversary model also consists of two types of adversaries, Type I (uncertified entity) and Type II adversary (honest-but-curious CA). Both types of adversaries are extended from the accredited adversary models of CBS and SCBS schemes [30], [32], [33], [37] by adding two extra key leakage queries, namely, the certificate generation leak and signing leak queries. Both adversaries are permitted to continuously obtain partial information of the secret or private keys involved in the associated algorithm invocations.

Under the new adversary model with continual key leakage, the *first* LR-CBS scheme resistant to side-channel attacks is proposed and it possesses overall unbounded leakage property. For achieving overall unbounded leakage property, the proposed LR-CBS scheme adopts the key update technique used in [15], [19], [22], and [23] to refresh the CA's system secret key after (before) running each certificate generation algorithm and a signer's secret key and certificate after (before) running each signing algorithm. It is worth mentioning that the CA's and each signer's public keys are still unchanged. In the key update technique, the CA's system secret key is partitioned into two components while each signer's secret key and certificate are also divided into two components, respectively. Although adversaries may obtain partial information of two corresponding current components in the associated algorithm invocations, it is useless for recovering the original secret keys or certificates. In the generic bilinear group model [28], the security of the proposed LR-CBS scheme is formally proved to be existential unforgeability against adaptive chosen-message attacks of Types I and II adversaries under the new adversary model with continual key leakage.

The rest of the paper is structured as follows. Preliminaries are given in Section 2. In Section 3 demonstrates the framework and adversary model of LR-CBS schemes. A secure LR-CBS scheme resistant to side-channel attacks is presented in Section 4. The security of the proposed LR-CBS scheme is proved in Section 5. Comparisons with the previously proposed CBS and SCBS schemes are given in Section 6. In Section 7, conclusion and future work are discussed.

II. PRELIMINARIES

Here, several preliminaries are presented as follows.

A. ENTROPY

In order to measure the security impact of leakage information of secret values involved in cryptographic algorithms, we introduce the notion of entropy. Entropy is viewed as an estimation of uncertainty for unknown secret values. Let X and Y be two finite discrete random variables. Let $\Pr[X = x]$ and $\Pr[Y = y]$ represent the associated probabilities of $X = x$ and $Y = y$, respectively. The min-entropy of a random variable denotes the estimation of some value with the largest probability. In the following, we define two kinds of min-entropies.

1. Min-entropy of X: $H_\infty(X) = -\log_2(\max_x \Pr[X = x])$.
2. Average conditional min-entropy of X under the event $Y = y$: $\widetilde{H}_\infty(X|Y) = -\log_2(E_{y\leftarrow Y}[\max_x \Pr[X = x| Y = y]])$

Indeed, an unknown secret value may be regarded as a discrete random variable. For discrete random variables with partial leakage information, two consequences are derived as follows.

Lemma 1 [40]: Let $f : X \to \{0, 1\}^\lambda$ represent a leakage function on a secret value X (i.e. a discrete random variable) while its output bit-length is bounded to λ bits. The average conditional min-entropy of X under the event $f(X)$ has $\widetilde{H}_\infty(X|f(X)) \geqq H_\infty(X) - \lambda$.

Lemma 2 [15]: Let $F \in Z_p[X_1, X_2, \ldots, X_n]$ denote a non-zero polynomial of degree at most d while associating a leakage function with the maximal output bit-length λ. Let P_i (for $i = 1, 2, \ldots, n$) be the associated probability distributions on $X_i = x_i$ such that $H_\infty(P_i) \geqq \log p - \lambda$ and $0 \leqq \lambda \leqq \log p$. We have $\Pr[F(x_1, x_2, \ldots, x_n) = 0] \leqq \frac{d}{p} 2^\lambda$ if $x_i \xleftarrow{P_i} Z_p$ (for $i = 1, 2, \ldots, n$) are mutually independent. If $\lambda < \log p - \omega(\log\log p)$, $\Pr[F(x_1, x_2, \ldots, x_n) = 0]$ is negligible.

B. BILINEAR GROUPS

Let $G =< g >$ and G_T represent two multiplicative cyclic groups of a prime order p. A map $\hat{e} : G \times G \to G_T$ is an admissible bilinear pairing map if the following properties hold:

1. *Bilinearity*: for all $u, v \in Z_p^*$, $\hat{e}(g^u, g^v) = \hat{e}(g, g)^{uv}$.
2. *Computability*: for all $u, v \in G$, the operation $\hat{e}(u, v)$ can be computed efficiently.
3. *Non − degeneracy*: $\hat{e}(g, g) \neq 1$ which is regarded as a generator of G_T.

For the detailed settings of bilinear groups, a reader may refer to [4], [6], [41], and [42].

C. GENERIC BILINEAR GROUP MODEL

The generic bilinear group model [28] is regarded as a kind of adversary model that is played by an adversary and a challenger. For performing group operations, the adversary must issue the corresponding group queries (oracles) to the challenger to return the executing results. In the generic bilinear group model, three group queries Q_G, Q_T and Q_p represent the multiplication operation on the group G, the multiplication operation on the group G_T and the bilinear pairing map operation, respectively. In this model, each group element must be represented by a distinct bit string. To do so, two random injective functions $\Psi : Z_p \to \zeta$ and $\Psi_T : Z_p \to \zeta_T$ are employed to map the elements of G and G_T to two sets of bit strings ζ and ζ_T, respectively. $|\zeta|$ and $|\zeta_T|$, respectively, denote the amounts of all elements of ζ and ζ_T while satisfying $|\zeta| = |\zeta_T| = p$ and $\zeta \cap \zeta_T = \phi$. For any $u, v \in Z_p^*$, Q_G, Q_T and Q_p, respectively, have the following properties.

- $Q_G(\Psi(u), \Psi(v)) \to \Psi(u + v \bmod p)$.
- $Q_T(\Psi_T(u), \Psi_T(v)) \to \Psi_T(u + v \bmod p)$.
- $Q_p(\Psi(u), \Omega(v)) \to \Psi_T(uv \bmod p)$.

Note that $\Psi(1)$ represents the generator g of G and $\Psi_T(1)$ denotes the generator $\hat{e}(g, g)$ of G_T. In the generic bilinear group model, if a probabilistic polynomial time (PPT) adversary can efficiently find a collision of the multiplicative group G or G_T, it means that the adversary may solve the *discrete logarithm problem* on G or G_T [28].

III. FRAMEWORK AND ADVERSARY MODEL

In this section, we define a new framework and adversary model of leakage-resilient certificate- based signature (LR-CBS) schemes resistant to side-channel attacks under the continual leakage model.

A. FRAMEWORK OF LR-CBS SCHEME

A LR-CBS scheme composes of two roles, namely, users (signers/verifiers) and a trusted certificate authority (CA). A user with identity ID first sets her/his secret/public key pair (SK_{ID}, PK_{ID}) while sending the public key PK_{ID} to the CA. By the user's PK_{ID}, ID and validity period, the CA uses a system secret key SK_{CA} to generate the user's associated certificate CK_{ID}, where CK_{ID} is viewed as a part of the user's private key. Hence, a user's private key consists of her/his secret key SK_{ID} and up-to-date certificate CK_{ID}.

For achieving the overall unbounded leakage [15], [19], [22], [23] a user's secret key SK_{ID} and certificate CK_{ID} must be divided into two parts and separately stored in the memory. Also, the CA's system secret key SK_{CA} is divided and stored. The point is that the CA's system secret key must be updated after (before) running each certificate generation algorithm and a user's secret key and certificate after (before) running each signing algorithm. The detailed framework of LR-CBS scheme is defined as follows.

Definition 1: A LR-CBS scheme consists of five algorithms as below:

- *Setup:* This algorithm is performed by the CA that takes as input a security parameter τ, and obtains an initial system secret key $SK_{CA} = (SK_{CA,0,1}, SK_{CA,0,2})$ and public parameters PP. The CA publishes PP and keeps $(SK_{CA,0,1}, SK_{CA,0,2})$ in secret.
- *User key generation:* This algorithm is performed by a user that takes as input an identity ID, and obtains the user's initial secret key $SK_{ID} = (SK_{ID,0,1}, SK_{ID,0,2})$ and the first partial public key $PK_{ID,1}$.
- *Certificate generation:* For the i-th *Certificate generation* algorithm invocation, the CA first refreshes the current system secret key $(SK_{CA,i,1}, SK_{CA,i,2})$ using $(SK_{CA,i-1,1}, SK_{CA,i-1,2})$. This algorithm is performed by the CA that takes as input a user's ID and the first partial public key $PK_{ID,1}$, and returns the user's certificate CK_{ID} and the second partial public key $PK_{ID,2}$ to the user. Upon receiving CK_{ID} and $PK_{ID,2}$, the user divides CK_{ID} into an initial certificate $(CK_{ID,0,1}, CK_{ID,0,2})$ and sets her/his public key $PK_{ID} = (PK_{ID,1}, PK_{ID,2})$.
- *Signing:* For the j-th *Signing* algorithm invocation of a user (signer) with identity ID, the signer first refreshes the current secret key $(SK_{ID,j,1}, SK_{ID,j,2})$ using $(SK_{ID,j-1,1}, SK_{ID,j-1,2})$ and the current certificate $(CK_{ID,j,1}, CK_{ID,j,2})$ using $(CK_{ID,j-1,1}, CK_{ID,j-1,2})$. This algorithm is performed by the signer that takes as input a message m, and returns a signature σ.
- *Verifying:* This algorithm is performed by a user (verifier) that takes as input (ID, PK_{ID}, m, σ), and outputs either "accept" or "reject".

B. ADVERSARY MODEL OF LR-CBS SCHEME

By the framework of LR-CBS scheme in the previous subsection, adversaries can obtain partial information of the CA's current system secret key $(SK_{CA,i,1}, SK_{CA,i,2})$ in the i-th *Certificate generation* algorithm invocation while the outputs of two leakage functions $f_{CG,i}$ and $h_{CG,i}$ represent partial information of $(SK_{CA,i,1}, SK_{CA,i,2})$. In the j-th *Signing* algorithm invocation of a user with identity ID, adversaries can obtain partial information of the signer's current secret key $(SK_{ID,j,1}, SK_{ID,j,2})$ and certificate $(CK_{ID,j,1}, CK_{ID,j,2})$ while the outputs of two leakage functions $f_{S,j}$ and $h_{S,j}$ represent partial information of both $(SK_{ID,j,1}, SK_{ID,j,2})$ and $(CK_{ID,j,1}, CK_{ID,j,2})$. The output of each leakage function is bounded to λ bits, namely, $|f_{CG,i}|, |h_{CG,i}|, |f_{S,j}|, |h_{S,j}| \leq \lambda$, where $|\cdot|$ denotes the output bit-length and λ is the leakage parameter. The syntaxes of $f_{CG,i}$, $h_{CG,i}$, $f_{S,j}$ and $h_{S,j}$ are respectively defined as below.

- $\Lambda f_{CG,i} = f_{CG,i}(SK_{CA,i,1}, Rf_{CG,i})$.
- $\Lambda h_{CG,i} = h_{CG,i}(SK_{CA,i,2}, Rh_{CG,i})$.
- $\Lambda f_{S,j} = f_{S,j}(SK_{ID,j,1}, CK_{ID,j,1}, Rf_{S,j})$.
- $\Lambda h_{S,j} = h_{S,j}(SK_{ID,j,2}, CK_{ID,j,2}, Rh_{S,j})$.

Here, $Rf_{CG,i}$, $Rh_{CG,i}$, $Rf_{S,j}$ and $Rh_{S,j}$ denote the random values used in the computation rounds of the associated algorithm invocations.

Based on the accredited adversary model of CBS schemes [30], [32], [33], [37], a new adversary model of LR-CBS scheme is defined here. In this model, during the life time of LR-CBS scheme, adversaries are permitted to continuously get partial information of the CA's system secret key used in each *Certificate generation* algorithm invocation, a signer's secret key and certificate used in the signing phase, and random values involved in both algorithm invocations. The new adversary model consists of two types of adversaries, namely, Type I (uncertified entity) and Type II adversary (honest-but-curious CA).

- Type I adversary (uncertified entity): This adversary is able to obtain the secret key of any entity, but cannot get the certificate of a target entity. Meanwhile, the adversary can get partial information of both the CA's current system secret key in each *Certificate generation* algorithm invocation and a signer's certificate in each *Signing* algorithm invocation.
- Type II adversary (honest-but-curious CA): This adversary possesses the system secret key so that it is able to generate the certificate of any entity, but cannot get the secret key of a target entity. Meanwhile, the adversary can get partial information of a signer's current secret key in each *Signing* algorithm invocation.

In the following, we employ a security game G_{LR-CBS} to represent the new adversary model of LR-CBS scheme under the continual leakage model.

Definition 2 (G_{LR-CBS}): The security game G_{LR-CBS} is played by an adversary A (Types I or II adversaries) and a challenger B. If no PPT adversary A with a non-negligible advantage can win G_{LR-CBS}, we say that the LR-CBS scheme is existential unforgeability against adaptive chosen-message attacks (UF-LR-CBS-ACMA). The security game G_{LR-CBS} consists of three phases as follows.

- *Setup phase*. By taking a security parameter τ as input, the challenger B performs the *Setup* algorithm presented in Definition 1, and obtains an initial system secret key $SK_{CA} = (SK_{CA,0,1}, SK_{CA,0,2})$ and public parameters PP. If A is of Type II adversary, SK_{CA} and PP are sent to A. Otherwise, B sends PP to A and keeps ($SK_{CA,0,1}$, $SK_{CA,0,2}$) in secret.
- *Query phase*. A may issue a number of queries to B adaptively as below:
  - *User key generation query* (ID): By taking a user's ID as input, B generates the associated initial secret key $SK_{ID} = (SK_{ID,0,1}, SK_{ID,0,2})$ and the first partial public key $PK_{ID,1}$.
  - *Secret key query* (ID): By taking a user's ID as input, B returns the user's initial secret key $SK_{ID} = (SK_{ID,0,1}, SK_{ID,0,2})$ to A. Note that if the *Public key replace query* (ID) has been ever issued, this query is forbidden.
  - *Certificate generation query* (ID, $PK_{ID,1}$): By taking a user's ID and the associated first partial public key $PK_{ID,1}$ as input, B responds the user's certificate CK_{ID} and the second partial public key $PK_{ID,2}$ to A.
  - *Certificate generation leak query* (i, $f_{CG,i}$, $h_{CG,i}$): For the i-th *Certificate generation query*, the *Certificate generation leak query* is allowed to be issued only once. By taking two leakage functions $f_{CG,i}$ and $h_{CG,i}$ as input, B generates the leakage information $\Lambda f_{CG,I}$ and $\Lambda h_{CG,i}$ about the CA's current system secret key ($SK_{CA,i,1}$, $SK_{CA,i,2}$), and returns $\Lambda f_{CG,I}$ and $\Lambda h_{CG,i}$ to A.
  - *Public key retrieve query* (ID): By taking a user's ID as input, B returns the associated public key $PK_{ID} = (PK_{ID,1}, PK_{ID,2})$.
  - *Signing query* (ID, m). For the j-th *Signing* algorithm invocation of a user (signer) with identity ID, the signer first refreshes the current secret key ($SK_{ID,j,1}$, $SK_{ID,j,2}$) using ($SK_{ID,j-1,1}$, $SK_{ID,j-1,2}$) and the current certificate ($CK_{ID,j,1}$, $CK_{ID,j,2}$) using ($CK_{ID,j-1,1}$, $CK_{ID,j-1,2}$). By taking a message m as input, B returns a signature σ.
  - *Signing leak query* (ID, j, $f_{S,j}$, $h_{S,j}$): For the j-th *Signing query* of the signer with identity ID, the *Signing leak query* is allowed to be issued only once. By taking two leakage functions $f_{S,j}$ and $h_{S,j}$ as input, B generates the leakage information $\Lambda f_{S,j}$ and $\Lambda h_{S,j}$ about the signer's current secret key ($SK_{ID,j,1}$, $SK_{ID,j,2}$) and certificate ($CK_{ID,j,1}$, $CK_{ID,j,2}$). Finally, B returns $\Lambda f_{S,j}$ and $\Lambda h_{S,j}$ to A.
- *Forgery phase*. In the phase, A generates a tuple (ID^*, $PK^*_{ID} = (PK^*_{ID,1}, PK^*_{ID,2})$, m^*, σ^*). We say that A wins the security game G_{LR-CBS} if the following conditions hold.
  (1) The output of the *Verifying* algorithm on (ID^*, PK^*_{ID}, m^*, σ^*) is "accept".
  (2) The *Signing query* on (ID^*, m^*) has never been issued.
  (3) The *Certificate generation query* on (ID^*, $PK^*_{ID,1}$) has never been issued if A is of Type I adversary. If A is of Type II adversary, both the *Secret key query* and *public key replace query* on ID^* have never been issued.

IV. THE PROPOSED LR-CBS SCHEME

In this section, the first LR-CBS scheme resistant to side-channel attacks is proposed that consists of five algorithms as follows.

- ***Setup***: This algorithm is performed by the CA that takes as input a security parameter τ, and sets an admissible bilinear pairing map $\hat{e}$ and its two associated groups $G =< g >$ and $G_T =< \hat{e}(g, g) >$ of a prime order p, where g and $\hat{e}(g, g)$ are a generator of G and G_T, respectively. The algorithm runs the following procedures to set the CA's initial system secret key $SK_{CA} = (SK_{CA,0,1}, SK_{CA,0,2})$ and public parameters PP:
  (1) Randomly choose $k \in Z^*_p$, and set a system secret key $SK_{CA} = g^k$ and the system public key $PK_{CA} = \hat{e}(g, g^k)$.

(2) Randomly choose $\alpha \in Z_p^*$, and set the initial system secret key $(SK_{CA,0,1}, SK_{CA,0,2}) = (g^{\alpha}, SK_{CA} \cdot g^{-\alpha})$.
(3) Randomly choose $\mu, v, x, y \in Z_p^*$, and set $U = g^{\mu}$, $V = g^{v}$, $X = g^{x}$ and $Y = g^{y}$.
(4) Set $PP = (G, G_T, p, \hat{e}, g, PK_{CA}, U, V, X, Y)$.

Finally, the CA publishes PP and keeps $(SK_{CA,0,1}, SK_{CA,0,2})$ in secret.

- ***User Key generation***: This algorithm is performed by a user that takes as input an identity ID and runs the following procedures to set the user's initial secret key $SK_{ID} = (SK_{ID,0,1}, SK_{ID,0,2})$ and the first partial public key $PK_{ID,1}$.
  (1) Randomly choose $s \in Z_p^*$, and set the user's secret key $SK_{ID} = g^{s}$ and the first partial public key $PK_{ID,1} = \hat{e}(g, g^{s})$.
  (2) Randomly choose $\beta \in Z_p^*$, and set the user's initial secret key $(SK_{ID,0,1}, SK_{ID,0,2}) = (g^{\beta}, SK_{ID} \cdot g^{-\beta})$.
- ***Certificate generation***: For the i-th *Certificate generation* algorithm invocation, the CA first refreshes the current system secret key $(SK_{CA,i,1}, SK_{CA,i,2})$ using $(SK_{CA,i-1,1}, SK_{CA,i-1,2})$. This algorithm is performed by the CA that takes as input a user's ID and the associated partial public key $PK_{ID,1}$, and runs the following procedures to set the user's initial certificate $CK_{ID} = (CK_{ID,0,1}, CK_{ID,0,2})$ and the second partial public key $PK_{ID,2}$.
  (1) Randomly choose $\gamma \in Z_p^*$, and refresh the CA's current system secret key $(SK_{CA,i,1}, SK_{CA,i,2}) = (SK_{CA,i-1,1} \cdot g^{\gamma}, SK_{CA,i-1,2} \cdot g^{-\gamma})$.
  (2) Randomly choose $t \in Z_p^*$, and compute the user's second partial public key $PK_{ID,2} = g^{t}$.
  (3) Set $b = ID||PK_{ID,1}$, and compute the temporary information $TI_{CG} = SK_{CA,i,1} \cdot (U \cdot V^{b})^{t}$ and the user's certificate $CK_{ID} = SK_{CA,i,2} \cdot TI_{CG}$.
  (4) Finally, the CA returns the certificate CK_{ID} and the second partial public key $PK_{ID,2}$ to the user.

  Upon receiving CK_{ID} and $PK_{ID,2}$, the user runs the following procedures to divide CK_{ID} into an initial certificate $(CK_{ID,0,1}, CK_{ID,0,2})$ and set her/his public key $PK_{ID} = (PK_{ID,1}, PK_{ID,2})$.
  (1) Randomly choose $\delta \in Z_p^*$, and set the user's initial certificate $(CK_{ID,0,1}, CK_{ID,0,2}) = (g^{\delta}, CK_{ID} \cdot g^{-\delta})$.
  (2) Set the user's public key $PK_{ID} = (PK_{ID,1}, PK_{ID,2})$.
- ***Signing***: For the j-th *Signing* algorithm invocation of a user (signer) with ID and $PK_{ID} = (PK_{ID,1}, PK_{ID,2})$, the signer first refreshes the current secret key $(SK_{ID,j,1}, SK_{ID,j,2})$ using $(SK_{ID,j-1,1}, SK_{ID,j-1,2})$ and the current certificate $(CK_{ID,j,1}, CK_{ID,j,2})$ using $(CK_{ID,j-1,1}, CK_{ID,j-1,2})$. This algorithm is performed by the signer that takes as input a message m, and runs the following procedures to return a signature $\sigma = (\sigma_1, \sigma_2)$.
  (1) Randomly choose $\beta \in Z_p^*$, and refresh the user's current secret key $(SK_{ID,j,1}, SK_{ID,j,2}) = (SK_{ID,j-1,1} \cdot g^{\beta}, SK_{ID,j-1,2} \cdot g^{-\beta})$.
  (2) Randomly choose $\delta \in Z_p^*$, and refresh the user's current certificate $(CK_{ID,j,1}, CK_{ID,j,2}) = (CK_{ID,j-1,1} \cdot g^{\delta}, CK_{ID,j-1,2} \cdot g^{-\delta})$.
  (3) Randomly choose $\eta \in Z_p^*$, and compute $\sigma_1 = g^{\eta}$, the temporary information $TI_S = SK_{ID,j,1} \cdot CK_{ID,j,1} \cdot (X \cdot Y^{m})^{\eta}$ and $\sigma_2 = SK_{ID,j,2} \cdot CK_{ID,j,2} \cdot TI_S$.
  (4) Set the signature $(ID, PK_{ID} = (PK_{ID,1}, PK_{ID,2}), m, \sigma = (\sigma_1, \sigma_2))$.
- ***Verifying***: Given a signature $(ID, PK_{ID} = (PK_{ID,1}, PK_{ID,2}), m, \sigma = (\sigma_1, \sigma_2))$, a verifier sets $b = ID||PK_{ID,1}$ and accepts the signature if the verifying equality $\hat{e}(g, \sigma_2) = PK_{ID,1} \cdot PK_{CA} \cdot \hat{e}(PK_{ID,2}, U \cdot V^{b}) \cdot \hat{e}(\sigma_1, X \cdot Y^{m})$ holds; otherwise rejects it.

By the key refreshing technique, we have

- $SK_{CA} = SK_{CA,0,1} \cdot SK_{CA,0,2} = \ldots = SK_{CA,i-1,1} \cdot SK_{CA,i-1,2} = SK_{CA,i,1} \cdot SK_{CA,i,2}$.
- $SK_{ID} = SK_{ID,0,1} \cdot SK_{ID,0,2} = \ldots = SK_{ID,j-1,1} \cdot SK_{ID,j-1,2} = SK_{ID,j,1} \cdot SK_{ID,j,2}$.
- $CK_{ID} = CK_{ID,0,1} \cdot CK_{ID,0,2} = \ldots = CK_{ID,j-1,1} \cdot CK_{ID,j-1,2} = CK_{ID,j,1} \cdot CK_{ID,j,2}$.

Hence, the correctness of the verifying equality is shown as follows.

$$
\begin{aligned}
&\hat{e}(g, \sigma_2) \\
&= \hat{e}(g, SK_{ID,j,2} \cdot CK_{ID,j,2} \cdot TI_S) \\
&= \hat{e}(g, SK_{ID,j,2} \cdot CK_{ID,j,2} \cdot SK_{ID,j,1} \cdot CK_{ID,j,1} \cdot (X \cdot Y^{m})^{\eta}) \\
&= \hat{e}(g, SK_{ID,j,2} \cdot SK_{ID,j,1} \cdot CK_{ID,j,2} \cdot CK_{ID,j,1} \cdot (X \cdot Y^{m})^{\eta}) \\
&= \hat{e}(g, SK_{ID} \cdot CK_{ID} \cdot (X \cdot Y^{m})^{\eta}) \\
&= \hat{e}(g, SK_{ID} \cdot SK_{CA,i,2} \cdot TI_{CG} \cdot (X \cdot Y^{m})^{\eta}) \\
&= \hat{e}(g, SK_{ID} \cdot SK_{CA,i,2} \cdot SK_{CA,i,1} \cdot (U \cdot V^{b})^{t} \cdot (X \cdot Y^{m})^{\eta}) \\
&= \hat{e}(g, SK_{ID} \cdot SK_{CA} \cdot (U \cdot V^{b})^{t} \cdot (X \cdot Y^{m})^{\eta}) \\
&= \hat{e}(g, SK_{ID}) \cdot \hat{e}(g, SK_{CA}) \cdot \hat{e}(g, (U \cdot V^{b})^{t} \cdot \hat{e}(g, (X \cdot Y^{m})^{\eta}) \\
&= \hat{e}(g, g^{s}) \cdot \hat{e}(g, g^{k}) \cdot \hat{e}(g^{t}, (U \cdot V^{b})^{t} \cdot \hat{e}(g^{\eta}, (X \cdot Y^{m})) \\
&= PK_{ID,1} \cdot PK_{CA} \cdot \hat{e}(PK_{ID,2}, U \cdot V^{b}) \cdot \hat{e}(\sigma_1, X \cdot Y^{m}).
\end{aligned}
$$

V. SECURITY ANALYSIS

In the proposed LR-CBS scheme, there are two types of adversaries that include Type I adversary (uncertified entity) and Type II adversary (honest-but-curious CA) according to the security game G_{LR-CBS}. In the generic bilinear group model, Theorems 1 and 2 demonstrate that the proposed LR-CBS scheme is existential unforgeability against UF-LR-CBS-ACMA attacks for Type I and Type II adversaries, respectively.

Theorem 1: In the generic bilinear group model, the proposed LR-CBS scheme is existential unforgeability against Type I adversary's UF-LR-CBS-ACMA attacks.

Proof: Let A_I be Type I adversary (uncertified entity) and can adaptively issue all queries in the security game G_{LR-CBS} at most q times. In the generic bilinear group model, there are two groups G and G_T, and each element of both G and G_T is encoded by a distinct bit-string. In addition, three group queries (oracles) Q_G, Q_T and Q_p, respectively, denote

the multiplication operation on the group G, the multiplication operation on the group G_T and the bilinear pairing map operation from $G \times G$ to G_T. Hence, Q_G, Q_T and Q_p must be added to the *Query* phase of the security game G_{LR-CBS} played by the adversary A_I and a challenger B. Three phases of the security game G_{LR-CBS} for the proposed LR-CBS scheme are given as follows.

- *Setup phase*: By taking a security parameter τ as input, B performs the Setup algorithm of the proposed LR-CBS scheme to produce an initial system secret key $SK_{CA} = (SK_{CA,0,1}, SK_{CA,0,2})$ and public parameters $PP = (G, G_T, p, g, \hat{e}, PK_{CA}, U, V, X, Y)$. Meanwhile, B creates three lists L_G, L_T and L_K to maintain the input parameters and associated responses of queries issued by A_I.
  - Two lists L_G and L_T are used to maintain all elements of G and G_T, respectively.
    (1) L_G is used to record the elements of G using the format $(\Omega G_{m,n,r}, \zeta G_{m,n,r})$. Each record $(\Omega G_{m,n,r}, \zeta G_{m,n,r})$ denotes an element of G, where $\Omega G_{m,n,r}$ is a multivariate polynomial with variates in G and coefficients in Z_p, and $\zeta G_{m,n,r}$ is the encoded bit-string of $\Omega G_{m,n,r}$. The indices m, n and r mean the m-type of query, the n-th query and the r-th element of G, respectively. Six records $(\Omega g, \check{c}aG_{I,1,1})$, $(\Omega U, \zeta G_{I,1,2})$, $(\Omega V, \zeta G_{I,1,3})$, $(\Omega X, \zeta G_{I,1,4})$, $(\Omega Y, \zeta G_{I,1,5})$ and $(\Omega SK_{CA}, \zeta G_{I,1,6})$ are initially added in L_G.
    (2) L_T is used to record the elements of G_T using the format $(\Omega T_{m,n,r}, \zeta T_{m,n,r})$. Each record $(\Omega T_{m,n,r}, \zeta T_{m,n,r})$ denotes an element of G_T, where $\Omega T_{m,n,r}$ is a multivariate polynomial with variates in G/G_T and coefficients in Z_p, and $\zeta G_{m,n,r}$ is the encoded bit-string of $\Omega T_{m,n,r}$. Three indices m, n and r have the same meanings as L_G. A record $(\Omega PK_{CA}, \zeta T_{I,1,1})$ is initially added in L_T, where $\Omega PK_{CA} = \Omega g \cdot \Omega SK_{CA}$.

    For the related queries in the *Query* phase described later, B uses the following two rules to maintain L_G and L_T.
    (1) Upon receiving a transformation request along with $\Omega G_{m,n,r}/\Omega T_{m,n,r}$, B checks whether there exists $(\Omega G_{m,n,r}, \zeta G_{m,n,r})/(\Omega T_{m,n,r}, \zeta T_{m,n,r})$ in L_G/L_T. If it is found, B returns the corresponding bit-string $\zeta G_{m,n,r}/\zeta T_{m,n,r}$. Otherwise, B randomly chooses and returns a distinct encoded bit-string $\zeta G_{m,n,r}/\zeta T_{m,n,r}$ while adding $(\Omega G_{m,n,r}, \zeta G_{m,n,r})/(\Omega T_{m,n,r}, \zeta T_{m,n,r})$ in L_G/L_T.
    (2) Upon receiving a transformation request along with $\zeta G_{m,n,r}/\zeta T_{m,n,r}$ in L_G/L_T, B returns the corresponding polynomial $\Omega G_{m,n,r}/\Omega T_{m,n,r}$.
  - L_K consists of tuples with format $(ID, replace, \Omega SK_{ID}, \Omega PK_{ID,1}, \Omega CK_{ID}, \Omega PK_{ID,2})$, where ΩSK_{ID}, $\Omega PK_{ID,1}$, ΩCK_{ID} and $\Omega PK_{ID,2}$ are multivariate polynomials in L_G /L_T that respectively denote a user's secret key SK_{ID}, certificate CK_{ID} and public key $(PK_{ID,1}, PK_{ID,2})$. The *replace* field is initially set to "false", which denotes that the user's public key has never been replaced by A_I. If A_I issues the *Public key replace query* (ID) in the *Query* phase, B changes the *replace* field of the ID's tuple to be "true".

  At the end of this phase, B returns the corresponding bit-strings of these public parameters Ωg, ΩU, ΩV, ΩX, ΩY and ΩPK_{CA} to A_I.
- *Query phase*: In the phase, A_I may adaptively issue the following queries at most q times.
  - *Group query* $Q_G(\zeta G_{Q,i,1}, \zeta G_{Q,i,2}, OP)$: For the i-th query Q_G along with $(\zeta G_{Q,i,1}, \zeta G_{Q,i,2})$ and an OP operation (multiplication/division), B runs the following procedures to return the resulting bit-string $\zeta G_{Q,i,3}$.
    (1) B transforms two bit-strings $\zeta G_{Q,i,1}$ and $\zeta G_{Q,i,2}$ to get the associated polynomials $\Omega G_{Q,i,1}$ and $\Omega G_{Q,i,2}$ in L_G, respectively.
    (2) B computes the polynomial $\Omega G_{Q,i,3} = \Omega G_{Q,i,1} + \Omega G_{Q,i,2}$ if $OP =$ "multiplication", or the polynomial $\Omega G_{Q,i,3} = \Omega G_{Q,i,1}$ - $\Omega G_{Q,i,2}$ if $OP =$ "division".
    (3) B transforms and returns the bit-string $\zeta G_{Q,i,3}$ of the resulting polynomial $\Omega G_{Q,i,3}$.
  - *Group query* $Q_T(\zeta T_{Q,i,1}, \zeta T_{Q,i,2}, OP)$: For the i-th query Q_T along with $(\zeta T_{Q,i,1}, \zeta T_{Q,i,2})$ and an OP operation (multiplication/division), B runs the similar procedures in the *Group query* Q_G and returns the bit-string $\zeta T_{Q,i,3}$.
  - *Pairing query* $Q_P(\zeta G_{P,i,1}, \zeta G_{P,i,2})$: For the i-th query Q_P along with $(\zeta G_{P,i,1}, \zeta G_{P,i,2})$, B runs the following procedures:
    (1) B transforms $\zeta G_{P,i,1}$ and $\zeta G_{P,i,2}$ to get the associated polynomials $\Omega G_{P,i,1}$ and $\Omega G_{P,i,2}$, respectively.
    (2) B computes the polynomial $\Omega T_{P,i,1} = \Omega G_{P,i,1} \cdot \Omega G_{P,i,2}$.
    (3) B transforms and returns the bit-string $\zeta T_{P,i,1}$ of the resulting polynomial $\Omega T_{P,i,1}$.
  - *User key generation query* (ID): For the i-th User key generation query along with the identity ID, B searches $(ID, replace, \Omega SK_{ID}, \Omega PK_{ID,1}, \Omega CK_{ID}, \Omega PK_{ID,2})$ in L_K. If it is found, B returns the corresponding bit-strings of ΩSK_{ID} and $\Omega PK_{ID,1}$ to A_I. Otherwise, B runs the following procedures:
    (1) B selects a new variate $\Omega TG_{UKG,i,1}$ in G.
    (2) B sets the user's secret key polynomial $\Omega SK_{ID} = \Omega TG_{UKG,i,1}$ and the first partial public key polynomial $\Omega PK_{ID,1} = \Omega TG_{UKG,i,1} \cdot \Omega_g$. Meanwhile, B adds $(ID, false, \Omega SK_{ID}, \Omega PK_{ID,1}, -, -)$ in L_K.

(3) B transforms and returns the corresponding bit-strings ζSK_{ID} and $\zeta PK_{ID,1}$ of both ΩSK_{ID} and $\Omega PK_{ID,1}$ to A_I.

- *Secret key query* (ID): Upon receiving this query along with ID, B searches (ID, *replace*, ΩSK_{ID}, $\Omega PK_{ID,1}$, ΩCK_{ID}, $\Omega PK_{ID,2}$) in L_K. If it is found and the replace = "false", B gets ΩSK_{ID} and transforms it to return ζSK_{ID} to A_I. Otherwise, B issues the *User key generation query* (ID) to return the corresponding bit-strings ζSK_{ID} and $\zeta PK_{ID,1}$ to A_I.
- *Certificate generation query* (ID, $\zeta PK_{ID,1}$): For the i-th query along with ID and the first partial public key bit-string $\zeta PK_{ID,1}$, B runs the following procedures:

(1) B chooses a new variate $\Omega TG_{CG,i,1}$ in G to set the second partial public key polynomial $\Omega PK_{ID,2} = \Omega TG_{CG,i,1}$.
(2) B sets $b = ID||\zeta PK_{ID,1}$.
(3) B chooses a new variate $TG_{CG,i,2}$ in G and sets the certificate polynomial $\Omega CK_{ID} = \Omega SK_{CA} + \Omega TG_{CG,i,2} \cdot (\Omega U + b \cdot \Omega V)$ while updating (ID, *false*, ΩSK_{ID}, $\Omega PK_{ID,1}$, $\Omega PK_{ID,2}$, ΩCK_{ID}) in L_K.
(4) B transforms and returns $\zeta PK_{ID,2}$ and ζCK_{ID} of $\Omega PK_{ID,2}$ and ΩCK_{ID} to A_I.

- *Certificate generation leak query* (i, $f_{CG,i}$, $h_{CG,i}$): For the i-th *Certificate generation leak query* along with two leakage functions $f_{CG,i}$ and $h_{CG,i}$ such that $|f_{CG,i}| \leq \lambda$ and $|h_{CG,i}| \leq \lambda$, B sends the leakage information $\Lambda f_{CG,i}$ and $\Lambda h_{CG,i}$ to A_I, where $\Lambda f_{CG,i} = f_{CG,i}(SK_{CA,i,1}, \gamma, t)$ and $\Lambda h_{CG,i} = h_{CG,i}(SK_{CA,i,2}, \gamma, TI_{CG})$. Note that A_I can issue the *Certificate generation leak query* only once for the i-th *Certificate generation query*.
- *Public key retrieve query* (ID): Upon receiving this query along with ID, B searches (ID, *replace*, ΩSK_{ID}, $\Omega PK_{ID,1}$, ΩCK_{ID}, $\Omega PK_{ID,2}$) in L_K, and returns the user's public key bit-strings $\zeta PK_{ID,1}$ and $\zeta PK_{ID,2}$ to A_I.
- *Public key replace query* (ID, ($\zeta PK'_{ID,1}$, $\zeta PK'_{ID,2}$)): Upon receiving this query along with ID and the new public key bit-strings $\zeta PK'_{ID,1}$ and $\zeta PK'_{ID,2}$, B transforms ($\zeta PK'_{ID,1}$, $\zeta PK'_{ID,2}$) to get the public key polynomials $\Omega PK'_{ID,1}$ and $\Omega PK'_{ID,2}$ while updating (ID, *true*, *null*, $\Omega PK'_{ID,1}$, *null*, $\Omega PK'_{ID,2}$) in L_K.
- *Singing query* (ID, m): For the i-th *Signing query* of ID along with the message m, B runs the following procedures to get the signature polynomials $\Omega\sigma_1$ and $\Omega\sigma_2$.

(1) B uses ID to search (ID, *replace*, ΩSK_{ID}, $\Omega PK_{ID,1}$, ΩCK_{ID}, $\Omega PK_{ID,2}$) in L_K.
(2) B chooses a new variate $\Omega TG_{S,i,1}$ in G and sets $\Omega\sigma_1 = \Omega TG_{S,i,1}$.
(3) B sets $\Omega\sigma_2 = \Omega SK_{ID} + \Omega CK_{ID} + \Omega TG_{S,i,1} \cdot (\Omega X + m \cdot \Omega Y)$.
(4) B respectively transforms $\Omega\sigma_1$ and $\Omega\sigma_2$ to return the signature bit-strings $\zeta\sigma_1$ and $\zeta\sigma_2$ to A_I.

- *Signing leak query* (ID, j, $f_{S,j}$, $h_{S,j}$): For the j-th *Signing leak query* of the signer ID along with two leakage functions $f_{S,j}$ and $h_{S,j}$ such that $|f_{S,j}| \leq \lambda$ and $|h_{S,j}| \leq \lambda$, B sends the leakage information $\Lambda f_{S,j}$ and $\Lambda h_{S,j}$ to A_I, where $\Lambda f_{S,j} = f_{S,j}(SK_{ID,j,1}, CK_{ID,j,1}, \beta, \delta, \eta)$ and $\Lambda h_{S,j} = h_{S,j}(SK_{ID,j,2}, CK_{ID,j,2}, \beta, \delta, \eta, TI_S)$. Note that A_I can issue the *Signing leak query* only once for the j-th *Signing query*.

- *Forgery phase*: The adversary A_I outputs (ID^*, ($\zeta PK_{ID^*,1}$, $\zeta PK_{ID^*,2}$), m^*, ($\zeta\sigma_1^*$, $\zeta\sigma_2^*$)). The *Certificate generation query* (ID^*, $\zeta PK^*_{ID,1}$) is disallowed to be issued in the *Query phase*. B transforms the bit-strings to get the corresponding polynomials $\Omega PK_{ID^*,1}$, $\Omega PK_{ID^*,2}$, $\Omega\sigma_1^*$ and $\Omega\sigma_2^*$ while setting $b^* = ID^*||\zeta PK_{ID^*,1}$. The adversary A_I wins the game if the equality $g^{\Omega P\Omega_2^*} = \Omega PK_{ID^*,1} + \Omega PK_{CA} + \Omega PK_{ID^*,2} \cdot (\Omega U + b \cdot \Omega V) + \Omega\sigma_1^* \cdot (\Omega X \cdot + m^* \cdot \Omega Y)$ holds.

Before evaluating the advantage that A_I wins the security game G_{LR-CBS}, we have to discuss the total amount of both L_G and L_T, and the maximal polynomial degrees of all elements in L_G and L_T.

(1) In the *Setup phase*, two lists L_G and L_T are created while six elements are initially added in L_G and one element is initially added in L_T. In the *Query phase*, six kinds of queries could increase new elements in L_G and L_T as follows.
   - For each query of Q_G, Q_T and Q_P, at most three new elements are putted in L_G or L_T.
   - For each *Signing query*, at most three new elements are putted in L_G.
   - For each *Certificate generation query*, at most three new elements are putted in L_G.
   - For each *User key generation query*, at most one new element is putted in L_G and L_T, respectively.

   Let q_O be the total amount of all Q_G, Q_T and Q_P queries. Let q_S, q_C and q_U, respectively, denote the amounts of the *Signing query*, *Certificate generation query* and *User key generation query* issued by A_I. Let $|L_G|$ and $|L_T|$ be the amounts of all elements in L_G and L_T, respectively. Since A_I may issue all queries at most q times, we have $|L_G| + |L_T| \leqq 7 + 3q_O + 3q_S + 3q_C + 2q_U \leqq 3q$.
(2) The maximal polynomial degree of all elements in L_G is 2 due to the following reasons.
   - Ωg, ΩU, ΩV, ΩX, ΩY and ΩSK_{CA} are new variates so that these polynomials have degree 1.
   - The certificate ΩCK_{ID} has degree 2.
   - In the *Signing query*, $\Omega\sigma_1$ has degree 1 and $\Omega\sigma_2$ has degree 2.
   - In the group query Q_G, the degree of $\Omega G_{Q,i,3}$ is the maximal degree of $\Omega G_{Q,i,1}$ or $\Omega G_{Q,i,2}$.
(3) The maximal polynomial degree of all elements in L_T is 4 due to the following reasons.

- The CA's public key ΩPK_{CA} has degree 2.
- In the *User key generation query*, $\Omega PK_{ID,1}$ has degree 2.
- In Q_P, the maximal polynomial degree of all elements in L_T is at most 4 since it is computed by two polynomials in L_G.
- In Q_T, the degree of $\Omega T_{Q,i,3}$ is the maximal degree of $\Omega T_{Q,i,1}$ and $\Omega T_{Q,i,2}$.

Assume that the total amount of all variates in both L_G and L_T is n. Hence, B chooses n random values $z_1, z_2, \ldots, z_n$ in Z_p^*. A_I is said to win the security game G_{LR-CBS} when one of the following two cases occurs:

***Case* 1**: A_I may discover a collision in L_G or L_T. Namely, there are two polynomials ΩG_i and ΩG_j in L_G such that $\Omega G_i(z_1, z_2, \ldots, z_n) = \Omega G_j(z_1, z_2, \ldots, z_n)$, or there are two polynomials ΩT_i and ΩT_j in L_T such that $\Omega T_i(z_1, z_2, \ldots, z_n) = \Omega T_j(z_1, z_2, \ldots, z_n)$.

***Case* 2**: A_I may output a valid signature $(ID, (\zeta PK_{ID^*,1}, \zeta PK_{ID^*,2}), m^*, (\zeta\sigma_1^*, \zeta\sigma_2^*))$ such that $\Omega g \cdot \Omega\sigma_2^* = \Omega PK_{ID^*,1} + \Omega PK_{CA} + \Omega PK_{ID^*,2} \cdot (\Omega U + b \cdot \Omega V) + \Omega\sigma_1^* \cdot (\Omega X \cdot + m \cdot \Omega Y)$.

Firstly, let us discuss A_I's advantage in G_{LR-CBS} without requesting *Signing leak query* and *Certificate generation leak query*. Afterward, A_I's advantage in G_{LR-CBS} with issuing *Signing leak query* and *Certificate generation leak query* is evaluated.

• ***Without requesting Signing leak query and Certificate generation leak query***: In the situation, except *Signing leak query* or *Certificate generation leak query*, A_I is allowed to issue the other queries in G_{LR-CBS}. The advantage that A_I wins G_{LR-CBS} has two cases as follows.

***Case* 1** $\mathcal{V}$ Let us evaluate the advantage that A_I respectively discovers a collision in L_G and L_T. Let ΩG_i and ΩG_j be two distinct element polynomials in L_G. The advantage of discovering a collision is the probability that $\Omega G_C = \Omega G_i - \Omega G_j$ is a zero polynomial, namely, $\Omega G_C(z_1, z_2, \ldots, z_n) = 0$. By Lemma 2, because the maximal polynomial degree of all elements in L_G is at most 2 and no leak query is allowed ($\lambda = 0$), the probability of $\Omega G_C(z_1, z_2, \ldots, z_n) = 0$ is at most $2/p$. Since there are $2^{|L_G|}$ distinct pairs $(\Omega G_i, \Omega G_j)$ in L_G, the advantage of discovering a collision in L_G is at most $(2/p)^{2^{|L_G|}}$. Similarly, the advantage of discovering a collision in L_T is at most $(4/p)^{2^{|L_T|}}$. By the result mentioned earlier, we have $|L_G| + |L_T| \leqq 3q$. Let Pr[Case 1] denote the advantage that *Case* 1 happens, we have the following inequality.

$$\begin{aligned}\Pr[Case\ 1] &\leqq (2/p)^{2^{|L_G|}} + (4/p)^{2^{|L_T|}} \\ &\leqq (4/p)(|L_G| + |L_T|)^2 \\ &\leqq 36q^2/p.\end{aligned}$$

***Case* 2** $\mathcal{V}$ The probability of this case is the advantage that A_I outputs a valid signature$(ID^*, (\zeta PK_{ID^*,1}, \zeta PK_{ID^*,2}), m^*, (\zeta\sigma_1^*, \zeta\sigma_2^*))$, namely, the signature satisfies $\Omega f = \Omega PK_{ID^*,1} + \Omega PK_{CA} + \Omega PK_{ID^*,2} \cdot (\Omega U + b \cdot \Omega V) + \Omega\sigma_1^* \cdot (\Omega X + m \cdot \Omega Y) - \Omega g \cdot \Omega\sigma_2^* = 0$. Since Ωf has degree at most 3, the advantage of forging a valid signature is at most $3/p$.

By Cases 1 and 2, the advantage that A_I wins G_{LR-CBS} without requesting *Signing leak query* or *Certificate generation leak query*, denoted by $\Pr_{A-I-W}$, satisfies the following inequality.

$$\begin{aligned}Adv_{A-I-W} &\leqq \Pr[Case\ 1] + \Pr[Case\ 2] \\ &\leqq 36q^2/p + 3/p \\ &\leqq O(q^2/p).\end{aligned}$$

If $q = poly(\log p)$, Adv_{A-I-W} is negligible.

• ***With requesting Signing leak query and Certificate generation leak query***: In this situation, A_I is allowed to request the *Signing leak query* and *Certificate generation leak query*. For the i-th *Certificate generation leak query* along with two leakage functions $f_{CG,i}$ and $h_{CG,i}$ such that $|f_{CG,i}| \leq \lambda$ and $|h_{CG,i}| \leq \lambda$, A_I may obtain the leakage information $\Lambda f_{CG,i} = f_{CG,i}(SK_{CA,i,1}, \gamma, t)$ and $\Lambda h_{CG,i} = h_{CG,i}(SK_{CA,i,2}, \gamma, TI_{CG})$ that is discussed as below.

- γ, t: For each user's ID, γ and t are random values so that their leakage is helpless to recover the system secret key SK_{CA} and the user's certificate CK_{ID}.
- $(SK_{CA,i,1}, SK_{CA,i,2})$: For the CA's system secret key SK_{CA}, we have $SK_{CA} = SK_{CA,i-1,1} \cdot SK_{CA,i-1,2} = SK_{CA,i,1} \cdot SK_{CA,i,2}$. Meanwhile, the leakage information of $SK_{CA,i-1,1}/SK_{CA,i-1,2}$ is independent of that of $SK_{CA,i,1}/SK_{CA,i,2}$ due to the multiplicative blinding technique. Hence, A_I can obtain at most λ bits of SK_{CA}.
- TI_{CG}: TI_{CG} is a temporary value and used to generate CK_{ID}. Thus, A_I can obtain at most λ bits of CK_{ID}.

For the j-th *Signing leak query* of the signer ID along with two leakage functions $f_{S,j}$ and $h_{S,j}$ such that $|f_{S,j}| \leq \lambda$ and $|h_{S,j}| \leq \lambda$, A_I may obtain the leakage information $\Lambda f_{S,j} = f_{S,j}(SK_{ID,j,1}, CK_{ID,j,1}, \beta, \delta, \eta)$ and $\Lambda h_{S,j} = h_{S,j}(SK_{ID,j,2}, CK_{ID,j,2}, \beta, \delta, \eta, TI_S)$ that is discussed as follows.

- β, δ: β and δ are random values so that their leakage is helpless to recover the user secret key SK_{ID} or the user's certificate CK_{ID}.
- η: η is a random value and used in the signature generation. Thus, A_I can obtain at most 2λ bits of σ_1.
- $(SK_{ID,j,1}, SK_{ID,j,2})$: A_I may get the user's whole secret key SK_{ID} by issuing the *Secret key query* with the target user's identity ID^*.
- $(CK_{ID,j,1}, CK_{ID,j,2})$: For the user's certificate CK_{ID}, we have $CK_{ID} = CK_{ID,j-1,1} \cdot CK_{ID,j-1,2} = CK_{ID,j,1} \cdot CK_{ID,j,2}$. Meanwhile, the leakage information of $CK_{ID,j-1,1}/CK_{ID,j-1,2}$ is independent of that of $CK_{ID,j,1}/CK_{ID,j,2}$ due to the multiplicative blinding technique. Hence, A_I can obtain at most λ bits of CK_{ID}.
- TI_S: TI_S is a temporary value and used to generate σ_2. Thus, A_I can obtain at most λ bits of σ_2.

In the following, let us evaluate the advantage that A_I wins G_{LR-CBS} with issuing the *Signing leak query* and *Certificate generation leak query*, denoted by Adv_{A-I}. By the *Public key replace query*, A_I can get the target user's secret key SK_{ID}. To forge a valid signature, the helpful information consists of the leakage information of the CA's system secret key

SK_{CA} and the target user's certificate CK_{ID}. Three events are defined as follows.

(1) Event *ESKCA*: It means the event that A_I gets the CA's whole system secret key SK_{CA} by both $\Lambda f_{CG,i}$ and $\Lambda h_{CG,i}$ while means the complement of the event *ESKCA*.
(2) Event *ECK*: It means the event that A_I gets the user's whole certificate key CK_{ID} by both $\Lambda f_{S,j}$ and $\Lambda h_{S,j}$ while means the complement of the event *ECK*.
(3) Event *EF*: It means the event that A_I forges a valid signature.

By the probabilities of the events, the advantage Adv_{A-I} satisfies the following inequality.

$$\begin{aligned} Adv_{A-I} &= \Pr[EF] \\ &= \Pr[EF \wedge (ECK \vee ESKCA)] \\ &\quad + Pr[EF \wedge (\overline{ECK} \wedge \overline{ESKCA})] \\ &= \Pr[EF \wedge ESKCA] + Pr[EF \wedge \overline{ESKCA} \wedge ECK] \\ &\quad + Pr[EF \wedge (\overline{ECK} \wedge \overline{ESKCA})] \\ &\leqq \Pr[ESKCA] + Pr[\overline{ESKCA} \wedge ECK] \\ &\quad + Pr[EF \wedge (\overline{ECK} \wedge \overline{ESKCA})]. \end{aligned}$$

Let us discuss the upper bound of Pr[*ESKCA*]. In the *Certificate generation* phase of our LR- CBS scheme, the user's certificate CK_{ID} is a signature on the user's information $ID||PK_{ID,1}$ by adopting the signature scheme in [15]. The probability Pr[*ESKCA*] is bounded by the probability that the adversary can compute the CA's whole system secret key SK_{CA}. By applying Lemma 5 in [15], we have Pr[*ESKCA*] $\leqq O((q^2/p)^*2^{2\lambda})$. Next, let us discuss the upper bound of Pr[$\overline{ESKCA} \wedge ECK$]. Since A_I is a Type I adversary, A_I can get the secret key SK_{ID} of any entity, but is unable to obtain the certificate CK_{ID} of a target entity. Under the event, A_I cant obtain any useful information to forge a signature by *Certificate generation leak query*. However, A_I may get some useful information by *Signing leak query*. In this case, Pr[$\overline{ESKCA} \wedge ECK$] is the event that A_I can obtain the user's certificate CK_{ID} by both $\Lambda f_{S,j}$ and $\Lambda h_{S,j}$. Since A_I can obtain the secret key of any entity, the probability of forging a signature under the condition $\overline{ESKCA} \wedge ECK$ is similar to the probability Pr[*ESKCA*]. Therefore, we have Pr[$\overline{ESKCA} \wedge ECK$] $\leqq O((q^2/p)^*2^{2\lambda})$. Under the event $\overline{ECK} \wedge \overline{ESKCA}$, the meaningful leakage information for A_I to forge a valid signature is the partial information of the user current certificate ($CK_{ID,j,1}$, $CK_{ID,j,2}$). Since A_I can learn at most λ bits information about the user current certificate, the probability of A_I forging a valid signature with at most λ bits leakage information is Pr[$EF \wedge (\overline{ECK} \wedge \overline{ESKCA})$] $\leqq O((q^2/p)^*2^{\lambda})$. By the discussions above, we have the following inequality.

$$\begin{aligned} Adv_{A-I} &= \Pr[EF] \\ &\leqq \Pr[ESKCA] + Pr[\overline{ESKCA} \wedge ECK] \\ &\quad + Pr[EF \wedge (\overline{ECK} \wedge \overline{ESKCA})] \\ &\leqq O((q^2/p)^*2^{2\lambda}) + O((q^2/p)^*2^{2\lambda}) \\ &\quad + O((q^2/p)^*2^{\lambda}) \end{aligned}$$

Therefore, we have $Adv_{A-I} \leqq O((q^2/p)^*2^{2\lambda})$. By Lemma 2, if $\lambda < \log p - \omega(\log\log p)$, Adv_{A-I} is negligible. □

Theorem 2: In the generic bilinear group model, the proposed LR-CBS scheme is existential unforgeability against Type II adversary's UF-LR-CBS-ACMA attacks.

Proof: Let A_{II} be of Type II adversary (honest-but-curious CA) so that it knows the CA's system secret key and does not need to issue the *Certificate generation query* and *Certificate generation leak query* to B in the security game G_{LR-CBS}. As the proof of Theorem 1, A_{II} may also issue all other queries adaptively at most q times. Three phases of the security game G_{LR-CBS} for the proposed LR-CBS scheme are given as follows.

- *Setup phase*: As the *Setup* phase of Theorem 1, B produces the CA's system secret key SK_{CA} and public parameters $PP = (G, G_T, p, g, \hat{e}, PK_{CA}, U, V, X, Y)$ of the proposed LR-CBS scheme. In addition, three lists L_G, L_T and L_K are also created to maintain the input parameters and associated responses of queries issued by A_{II}. At the end of this phase, B returns the bit-strings of these public parameters g, U, V, X, Y and PK_{CA} to A_{II}. B also returns the system secret key bit- string ζSK_{CA} to A_{II} since A_{II} is an honest-but-curious CA.
- *Query phase*: Because A_{II} an honest-but-curious CA and holds the CA's system secret key, it can generate certificates of all entities. In this phase, A_{II} may request the following queries adaptively at most q times.
  - Q_G, Q_T, Q_P, *User key generation*, *Secret key*, *Public key retrieve*, *Public key replace queries* are identical to those queries in the *Query* phase of Theorem 1.
  - *Signing query (ID, m)*: For the i-th *Signing query* of *ID* along with the message m, B runs the following procedures to get the signature polynomials $\Omega\sigma_1$ and $\Omega\sigma_2$.
    (1) B uses *ID* to search (*ID*, *replace*, ΩSK_{ID}, $\Omega PK_{ID,1}$, ΩCK_{ID}, $\Omega PK_{ID,2}$) in L_K. If the user's secret key polynomial ΩSK_{ID} is not found in L_K, B issues the *User key generation query(ID)*. Moreover, if the certificate ΩCK_{ID} is not found in L_K, B uses the records of the queries Q_G, Q_T and Q_P to get the corresponding polynomial ΩCK_{ID} and the second partial public key polynomial $\Omega PK_{ID,2}$.
    (2) B chooses a new variate $\Omega TG_{S,i,1}$ in G and sets $\Omega\sigma_1 = \Omega TG_{S,i,1}$.
    (3) B computes $\Omega\sigma_2 = \Omega SK_{ID} + \Omega CK_{ID} + \Omega TG_{S,i,1} \cdot (\Omega X + m \cdot \Omega Y)$.
    (4) B respectively transforms $\Omega\sigma_1$ and $\Omega\sigma_2$ to return the signature bit-strings $\zeta\sigma_1$ and $\zeta\sigma_2$ to A_{II}.
  - *Signing leak query* (*ID*, j, $f_{S,j}$, $h_{S,j}$): For the j-th *Signing leak query* along with two leakage functions $f_{S,j}$ and $h_{S,j}$ such that $|f_{S,j}| \leq \lambda$ and $|h_{S,j}| \leq \lambda$, B sends the leakage information $\Lambda f_{S,j}$ and $\Lambda h_{S,j}$ to A_{II}, where $\Lambda f_{S,j} = f_{S,j}(SK_{ID,j,1}, CK_{ID,j,1}, \beta, \delta, \eta)$ and $\Lambda h_{S,j} = h_{S,j}(SK_{ID,j,2}, CK_{ID,j,2}, \beta, \delta, \eta, TI_S)$.

Note that A_{II} can issue the *Signing leak query* only once for the j-th *Signing query*.

- *Forgery phase*: The adversary A_{II} outputs (ID^*, ($\zeta PK_{ID^*,1}$, $\zeta PK_{ID^*,2}$), m^*, ($\zeta\sigma_1^*$, $\zeta\sigma_2^*$)). The *Secret key query* (ID^*) and *Public key replace query* (ID^*, ($\zeta PK'_{ID^*,1}$, $\zeta PK'_{ID^*,2}$)) have never been issued in the *Query* phase. B transforms these bit-strings to get the corresponding polynomials $\Omega PK_{ID^*,1}$, $\Omega PK_{ID^*,2}$, $\Omega\sigma_1^*$ and $\Omega\sigma_2^*$ while setting $b^* = ID^*||\zeta PK_{ID^*,1}$. The adversary A_{II} wins the game if the equality $\Omega g \cdot \Omega\sigma_2^* = \Omega PK_{ID^*,1} + \Omega PK_{CA} + \Omega PK_{ID^*,2} \cdot (\Omega U + b \cdot \Omega V) + \Omega\sigma_1^* \cdot (\Omega X + m^* \cdot \Omega Y)$ holds.

By similar arguments with Theorem 1, the total amount of elements in both L_G and L_T satisfies the inequality $|L_G| + |L_T| \leqq 3q$. The maximal polynomial degrees of all elements in L_G and L_T are at most 2 and 4, respectively.

• ***Without issuing Signing leak query***: By similar arguments with Theorem 1, we have Pr[Case 1] $\leqq 36q^2/p$ and Pr[Case 2] $\leqq 3/p$. The advantage that A_{II} wins G_{LR-CBS} without issuing *Signing leak query*, denoted by Adv_{A-II-W}, satisfies the following inequality.

$$\begin{aligned} Adv_{A-II-W} &\leqq \Pr[Case\ 1]] + \Pr[Case\ 2]] \\ &\leqq 36q^2/p + 3/p \\ &= O(q^2/p). \end{aligned}$$

If $q = poly(\log p)$, Adv_{A-II-W} is negligible.

• ***With issuing Signing leak query***: A_{II} is allowed to issue the *Signing leak query*. For the j-th *Signing leak query* of ID along with two leakage functions $f_{S,j}$ and $h_{S,j}$ such that $|f_{S,j}| \leq \lambda$ and $|h_{S,j}| \leq \lambda$, A_{II} may obtain the leakage information $\Lambda f_{S,j}(SK_{ID,j,1}, CK_{ID,j,1}, \beta, \delta, \eta)$ and $\Lambda h_{S,j}(SK_{ID,j,2}, CK_{ID,j,2}, \beta, \delta, \eta, TI_S)$ that are discussed as follows.

- β, δ: β and δ are random values for refreshing the user's secret key SK_{ID}. At most 2λ bits of β and δ is helpless to recover the user's secret key SK_{ID}.
- η: η is a random value during the signature generation. Thus, A_{II} can obtain at most 2λ bits of σ_1.
- ($SK_{ID,j,1}$, $SK_{ID,j,2}$): For the secret key SK_{ID} of the user with ID, we have $SK_{ID} = SK_{ID,j-1,1} \cdot SK_{ID,j-1,2} = SK_{ID,j,1} \cdot SK_{ID,j,2}$. Meanwhile, the leakage information of $SK_{ID,j-1,1}/SK_{ID,j-1,2}$ is independent of that of $SK_{ID,j,1}/SK_{ID,j,2}$ due to the multiplicative blinding technique. Hence, A_{II} can obtain at most λ bits of SK_{ID}.
- ($CK_{ID,j,1}$, $CK_{ID,j,2}$): A_{II} is an honest-but-curious CA and holds the CA's system secret key, it can generate the certificate CK_{ID} of any entity.
- TI_S: TI_S is a temporary value and used to generate σ_2. Thus, A_{II} can obtain at most λ bits of σ_2.

Now, let us evaluate the advantage that A_{II} wins G_{LR-CBS} with issuing the *Signing leak query*, denoted by Adv_{A-II}. Since A_{II} simulates the role of honest-but-curious CA, it can generate the target user's certificate CK_{ID}. The useful information of forging a valid signature is decided by the leakage information of the target user's secret key SK_{ID}. Two events are defined as follows.

(1) Event *ESKID*: It means the event that A_{II} gets the user's whole secret key SK_{ID} by both $\Lambda f_{S,j}$ and $\Lambda h_{S,j}$ while $\overline{ESKID}$ means the complement of the event *ESKID*.
(2) Event *EF*: It means the event that A_{II} forges a valid signature.

By the probabilities of the events, the advantage Adv_{A-II} satisfies the following inequality.

$$\begin{aligned} Adv_{A-II} &= \Pr[EF] \\ &= \Pr[EF \wedge ESKID] + \Pr[EF \wedge \overline{ESKID}] \\ &\leqq \Pr[ESKID] + \Pr[EF \wedge \overline{ESKID}] \end{aligned}$$

Let us discuss the upper bound of $\Pr[EF \wedge \overline{ESKID}]$. Under the event $\overline{ESKID}$, since A_{II} can learn at most λ bits information for the user's secret key SK_{ID}, the advantage that A_{II} forges a valid signature is $\Pr[EF \wedge \overline{ESKID}] \leqq O((q^2/p)^* 2^{\lambda})$. In the situation without issuing *Signing leak query*, A_{II}'s advantage has the inequality $Adv_{A-II-W} \leqq 36q^2/p = O(q^2/p)$. Since A_{II} can learn at most λ bits of the user's secret key SK_{ID}, we have $\Pr[ESKID] \leqq O((q^2/p) \cdot 2^{\lambda})$.

$$\begin{aligned} Adv_{A-II} &= \Pr[EF] \\ &\leqq \Pr[ESKID] + Pr[EF \wedge \overline{ESKID}] \\ &\leqq O((q^2/p) \cdot 2\lambda) + O((q^2/p)^* 2^{\lambda}). \end{aligned}$$

Therefore, we have $Adv_{A-II} \leqq O((q^2/p)^* 2^{\lambda})$. By Lemma 2, if $\lambda < \log p - \omega(\log\log p)$, Adv_{A-II} is negligible. □

VI. PERFORMANCE ANALYSIS AND COMPARISONS

In this section, the performance analysis and comparisons are given. For convenience, the following notations are defined to denote the computation costs of two time-consuming operations in suitable bilinear pairing groups [42].

- T_p: The running time of a bilinear pairing operation $\hat{e} : G \times G \to G_T$.
- T_e: The running time of an exponentiation operation in G or G_T.

Indeed, the running time of a multiplication operation on G or G_T is smaller than both T_p and T_e so that it is negligible. In addition, $|G|$ denotes the bit-length of one element in G.

Table 1 lists the comparisons between several previous CBS or SCBS schemes [31], [37], [39] and our LR-CBS scheme in terms of signing computation cost, verifying computation cost, signature size, proof model, CA adversary type and side-channel attacks. Obviously, the performance of the proposed LR-CBS scheme is not better than that of the previously proposed CBS or SCBS schemes [31], [37], [39]. For the CA adversary type, Zhou and Cui's CBS scheme [39] is secure against the malicious-but-passive CA attack, namely, the CA may forge a new signature (ID, m, σ') from an existing signature (ID, m, σ), where ID and m are the same identity and message, respectively. Nevertheless, the CA cannot forge a signature on an arbitrary message m if a signer with identity ID did not generate a signature on the message m. Indeed, the adversary model against the honest-but-curious CA attack defined in [30], [32], [33], and [37] is enough to

TABLE 1. Comparisons between several previous CBS/SCBS schemes and our LR-CBS scheme.

	Liu *et al.*'s CBS scheme [31]	Hung *et al.*'s SCBS scheme [37]	Zhou and Cui's CBS scheme	our LR-CBS scheme
Signing	$8T_e$	$2T_e$	$4T_e$	$7T_e$
Verifying	$4T_p + 2T_e$	$3T_p + T_e$	$7T_p + 2T_e$	$3T_p + 2T_e$
Signature size	$3\|G\|$	$\|G\|$	$3\|G\|$	$2\|G\|$
Proof model	Standard model	Random oracle model	Standard model	Generic bilinear group model
CA adversary type	Honest-but-curious	Honest-but-curious	Malicious-but-passive	Honest-but-curious
Side-channel attacks	Insecure	Insecure	Insecure	Secure

model the abilities of adversaries. Indeed, all existing CBS or SCBS schemes (including [31], [37], and [39]) do not resist side-channel attacks. Up to date, no leakage-resilient CBS (LR-CBS) scheme resistant to side-channel attacks is proposed. The point is that our scheme is the first LR-CBS scheme which not only is resistant to side-channel attacks but also possesses overall unbounded leakage property.

VII. CONCLUSIONS AND FUTURE WORK

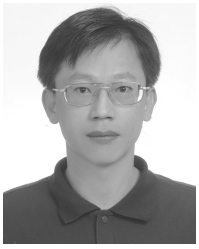
A novel adversary model of LR-CBS schemes resistant to side-channel attacks under the continual leakage model has been defined. The novel adversary model permits Types I and II adversaries to continuously obtain partial information of both the CA's system secret key in the certificate generation algorithm and a signer's secret key and associated certificate in the signing algorithm. Under the novel adversary model with continual key leakage, the *first* LR-CBS scheme resistant to side-channel attacks was proposed. For resisting to continual key leakage, the proposed LR-CBS scheme adopts the key update technique to refresh the CA's system secret key after running each certificate generation algorithm and a signer's secret key and certificate after running each signing algorithm. Meanwhile, in the generic bilinear group model, the proposed LR-CBS scheme is proved to be existential unforgeability against adaptive chosen-message attacks of Types I and II adversaries under the novel adversary model with continual key leakage. As compared with several previous CBS and SCBS schemes, our proposed LR-CBS requires some extra computation operations due to the key update process. The point is that our scheme is the first LR-CBS scheme resistant to side-channel attacks under the continual leakage model. Certainly, it is an interesting issue and future work to propose a novel LR-CBS scheme against the malicious-but-passive CA attack.

REFERENCES

[1] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, 1978.
[2] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inf. Theory*, vol. IT-31, no. 4, pp. 469–472, Jul. 1985.
[3] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology—CRYPTO*, vol. 196. Berlin, Germany: Springer, 1984, pp. 47–53.
[4] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology—CRYPTO*, vol. 2139. Berlin, Germany: Springer, 2001, pp. 213–229.
[5] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Advances in Cryptology—ASIACRYPT*, vol. 2894. Berlin, Germany: Springer, 2003, pp. 452–473.
[6] T.-T. Tsai and Y.-M. Tseng, "Revocable certificateless public key encryption," *IEEE Syst. J.*, vol. 9, no. 3, pp. 824–833, Sep. 2015.
[7] Y.-H. Hung, Y.-M. Tseng, and S.-S. Huang, "A revocable certificateless short signature scheme and its authentication application," *Informatica*, vol. 27, no. 3, pp. 549–572, 2016.
[8] C. Gentry, "Certificate-based encryption and the certificate revocation problem," in *Advances in Cryptology—EUROCRYPT*, vol. 2656. Berlin, Germany: Springer, 2003, pp. 272–293.
[9] P. C. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems," in *Advances in Cryptology—CRYPTO*, vol. 1109. Berlin, Germany: Springer, 1996, pp. 104–113.
[10] D. Boneh, R. A. Demillo, and R. J. Lipton, "On the importance of checking cryptographic protocols for faults," in *Advances in Cryptology—EUROCRYPT*, vol. 1233. Berlin, Germany: Springer, 1997, pp. 37–51.
[11] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO*, vol. 1666. Berlin, Germany: Springer, 1999, pp. 388–397.
[12] D. Brumley and D. Boneh, "Remote timing attacks are practical," *Comput. Netw.*, vol. 48, no. 5, pp. 701–716, 2005.
[13] 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, 2013.
[14] 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, 2016.
[15] D. Galindo and S. Vivek, "A practical leakage-resilient signature scheme in the generic group model," in *Proc. SAC*, vol. 7707. Berlin, Germany: Springer, 2013, pp. 50–65.
[16] F. Tang, H. Li, Q. Niu, and B. Liang, "Efficient leakage-resilient signature schemes in the generic bilinear group model," in *Proc. Int. Conf. Inf. Secur. Pract. Exper.*, vol. 8434. Berlin, Germany: Springer, 2014, pp. 418–432.
[17] Z. Brakerski, Y. Kalai, J. Katz, and V. Vaikuntanathan, "Overcoming the hole in the bucket: Public-key cryptography resilient to continual memory leakage," in *Proc. 51st Annu. IEEE Symp. Found. Comput. Sci.*, Oct. 2010, pp. 501–510.
[18] 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*, vol. 7237. Berlin, Germany: Springer, 2012, pp. 117–134.
[19] J.-D. Wu, Y.-M. Tseng, and S.-S. Huang, "Leakage-resilient ID-based signature scheme in the generic bilinear group model," *Secur. Commun. Netw.*, vol. 9, no. 17, pp. 3987–4001, 2016.
[20] Q. Lin, H. Yan, Z. Huang, W. Chen, J. Shen, and Y. Tang, "An ID-based linearly homomorphic signature scheme and its application in blockchain," *IEEE Access*, vol. 6, pp. 20632–20640, 2018.
[21] H. Xiong, T. H. Yuen, C. Zhang, S. M. Yiu, and Y.-J. He, "Leakage-resilient certificateless public key encryption," in *Proc. 1st ACM Workshop Asia Public-Key Cryptography*, 2013, pp. 13–22.
[22] J.-D. Wu, Y.-M. Tseng, S.-S. Huang, and W.-C. Chou, "Leakage-resilient certificateless key encapsulation scheme," *Informatica*, vol. 29, no. 1, pp. 125–155, 2018.
[23] J.-D. Wu, Y.-M. Tseng, and S.-S. Huang, "Leakage-resilient certificateless signature under continual leakage model," *Inf. Technol. Control*, vol. 47, no. 2, pp. 286–363, 2018.
[24] 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.

[25] 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.
[26] 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.
[27] J. Alwen, Y. Dodis, and D. Wichs, "Leakage-resilient public-key cryptography in the bounded-retrieval model," in *Advances in Cryptology—CRYPTO*, vol. 5677. Berlin, Germany: Springer, 2009, pp. 36–54.
[28] D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchical identity based encryption with constant size ciphertext," in *Advances in Cryptology—EUROCRYPT*, vol. 3494. Berlin, Germany: Springer, 2005, pp. 440–456.
[29] B.-G. Kang, J.-H. Park, and S.-G. Hahn, "A Certificate-based signature scheme," in *Proc. CT-RSA*, vol. 2964. Berlin, Germany: Springer, 2004, pp. 99–111.
[30] J. Li, X. Huang, Y. Mu, W. Susilo, and Q. Wu, "Certificate-based signature: security model and efficient construction," in *Proc. EuroPKI*, vol. 4582. Berlin, Germany: Springer, 2007, pp. 110–125.
[31] J. K. Liu, J. Baek, W. Susilo, and J. Zhou, "Certificate-based signature schemes without pairings or random oracles," in *Proc. ISC*, vol. 5222. Berlin, Germany: Springer, 2008, pp. 285–297.
[32] J. Zhang, "On the security of a certificate-based signature scheme and its improvement with pairings," in *Proc. ISPEC*, vol. 5451. Berlin, Germany: Springer, 2009, pp. 47–58.
[33] W. Wu, Y. Mu, W. Susilo, and X. Huang, "Certificate-based signatures revisited," *J. Universal Comput. Sci.*, vol. 15, no. 8, pp. 1659–1684, 2009.
[34] J.-K. Liu, F. Bao, and J. Zhou, "Short and efficient certificate-based signature," in *Proc. Netw. Workshops*, vol. 6827. Berlin, Germany: Springer, 2011, pp. 167–178.
[35] L. Cheng, Y. Xiao, and G. Wang, "Cryptanalysis of a certificate-based on signature scheme," *Procedia Eng.*, vol. 29, pp. 2821–2825, Jan. 2012.
[36] J. Li, X. Huang, Y. Zhang, and L. Xu, "An efficient short certificate-based signature scheme," *J. Syst. Softw.*, vol. 85, no. 2, pp. 314–322, Feb. 2012.
[37] Y.-H. Hung, S.-S. Huang, and Y.-M. Tseng, "A short certificate-based signature scheme with provable security," *Inf. Technol. Control*, vol. 45, no. 3, pp. 243–253, 2015.
[38] Y. Lu and J. Li, "Improved certificate-based signature scheme without random oracles," *IET Inf. Security*, vol. 10, no. 2, pp. 80–86, 2016.
[39] C. Zhou and Z. Cui, "Certificate-based signature scheme in the standard model," *IET Inf. Secur.*, vol. 11, no. 5, pp. 256–260, Sep. 2017.
[40] 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, 2008.
[41] M. Scott, N. Costigan, and W. Abdulwahab, "Implementing cryptographic pairings on smartcards," in *Proc. CHES*, vol. 4249. Berlin, Germany: Springer, 2006, pp. 134–147.
[42] M. Scott, "On the efficient implementation of pairing-based protocols," in *Proc. Int. Conf. Cryptogr. Coding*, vol. 7089. Berlin, Germany: Springer, 2011, pp. 296–308.

JUI-DI WU received the B.S. and M.S. degrees from the Department of Mathematics, National Changhua University of Education, Taiwan, in 2006 and 2008, respectively, where he is currently pursuing the Ph.D. degree. His research interests include leakage-resilient cryptography and pairing-based cryptography.



YUH-MIN TSENG received the B.S. degree from National Chiao Tung University at Guangfu Campus, Hsinchu, Taiwan, in 1988, the M.S. degree from National Taiwan University, Taipei, Taiwan, in 1990, and the Ph.D. degree from National Chung Hsing University, Taichung, Taiwan, in 1999. He is currently a Professor with the Department of Mathematics, National Changhua University of Education, Changhua, Taiwan. He has published over 100 scientific journal and conference papers on various research areas of cryptography, security, and computer networks. His research interests include cryptography, network security, computer networks, and mobile communications. He is a member of the IEEE Computer Society, the IEEE Communications Society, and the Chinese Cryptology and Information Security Association. In 2006, he was a recipient of the Wilkes Award from the British Computer Society. He currently serves as an Editor for several international journals.

SEN-SHAN HUANG received the Ph.D. degree from the University of Illinois at Urbana–Champaign, under the supervision of Prof. B. 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 network security.

TUNG-TSO TSAI received the B.S. degree from the Department of Applied Mathematics, Chinese Culture University, Taiwan, in 2006, the M.S. degree from the Department of Applied Mathematics, National Hsinchu University of Education, Taiwan, in 2009, and the Ph.D. degree from the Department of Mathematics, National Changhua University of Education, Taiwan, in 2014. He is currently an Engineer with the Department of Research, Foxconn, Taiwan. His research interests include applied cryptography, pairing-based cryptography, and network security.