INFORMATICA, 2019, Vol. 30, No. 4, 711–728

DOI: http://dx.doi.org/10.15388/Informatica.2019.226

A New Provably Secure Certificateless Signature with Revocation in the Standard Model

Qian MEI, Yanan ZHAO, Hu XIONG*

School of Information and Software Engineering,
University of Electronic Science and Technology of China, Chengdu 610054, China
e-mail: xionghu.uestc@gmail.com

Received: November 2018; accepted: March 2019

Abstract. The primitive of certificateless signature, since its invention, has become a widely studied paradigm due to the lack of key escrow problem and certificate management problem. However, this primitive cannot resist catastrophic damage caused by key exposure. Therefore, it is necessary to integrate revocation mechanism into certificateless signature. In this paper, we propose a new certificateless signature scheme with revocation (RCLS) and prove its security under the standard model. In the meanwhile, our scheme can resist malicious-but-passive Key Generation Center (KGC) attacks that were not possible in previous solutions. The theoretical analysis shows our scheme has high efficiency and practicality.

Key words: certificateless signature, revocation, standard model.

1. Introduction

In the traditional public key infrastructure (PKI) based signature system (ElGamal, 1985), the user's identity is bound to the corresponding public key through a certificate which is issued by a trusted certificate authority. Obviously, the complex and expensive certificate management has become an obstacle to the development of PKI-based system. Therefore, the notion of identity-based (shorten as ID-based) signature scheme (Shamir, 1984) has been proposed to alleviate this problem. In ID-based signature, users' public keys are their unique identity information that is publicly known, while the user's private key is created by a private key generator (PKG) with a master secret key. In this way, the certificate is eliminated in ID-based signature because the public key is derived from the identity of the user. Since the private key of all users is created by PKG, the signature of any entity can easily be forged by PKG, which results in the notorious key escrow problem. Thankfully, the certificateless signature (CLS) system (Al-Riyami and Paterson, 2003) preserves the advantages of eliminating the required certificates in ID-based signature while avoiding key escrow problem. In a CLS-based system, the private key includes the secret value chosen by the entity itself and the partial private key generated by the Key Generation Center (KGC). It is easy to observe that the key escrow problem was solved successfully.

*Corresponding author.

Since then, many efficient CLS schemes were proposed in the literature (Huang *et al.*, 2007; Yap *et al.*, 2006; Zhang *et al.*, 2006; Karati *et al.*, 2018a).

For any public key system, it is indispensable to have the function of revocation. Imagine a situation in which the employee in charge of confidential documents in the company is leaving. In order to ensure that confidential files are not disclosed, the private key held by the employee must be revoked. Similarly, with the frequent use of signature operations in the signature system, private keys are inevitably leaked. In this case, the user with the compromised key is no longer reliable. In the traditional public key system, the revoked public key can be known by the user through the certificate revocation list (CRL) (Housley *et al.*, 2002). Apparently, this method does not apply to the certificateless cryptosystem due to the lack of certificates. So far, there are two revocation mechanisms in the process of certificateless public key cryptosystem development. One mechanism is to split the partial private key of the user generated by KGC into two parts, one is delivered to the user and the other is sent to the Security Mediator (an online mediator) (Ju *et al.*, 2005; Yap *et al.*, 2007). In this approach, the Security Mediator is required to create each signature which causes expensive burden. Moreover, it is necessary for the Security Mediator to maintain large quantities of secret keys which makes it easier for an attacker to compromise a key. Different from the Security Mediator based revocation approach, another certificateless system with revocation has been introduced where the user's partial private key is updated in period time (Tsai and Tseng, 2015). When the user's private key is compromised or the user leaves, KGC stops updating the partial private key. Nevertheless, most existing schemes in this mechanism prove their security under the random oracle model. This paradigm is to model the hash function as random oracles in the security proof. Unfortunately, when random oracles are instantiated by a concrete hash function, these schemes are vulnerable to be broken (Canetti *et al.*, 2004). Tsai *et al.* (2014) introduced a CLS scheme with revocation mechanism, whose security was theoretically proven by using the idea of standard model. However, we observed that the proposed scheme is neither efficient nor can resist malicious-but-passive KGC attacks.

For the above reasons, we propose a new certificateless signature scheme with revocation (RCLS) and prove its security under the standard model. In our scheme, the partial private key issued by the KGC is spilt into two independent parts, where the former is associated with the identity of the user and the latter is related to the time period. In the meanwhile, our scheme can resist malicious-but-passive KGC attacks. Specific contributions are as follows:

1. This paper first reveals the insecurity of the scheme in Tsai *et al.* (2014), and displays the forgery attack as well as the reason why their scheme is easily broken.
2. Next, this paper proposes a new certificateless signature scheme with revocation, whose security proof is provided in the standard model based on the Computational Diffie–Hellman assumption. And the attack mounted by the malicious-but-passive KGC is able to be resisted in our scheme.
3. Finally, by comparing the performance and properties with related works, the RCLS scheme in this paper outperforms the existing works.

1.1. *Related Work*

Al-Riyami and Paterson (2003) first introduced the notion of certificateless signature (CLS) scheme and gave a concrete construction of CLS scheme. In the CLS scheme, users' private key was divided into two parts. One part is chosen by users themselves and another part is generated by a third party called KGC, which successfully avoids the key escrow problem. Thanks to maintaining the advantages of eliminating the required certificates in ID-based signature while avoiding key escrow problem, much attention has been paid to the research of CLS. After Al-Riyami and Paterson (2003) proposed the first CLS scheme, dozens of CLS schemes were proposed in terms of different research lines (Jia *et al.*, 2018; Huang *et al.*, 2005; Karati *et al.*, 2018b; He *et al.*, 2012; Xiong *et al.*, 2019). By considering the criticism of random models, Liu *et al.* (2007) proposed the first concrete CLS scheme which was proved to be secure in the standard model. Unfortunately, Xiong *et al.* (2008) indicated that Liu *et al.*'s scheme cannot be able to resist malicious-but-passive KGC attack. In this attack, KGC can maliciously generate system parameters during the system setup stage and forge the signature with the knowledge of the secret value used to calculate the system parameters. For this type of attack, Xiong *et al.* also put forward an improved scheme in their paper. In the meanwhile, Yuan *et al.* (2009) presented a CLS scheme in the standard model. Unfortunately, Xia *et al.* (2010) showed that both Xiong *et al.*'s improved scheme and Yuan *et al.*'s scheme are insecure under the key replacement attack. Aiming to resist the key replacement attack proposed by Xia *et al.*, Yu *et al.* (2012) presented a new CLS scheme in the standard model. Subsequently, Yu *et al.*'s scheme was proved to be insecure under the attack of Xiong *et al.* Recently, Shim (2018a) proposed a CLS scheme which makes it possible to prove security in the standard model. In short, most of the existing schemes proposed in the standard are insecure.

The idea of revocation was first proposed by Boneh *et al.* (2001) and was used in RSA-type cryptosystems. In their scheme, a semi-trusted server called Security Mediator (SEM) is introduced to issue tokens. If a user wants to sign or decrypt a message, he/she must get the token for the message. The scheme revokes the ability of the user to sign or decrypt by stopping issuing tokens to the user. Following the works of Boneh *et al.* (2001), some well-designed schemes have been constructed in certificateless cryptosystem. Chow *et al.* (2006) introduced the concept of Security Mediator into the certificateless (SMC) cryptosystem for the first time, and proposed a formal security model. After that, the first pairing-free provable secure SMC scheme as well as a concrete construction was presented by Yap *et al.* (2007). Unfortunately, SEM is involved in the generation of each signature which causes expensive burden. Furthermore, it is necessary for the Security Mediator to preserve large numbers of secret keys which makes it easier for an attacker to compromise a key. To deal with this problem, Tsai and Tseng (2015) proposed a new RCLS scheme. In their scheme, the user's partial private key is updated in period time. When the user's private key is compromised or the user leaves, KGC stops updating the partial private key. At the same time, Shen *et al.* (2013) showed an efficient certificateless encryption (CLE) with the function of revocation, which was proved to be CCA2-secure under the standard model. Xiong and Qin (2015) presented an RCLS scheme which can

resist signing key exposure and applied it into the wireless body area networks. However, Shim (2018b) pointed out that Xiong *et al.*'s scheme is vulnerable to the type I adversary. Sun *et al.* (2014) proposed an RCLS schemes from bilinear pairings that was proved to be secure in the random oracle model. To eliminate the random oracle model, Tsai *et al.* (2014) introduced an RCLS scheme which was proven in the standard model. However, in this paper, Tsai *et al.*'s scheme is demonstrated to be neither efficient nor resistant to malicious-but-passive KGC attacks. After that, many CLS or CLE schemes with revocation mechanism were presented in the literature (Zhang *et al.*, 2015; Hung *et al.*, 2016; Sun *et al.*, 2018; Zheng *et al.*, 2017). However, most of the existing schemes cannot resist malicious-but-passive KGC attacks.

1.2. *Organization*

This paper's structure is as follows: some preliminaries including bilinear pairings, complexity assumption, system framework and security notions are introduced in Section 2. Section 3 briefly analyses Tsai *et al.'* scheme and then displays a forgery attack about their scheme. A concrete RCLS scheme and associated security proof are demonstrated in Section 4. Section 5 provides the performance evaluation. Section 6 summarizes this paper.

2. Preliminaries

This section describes some mathematical knowledge, formal definition and security model, which are utilized in the proposed revocable certificateless signature scheme.

2.1. *Bilinear Pairing*

Chosen two multiplicative cyclic groups $\mathcal{G}, \mathcal{G}_T$ of prime order p, given two random generators u, v of $\mathcal{G}$, the bilinear map $\hat{e} : \mathcal{G} \times \mathcal{G} \to \mathcal{G}_T$ needs to satisfy the following features:

1. Bilinearity: For any $a, b \in \mathcal{Z}_p^*$, $\hat{e}(u^a, v^b) = \hat{e}(u, v)^{ab}$.
2. Non-degeneracy: $\hat{e}(u, v) \neq 1$.
3. Computability: There exists an algorithm to compute bilinear map $\hat{e} : \mathcal{G} \times \mathcal{G} \to \mathcal{G}_T$.

2.2. *Mathematical Concept and Assumption*

- *Computational Diffie–Hellman (CDH) Problem*: Given a tuple (g, g^a, g^b) to calculate g^{ab} where $a, b \in \mathcal{Z}_p^*$, $g \in \mathcal{G}$.
- *Computational Diffie–Hellman (CDH) Assumption*: The CDH assumption in $\mathfrak{G}$ holds if there does not exist polynomial-time algorithm $\mathfrak{B}$ to solve the CDH problem with non-negligible advantage formulated as

$$\mathrm{Adv}_{\mathfrak{B}}^{CDH} = \Pr\left[\mathfrak{B}\left(g, g^a, g^b\right) = g^{ab} | g \in \mathcal{G},\ a, b \in \mathcal{Z}_p^*\right].$$

2.3. *Outline of RCLS*

An RCLS scheme consists of eight algorithms whose details are depicted below.

- Setup: A KGC produces a system master secret key SSK and system public parameters SPP on input the security parameter k.
- Partial-Private-Key-Extraction: A KGC produces a partial private key PPK_{ID} on input SPP, SSK and an identity ID.
- Time-Key-Update: A KGC produces the time key TK_T on input SSK, ID and a time period T.
- Secret-Value-Generation: A user with identity ID calculates the secret value sv_{ID} on input SPP.
- Public-Key-Generation: A user calculates the public key PK_{ID} on input SPP, ID and the secret value sv_{ID} of this identity ID.
- Secret-Key-Generation: A non-revocation user calculates the full secret key SK_{ID} on input SPP, ID, sv_{ID} and the corresponding PPK_{ID} and TK_T.
- RCL-Sign: A signer produces a signature σ on input SPP, ID, sv_{ID}, SK_{ID} and a message M.
- RCL-Verify: A verifier outputs *VALID* or *INVALID* to demonstrate signature σ's validity on input SPP, ID, σ, M, T, PK_{ID}.

2.4. *Security Model of RCLS*

According to the scheme (Al-Riyami and Paterson, 2003), there are two kinds of attackers in the certificateless signature setting, which are usually called the Type-I adversary $\mathfrak{A}_I$ and Type-II adversary $\mathfrak{A}_{II}$. $\mathfrak{A}_I$ models a malicious user who has the right to replace a legitimate user's public key without knowing the system master secret key. $\mathfrak{A}_{II}$ models a malicious-but-passive KGC who has knowledge of the system master secret key while is not allowed to replace any public key. For a RCLS scheme's security, there is one more adversary called a revoked user $\mathfrak{A}_{ru}$ who cannot obtain the time key but still has the right to replace the public key (Sun *et al.*, 2014). To better explain the attacking ability of these adversaries, we first define six oracles that adversary $\mathfrak{A} \in \{\mathfrak{A}_I, \mathfrak{A}_{II}, \mathfrak{A}_{ru}\}$ can access.

- *Public-Key-Extract Query*: After receiving an identity ID, this oracle produces the user's public key PK_{ID}.
- *Partial-Private-Key-Extract Query*: After receiving ID, this oracle produces the user ID's partial private key PPK_{ID}.
- *Time-Key-Update Query*: After receiving (ID, T), this oracle produces the time key TK_T.
- *Secret-Value-Extract Query*: After receiving ID, this oracle produces the user ID's secret value sv_{ID}.
- *Public-Key-Replace Query*: After receiving (ID, PK'_{ID}), a user ID's public key is replaced with PK'_{ID} through this oracle.
- *RCL-Sign Query*: After receiving ID, T, PK_{ID} and a message M, this oracle produces a valid signature σ.

DEFINITION 1. An RCLS scheme is existentially unforgeable (EUF) against a $(t, q_{PK}, q_{PPK}, q_{TK}, q_{SV}, q_{PKR}, q_S)$ Type-I adaptively chosen message (CMA) adversary $\mathfrak{A}_I$ if $\mathfrak{A}_I$ runs in polynomial time t, makes at most q_{PK} queries to the oracle *Public-Key-Extract Query*, q_{PPK} queries to the oracle *Partial-Private-Key-Extract Query*, q_{TK} queries to the oracle *Time-Key-Update Query*, q_{SV} queries to the oracle *Secret-Value-Extract Query*, q_{PKR} queries to the oracle *Public-Key-Replace Query*, q_S queries to the oracle *RCL-Sign Query* and wins in **Game I** with a negligible advantage.

Game I.

Setup: A challenger $\mathfrak{B}$ generates the system master secret key SSK and system public parameters SPP by performing the algorithm Setup. After that, $\mathfrak{B}$ returns SPP to $\mathfrak{A}_I$.

Query: $\mathfrak{A}_I$ queries onto all oracles defined above adaptively.

Forgery: After finishing all queries, $\mathfrak{A}_I$ outputs a forged signature σ^* on the message M^*.

DEFINITION 2. A RCLS scheme is $(t, q_{PK}, q_{PPK}, q_{TK}, q_{SV}, q_{PKR}, q_S)$-EUF-CMA-secure for Type-II adversary $\mathfrak{A}_{II}$ if $\mathfrak{A}_{II}$ wins in **Game II** with a negligible advantage.

Game II.

Setup: A challenger $\mathfrak{B}$ generates the system secret key SSK and system public parameters SPP by performing the algorithm Setup. After that, $\mathfrak{B}$ returns SPP and SSK to $\mathfrak{A}_{II}$.

Query: $\mathfrak{A}_{II}$ queries onto those oracles defined above adaptively except for the oracle *Partial-Private-Key-Extract Query* and the oracle *Time-Key-Update Query*.

Forgery: After finishing all queries, $\mathfrak{A}_{II}$ outputs a forged signature σ^* on the message M^*.

DEFINITION 3. A RCLS scheme is $(t, q_{PK}, q_{PPK}, q_{TK}, q_{SV}, q_{PKR}, q_S)$-EUF-CMA-secure for a revoked user $\mathfrak{A}_{ru}$ if $\mathfrak{A}_{ru}$ wins in **Game III** with a negligible advantage.

Game III.

Setup: A challenger $\mathfrak{B}$ generates the system secret key SSK and system public parameters SPP by performing the algorithm Setup. After that, $\mathfrak{B}$ returns SPP to $\mathfrak{A}_{ru}$.

Query: $\mathfrak{A}_{ru}$ queries onto all oracles defined above adaptively.

Forgery: After finishing all queries, $\mathfrak{A}_{ru}$ outputs a forged signature σ^* on the message M^*.

Note that, when the forgery satisfies the following requirements, adversary $\mathfrak{A} \in \{\mathfrak{A}_I, \mathfrak{A}_{II}, \mathfrak{A}_{ru}\}$ will win the above **Game I**, **Game II** and **Game III**:

1. If $\mathfrak{A} \in \mathfrak{A}_I$, $\mathfrak{A}$ has never queried the oracle *Partial-Private-Key-Extract Query* with ID^*.
2. If $\mathfrak{A} \in \mathfrak{A}_{II}$, $\mathfrak{A}$ has never queried the oracle *Secret-Value-Extract Query* with ID^* nor queried the oracle *Public-Key-Replace Query* with PK_{ID^*}.
3. If $\mathfrak{A} \in \mathfrak{A}_{ru}$, $\mathfrak{A}$ has never queried the oracle *Time-Key-Update Query* with (ID^*, T^*).
4. $\mathfrak{A}$ has never queried the oracle *RCL-Sign Query* with (ID^*, M^*, T^*).
5. $VALID \leftarrow \text{RCL-Verify}(\mathsf{SPP}, ID^*, \sigma^*, M^*, T^*, \mathsf{PK}_{ID^*})$.

3. A Brief Analysis of Tsai *et al.*'s Scheme

This section first sketches out the certificateless signature with revocation scheme of Tsai *et al.* (2014), and then demonstrates that Tsai *et al.*'s RCLS scheme cannot resist the malicious-but-passive KGC attacks.

3.1. *Overview of Tsai et al.'s RCLS Scheme*

Define five collision-resistant hash functions $\mathcal{H}_0 : \{0,1\}^* \to \{0,1\}^{n_u}, \mathcal{H}_1 : \{0,1\}^* \to \{0,1\}^{n_t}, \mathcal{H}_2 : \mathcal{G} \times \mathcal{G} \to \{0,1\}^{n_k}, \mathcal{H}_3 : \mathcal{G} \times \mathcal{G} \to \{0,1\}^{n_s}, \mathcal{H}_4 : \{0,1\}^* \to \{0,1\}^{n_m}$, where n_u, n_t, n_m, n_s, n_k, are fixed lengths from $\mathcal{Z}$.

- Setup: Taken k as the security parameter, KGC generates a bilinear map $\hat{e} : \mathcal{G} \times \mathcal{G} \to \mathcal{G}_T$, where $\mathcal{G}, \mathcal{G}_T$ are cyclic groups of order p. Furthermore, KGC picks $x, y \in \mathcal{Z}_p^*, g, g_1, g_2 \in \mathcal{G}$ and calculates $g_1 = g^{x+y}, \mathsf{SSK} = g_2^x$, where g, SSK denote a generator of $\mathcal{G}$ and the system secret key respectively. In addition, let g_2^y denotes the time secret key. After that, KGC randomly selects $u', t', k', s', m' \in \mathcal{G}$, and five vectors $\mathbf{U} = [u_i] \in \mathcal{G}^{n_u}, \mathbf{T} = [t_i] \in \mathcal{G}^{n_t}, \mathbf{K} = [k_i] \in \mathcal{G}^{n_k}, \mathbf{S} = [s_i] \in \mathcal{G}^{n_s}, \mathbf{M} = [m_i] \in \mathcal{G}^{n_m}$. Finally, KGC issues the system public parameters $\mathsf{SPP} = \langle \mathcal{G}, \mathcal{G}_T, \hat{e}, g, g_1, g_2, \mathbf{U}, \mathbf{T}, \mathbf{K}, \mathbf{S}, \mathbf{M}, \mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3, \mathcal{H}_4, u', t', k', s', m' \rangle$.
- Partial-Private-Key-Extraction: After receiving SSK, SPP, and a user's identity ID, KGC will first calculate a set as $\nu = \mathcal{H}_0(ID) = \{\nu_1, \nu_2, \ldots, \nu_{n_u}\}$. Then KGC calculates the user's partial private key $\mathsf{PPK}_{ID} = (\mathsf{PPK}^{(1)}, \mathsf{PPK}^{(2)}) = (g_2^x (u' \prod_{i=1}^{n_u} u_i^{\nu_i})^{r_\nu}, g^{r_\nu})$ where r_ν is randomly selected by KGC from $\mathcal{Z}_p^*$.
- Time-Key-Update: Upon receiving SSK, ID and a time period T, KGC calculates a set as $\nu t = \mathcal{H}_1(ID, t) = (\nu t_1, \nu t_2, \ldots, \nu t_{n_t})$, and sets the time key $\mathsf{TK}_T = (\mathsf{TK}^{(1)}, \mathsf{TK}^{(2)}) = (g_2^y (t' \prod_{i=1}^{n_t} t_i^{\nu t_i})^{r_t}, g^{r_t})$ where r_t is randomly selected by KGC from $\mathcal{Z}_p^*$.
- Secret-Value-Generation: A user with identity ID randomly picks $x_1, x_2 \in \mathcal{Z}_p^*$ and sets the secret value $sv_{ID} = (x_1, x_2)$.
- Public-Key-Generation: The user with identity ID calculates $\mathsf{PK}_{ID} = (\mathsf{PK}^{(1)}, \mathsf{PK}^{(2)}) = (g^{x_1}, g^{x_2})$ as the public key.
- Secret-Key-Generation: The user ID computes a set as $\nu u = \mathcal{H}_2(\mathsf{PK}^{(1)}, \mathsf{PK}^{(2)}) = \{\nu u_1, \nu u_2, \ldots, \nu u_{n_k}\}$ and $\nu s = \mathcal{H}_3(\mathsf{PK}^{(1)}, \mathsf{PK}^{(2)}) = \{\nu s_1, \nu s_2, \ldots, \nu s_{n_s}\}$. Then, the algorithm calculates the secret key $\mathsf{SK}_{ID} = g_2^{x_1} (k' \prod_{i=1}^{n_k} k_i^{\nu u_i})^{x_1} (s' \prod_{i=1}^{n_s} s_i^{\nu s_i})^{x_2}$.
- RCL-Sign: Upon receiving $\mathsf{PPK}_{ID} = (\mathsf{PPK}^{(1)}, \mathsf{PPK}^{(2)})$ and $\mathsf{TK}_T = (\mathsf{TK}^{(1)}, \mathsf{TK}^{(2)})$, a signer ID can sign a message $M \in \{0,1\}^*$ with a secret key SK_{ID} by performing the following steps:
  (1) Define a set as $\nu m = \mathcal{H}_4(M) = \{\nu m_1, \nu m_2, \ldots, \nu m_{n_m}\}$.
  (2) Randomly select $r_m \in \mathcal{Z}_p^*$ and calculate $\sigma_1 = \mathsf{PPK}^{(1)} \cdot \mathsf{TK}^{(1)} \cdot \mathsf{SK}_{ID} (m' \prod_{i=1}^{n_m} m_i^{\nu m_i})^{r_m}, \sigma_2 = \mathsf{PPK}^{(2)}, \sigma_3 = \mathsf{TK}^{(2)}, \sigma_4 = g^{r_m}$.
  (3) Output a revocable certificateless signature $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4)$ of the message M and return to a verifier.

- RCL-Verify: Given $\mathsf{SPP}, ID, \sigma, M, t, \mathsf{PK}_{ID}$, the verifier calculates five sets $\nu = \mathcal{H}_0(ID), \nu t = \mathcal{H}_1(ID, t), \nu u = \mathcal{H}_2(\mathsf{PK}^{(1)}, \mathsf{PK}^{(2)})$,$\nu s = \mathcal{H}_3(\mathsf{PK}^{(1)}, \mathsf{PK}^{(2)})$,$\nu m = \mathcal{H}_4(M)$. After that, the verifier can check the equation : $\hat{e}(g, \sigma_1) \stackrel{?}{=} \hat{e}(g_1, g_2) \cdot \hat{e}(\sigma_2, u' \prod_{i=1}^{n_u} u_i^{\nu_i}) \cdot \hat{e}(\sigma_3, t' \prod_{i=1}^{n_t} t_i^{\nu t_i}) \cdot \hat{e}(\mathsf{PK}^{(1)}, g_2(k' \prod_{i=1}^{n_k} k_i^{\nu u_i})) \cdot \hat{e}(\mathsf{PK}^{(2)}, s' \prod_{i=1}^{n_s} s_i^{\nu s_i}) \cdot \hat{e}(\sigma_4, m' \prod_{i=1}^{n_m} m_i^{\nu m_i})$. If the equation holds, output *VALID*, otherwise, output *INVALID*.

3.2. *Forgery Attack of Tsai et al.'s Scheme*

Tsai *et al.* alleged that their scheme (Tsai *et al.*, 2014) was secure against Type-I and Type-II adversaries under the standard model. After a careful investigation, however, we found that their scheme was insecure against a Type-II adversary. Then we show a concrete attack instance to demonstrate that the scheme in Tsai *et al.* (2014) is so vulnerable that any malicious-but-passive KGC, $\mathfrak{A}_{II}$, can forge a valid signature of message M^* for identity ID^*. The attack is as follows:

(1) $\mathfrak{A}_{II}$ randomly selects $\alpha, \beta, \gamma \in \mathcal{Z}_p^*$ and calculates $g_2^* = g^\gamma, k'^* = g^\alpha, s'^* = g^\beta$. Besides, $\mathfrak{A}_{II}$ sets $\mathbf{K}^* = [k_i] = [g^{\alpha_i}] \in \mathcal{G}^{n_k}, \mathbf{S}^* = [s_i] = [g^{\beta_i}] \in \mathcal{G}^{n_s}$, where $\alpha_i, \beta_i \in \mathbb{Z}_p^*$. Other parameters in the system master secret key and system public parameters are generated normally by the KGC. Finally, $\mathfrak{A}_{II}$ publishes these public parameters.

(2) By making a hash query on $(\mathsf{PK}^{(1)}, \mathsf{PK}^{(2)}), \mathfrak{A}_{II}$ can obtain the hash value $\nu u, \nu s$. Then $\mathfrak{A}_{II}$ can calculate $k'^* \prod_{i=1}^{n_k} k_i^{\nu u_i} = g^{(\alpha + \Sigma_{i=1}^{n_k} \alpha_i \nu u_i)}, s'^* \prod_{i=1}^{n_s} s_i^{\nu s_i} = g^{(\beta + \Sigma_{i=1}^{n_s} \beta_i \nu s_i)}$.

(3) $\mathfrak{A}_{II}$ randomly picks $a, b, c \in \mathcal{Z}_p^*$ and calculates $\sigma_2^* = g^a, \sigma_3^* = g^b, \sigma_4^* = g^c$.

(4) $\mathfrak{A}_{II}$ calculates $\sigma_1^* = g_1^\gamma \cdot (u' \prod_{i=1}^{n_u} u_i^{\nu_i})^a \cdot (t' \prod_{i=1}^{n_t} t_i^{\nu t_i})^b \cdot (\mathsf{PK}^{(1)})^{(\gamma + \alpha + \Sigma_{i=1}^{n_k} \alpha_i \nu u_i)} \cdot (\mathsf{PK}^{(2)})^{(\beta + \Sigma_{i=1}^{n_s} \beta_i \nu s_i)} \cdot (m' \prod_{i=1}^{n_m} m_i^{\nu m_i})^c$.

(5) The signature on the message M^* is $\sigma^* = (\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*)$.

Anyone can verify the signature σ^* through performing the algorithm RCL-Verify, which is to check whether the equation $\hat{e}(g, \sigma_1^*) \stackrel{?}{=} \hat{e}(g_1, g_2^*) \cdot \hat{e}(\sigma_2^*, u' \prod_{i=1}^{n_u} u_i^{\nu_i}) \cdot \hat{e}(\sigma_3^*, t' \prod_{i=1}^{n_t} t_i^{\nu t_i}) \cdot \hat{e}(\mathsf{PK}^{(1)}, g_2^*(k'^* \prod_{i=1}^{n_k} k_i^{\nu u_i})) \cdot \hat{e}(\mathsf{PK}^{(2)}, s'^* \prod_{i=1}^{n_s} s_i^{\nu s_i}) \cdot \hat{e}(\sigma_4^*, m' \prod_{i=1}^{n_m} m_i^{\nu m_i})$ holds. This verification will hold due to the following fact:

$$
\begin{aligned}
\hat{e}(g, \sigma_1^*) = \hat{e}\Bigg(g, g_1^\gamma \cdot \left(u' \prod_{i=1}^{n_u} u_i^{\nu_i}\right)^a \cdot \left(t' \prod_{i=1}^{n_t} t_i^{\nu t_i}\right)^b \cdot \left(\mathsf{PK}^{(1)}\right)^{(\gamma + \alpha + \Sigma_{i=1}^{n_k} \alpha_i \nu u_i)} \\
\cdot \left(\mathsf{PK}^{(2)}\right)^{(\beta + \Sigma_{i=1}^{n_s} \beta_i \nu s_i)} \cdot \left(m' \prod_{i=1}^{n_m} m_i^{\nu m_i}\right)^c\Bigg) \\
= \hat{e}\left(g_1, g^\gamma\right) \cdot \hat{e}\left(g^a, u' \prod_{i=1}^{n_u} u_i^{\nu_i}\right) \cdot \hat{e}\left(g^b, t' \prod_{i=1}^{n_t} t_i^{\nu t_i}\right) \\
\cdot \hat{e}\left(\mathsf{PK}^{(1)}, g^{(\gamma + \alpha + \Sigma_{i=1}^{n_k} \alpha_i \nu u_i)}\right)
\end{aligned}
$$

$$\cdot \hat{e}\left(\mathsf{PK}^{(2)}, g^{(\beta+\Sigma_{i=1}^{n_s}\beta_i \nu s_i)}\right) \cdot \hat{e}\left(g^c, m' \prod_{i=1}^{n_m} m_i^{\nu m_i}\right)$$

$$= \hat{e}(g_1, g_2^*) \cdot \hat{e}\left(\sigma_2^*, u' \prod_{i=1}^{n_u} u_i^{\nu_i}\right) \cdot \hat{e}\left(\sigma_3^*, t' \prod_{i=1}^{n_t} t_i^{\nu t_i}\right)$$

$$\cdot \hat{e}\left(\mathsf{PK}^{(1)}, g_2^*\left(k'^* \prod_{i=1}^{n_k} k_i^{\nu u_i}\right)\right)$$

$$\cdot \hat{e}\left(\mathsf{PK}^{(2)}, s'^* \prod_{i=1}^{n_s} s_i^{\nu s_i}\right) \cdot \hat{e}\left(\sigma_4^*, m' \prod_{i=1}^{n_m} m_i^{\nu m_i}\right).$$

Notice that $\mathfrak{A}_{II}$ neither knows $sv_{ID} = (x_1, x_2)$ nor replaces $(\mathsf{PK}^{(1)}, \mathsf{PK}^{(2)})$. Thus, the RCLS scheme of Tsai *et al.* cannot withstand the forgery attack mounted by a malicious-but-passive KGC. The underlying reason is that the user secret value sv_{ID} is embedded in the signature inappropriately such that the user public key g^{x_1}, g^{x_2} can be utilized by the malicious-but-passive KGC to forge the signature with the support of the random numbers associated with the system parameters. Specially, $\mathsf{SK}_{ID} = g_2^{x_1}(k' \prod_{i=1}^{n_k} k_i^{\nu u_i})^{x_1}(s' \prod_{i=1}^{n_s} s_i^{\nu s_i})^{x_2}$ has been regarded as one factor in the generation of signature σ_1, and this factor can be easily generated by raising the g^{x_1}, g^{x_2} with the exponentiations $\alpha, \beta, \gamma, \alpha_i, \beta_i$. Therefore, it is easy to forge the signature σ to make the equation hold in the algorithm RCL-Verify.

4. Our Proposed RCLS Scheme

This section describes a revocable certificateless signature scheme without random oracles and presents the concrete construction. Afterwards, this section represents the security analysis, which demonstrates that our proposed RCLS scheme is EUF-CMA-secure in the standard model.

4.1. *Construction*

Inspired by a certificateless signature scheme in the standard model that can resist the malicious-but-passive attacks (Shim, 2018a), we construct a certificateless signature scheme with revocation in the standard model. In the proposed scheme, it is possible to construct a RCLS scheme secure against the Type-I and II attackers as well as the malicious-but-passive attacks by using the term $(g_3^{x_2})^{x_1^{-1}}$. To be specific, although a malicious KGC calculates g_3 as g^{ω} of its own choice ω to implement the malicious-but-passive attack, the malicious KGC cannot calculate $(g_3^{x_2})^{x_1^{-1}}$ without the knowledge of x_1. In other words, if one uses $g_3^{x_2}$ instead of $(g_3^{x_2})^{x_1^{-1}}$, the malicious KGC can calculate $g_3^{x_2}$ by calculating $(g^{x_2})^{\omega}$ from the known user's public key g^{x_2}. In fact, there does not exist a probabilistic polynomial-time algorithm that can calculate $g_3^{x_2}$ with non-negligible advantage

on inputting g, g^{ω}, g^{x_2}. Besides, there does not exist a probabilistic polynomial-time algorithm that can calculate $(g_3^{x_2})^{x_1^{-1}}$ with non-negligible advantage on inputting $g_3, g_3^{x_1}, g_3^{x_2}$, which is equivalent to the CDH problem according to Bao *et al.* (2003). The detail description of the proposed RCLS scheme is presented as follows:

Define three collision-resistant hash functions $\mathcal{H}_0 : \{0,1\}^* \to \{0,1\}^{n_u}$, $\mathcal{H}_1 : \{0,1\}^* \to \{0,1\}^{n_t}$, $\mathcal{H}_2 : \{0,1\}^* \to \{0,1\}^{n_m}$, where n_u, n_t, n_m are fixed lengths from $\mathcal{Z}$.

- Setup: Taken k as the security parameter, KGC generates a bilinear map $\hat{e} : \mathcal{G} \times \mathcal{G} \to \mathcal{G}_T$, where $\mathcal{G}, \mathcal{G}_T$ are cyclic groups of order p. Furthermore, KGC picks $x, y \in \mathcal{Z}_p^*, g_2, g_3 \in \mathcal{G}$ and calculates $g_1 = g^{x+y}, A = \hat{e}(g_1, g_2), \mathsf{SSK} = (g_2^x, g_2^y)$, where g, SSK denote a generator of $\mathcal{G}$ and the system secret key respectively. After that, KGC randomly selects $\alpha, \beta, \gamma \in \mathcal{G}$, and three vectors $\mathbf{U} = [u_i] \in \mathcal{G}^{n_u}, \mathbf{T} = [t_i] \in \mathcal{G}^{n_t}, \mathbf{M} = [m_i] \in \mathcal{G}^{n_m}$. Define three functions f_1, f_2, f_3 via $f_1(\mathcal{U}) = \alpha \prod_{i \in \mathcal{U}} u_i$, $f_2(\mathcal{T}) = \beta \prod_{i \in \mathcal{T}} t_i$, $f_3(\mathcal{M}) = \gamma \prod_{i \in \mathcal{M}} m_i$, where $\mathcal{U} \subseteq \{1, 2, \ldots, n_u\}, \mathcal{T} \subseteq \{1, 2, \ldots, n_t\}, \mathcal{M} \subseteq \{1, 2, \ldots, n_m\}$. Finally, KGC issues the system public parameters $\mathsf{SPP} = \langle \mathcal{G}, \mathcal{G}_T, \hat{e}, g, g_1, g_2, g_3, A, f_1, f_2, f_3, \mathbf{U}, \mathbf{T}, \mathbf{M}, \mathcal{H}_0, \mathcal{H}_1, \mathcal{H}_2 \rangle$.
- Partial-Private-Key-Extraction: After receiving SSK, SPP, and a user's identity ID, KGC will first calculate a set as $\mathcal{F}_{ID} = \{i | u[i] = 1, u = \mathcal{H}_0(ID)\} \subseteq \{1, 2, \ldots, n_u\}$. Then KGC calculates the user's partial private key $\mathsf{PPK}_{ID} = (\mathsf{PPK}^{(1)}, \mathsf{PPK}^{(2)}) = (g_2^x f_1(\mathcal{F}_{ID})^{r_u}, g^{r_u})$ where r_u is randomly selected by KGC from $\mathcal{Z}_p^*$.
- Time-Key-Update: Upon receiving SSK, ID and a time period T, KGC calculates a set as $\mathcal{F}_{ID,t} = \{i | t'[i] = 1, t' = \mathcal{H}_1(ID, t)\} \subseteq \{1, 2, \ldots, n_t\}$, and sets the time key $\mathsf{TK}_T = (\mathsf{TK}^{(1)}, \mathsf{TK}^{(2)}) = (g_2^y f_2(\mathcal{F}_{ID,t})^{r_t}, g^{r_t})$ where r_t is randomly selected by KGC from $\mathcal{Z}_p^*$.
- Secret-Value-Generation: A user with identity ID randomly picks $x_1, x_2 \in \mathcal{Z}_p^*$ and sets the secret value $sv_{ID} = (x_1, x_2)$.
- Public-Key-Generation: The user ID calculates $\mathsf{PK}_{ID} = (\mathsf{PK}^{(1)}, \mathsf{PK}^{(2)}) = (g^{x_1}, g^{x_2})$ as the public key.
- Secret-Key-Generation: The user ID randomly selects $\lambda, \mu \in \mathcal{Z}_p^*$ and calculates the secret key $\mathsf{SK}_{ID} = (\mathsf{SK}^{(1)}, \mathsf{SK}^{(2)}, \mathsf{SK}^{(3)}) = ((\mathsf{PPK}^{(1)} \cdot \mathsf{TK}^{(1)} \cdot f_1(\mathcal{F}_{ID})^{\lambda} \cdot f_2(\mathcal{F}_{ID,t})^{\mu} \cdot g_3^{x_2})^{x_1^{-1}}, \mathsf{PPK}^{(2)} \cdot g^{\lambda}, \mathsf{TK}^{(2)} \cdot g^{\mu})$.
- RCL-Sign: Upon receiving SPP and PK_{ID}, a signer ID can sign a message $M \in \{0,1\}^*$ with a secret key SK_{ID} and signer's secret value sv_{ID} by performing the following steps:

  (1) Define a set as $\mathcal{F}_M = \{i | m[i] = 1, m = \mathcal{H}_2(M, ID, \mathsf{PK}_{ID})\} \subseteq \{1, 2, \ldots, n_m\}$.
  (2) Randomly select $v \in \mathcal{Z}_p^*$ and calculate $\sigma_1 = \mathsf{SK}^{(1)} \cdot (f_3(\mathcal{F}_M)^v)^{x_1^{-1}}$, $\sigma_2 = \mathsf{SK}^{(2)}, \sigma_3 = \mathsf{SK}^{(3)}, \sigma_4 = g^v$.
  (3) Output a revocable certificateless signature $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4)$ of the message M and return to a verifier.

- RCL-Verify: Given $\mathsf{SPP}, ID, \sigma, M, t, \mathsf{PK}_{ID}$, any verifier can check the equation $\hat{e}(\sigma_1, \mathsf{PK}^{(1)}) \stackrel{?}{=} A \cdot \hat{e}(f_1(\mathcal{F}_{ID}), \sigma_2) \cdot \hat{e}(f_2(\mathcal{F}_{ID,t}), \sigma_3) \cdot \hat{e}(f_3(\mathcal{F}_M), \sigma_4) \cdot \hat{e}(g_3, \mathsf{PK}^{(2)})$. If the equation holds, output *VALID*, otherwise, output *INVALID*.

4.2. *Security Analysis*

Theorem 1. *The proposed scheme is* $(t, q_{PPK}, q_{TK}, q_S, \epsilon)$*-EUF-CMA-secure in* ***Game I*** *defined in Section* 2 *if the* (t', ϵ')*-CDH assumption holds in* $\mathcal{G}$*, where*

$$\epsilon' \leqslant \frac{\epsilon}{16(q_{PPK} + q_S)(n_u + 1)(n_m + 1)q_S},$$
$$t' \approx t + O\big(n_u \cdot q_{PPK} + n_t \cdot q_{TK} + (n_u + n_t + n_m) \cdot q_S\big).$$

Proof. Assume that $\mathfrak{A}_I$ is a Type-I attacker against the proposed scheme. There is a t'-time algorithm $\mathfrak{B}$ that can solve the CDH problem with advantage at least ϵ' by interacting with $\mathfrak{A}_I$.

Let g be a generator of $\mathcal{G}$, g^a, g^b be two elements of $\mathcal{G}$ where $a, b \in \mathcal{Z}_p^*$. The algorithm $\mathfrak{B}$ can compute g^{ab} as the solution of CDH problem by simulating a challenger for $\mathfrak{A}_I$.

Setup: $\mathfrak{B}$ sets $l_u = 2(q_{PPK} + q_S), l_m = 2q_S$. Suppose that $l_u(n_u + 1) \leqslant p, l_m(n_m + 1) \leqslant p$. Next $\mathfrak{B}$ randomly selects two integers k_u, k_m with $0 \leqslant k_u \leqslant n_u, 0 \leqslant k_m \leqslant n_m$. Afterwards, the following integers are selected by $\mathfrak{B}$:

$$x' \in \mathcal{Z}_{l_u}, \qquad z' \in \mathcal{Z}_{l_m}, \qquad y', v', w' \in \mathcal{Z}_p^*,$$
$$\mathbf{X} = [x_i] \in \mathcal{Z}_{l_u}^{n_u}, \qquad \mathbf{Y} = [y_i] \in \mathcal{Z}_p^{n_u},$$
$$\mathbf{V} = [v_i] \in \mathcal{Z}_p^{n_t}, \qquad \mathbf{Z} = [z_i] \in \mathcal{Z}_{l_m}^{n_m}, \qquad \mathbf{W} = [w_i] \in \mathcal{Z}_p^{n_m}.$$

Besides, $\mathfrak{B}$ defines five functions for $u = \mathcal{H}_0(ID), ut = \mathcal{H}_1(ID, t), m = \mathcal{H}_2(M, ID, \mathsf{PK}_{ID})$ as follows:

$$F_1(u) = x' + \sum_{i \in \mathcal{U}} x_i - l_u k_u, \qquad J_1(u) = y' + \sum_{i \in \mathcal{U}} y_i, \quad \text{where } \mathcal{U} \subseteq \{1, 2, \ldots, n_u\},$$
$$J_2(ut) = v' + \sum_{i \in \mathcal{T}} v_i, \quad \text{where } \mathcal{T} \subseteq \{1, 2, \ldots, n_t\},$$
$$F_2(m) = z' + \sum_{i \in \mathcal{M}} z_i - l_m k_m, \; J_3(m) = w' + \sum_{i \in \mathcal{M}} w_i, \quad \text{where } \mathcal{M} \subseteq \{1, 2, \ldots, n_m\}.$$

After that, $\mathfrak{B}$ randomly selects $\alpha \in \mathcal{Z}_p^*$, then sets the following parameters

$$g_1 = g^a g^\alpha, \qquad g_2 = g^b,$$
$$u' = g_2^{-l_u k_u + x'} g^{y'}, \qquad u_i = g_2^{x_i} g^{y_i} \quad (1 \leqslant i \leqslant n_u),$$
$$t' = g^{v'}, \qquad t_i = g^{v_i} \quad (1 \leqslant i \leqslant n_t),$$
$$m' = g_2^{-l_m k_m + z'} g^{w'}, \qquad m_i = g_2^{z_i} g^{w_i} \quad (1 \leqslant i \leqslant n_m)$$

and constructs the following equations:

$$f_1(\mathcal{U}) = u' \prod_{i \in \mathcal{U}} u_i = g_2^{F_1(u)} g^{J_1(u)}, \qquad f_2(\mathcal{T}) = t' \prod_{i \in \mathcal{T}} t_i = g^{J_2(ut)},$$

$$f_3(\mathcal{M}) = m' \prod_{i \in \mathcal{M}} m_i = g_2^{F_2(m)} g^{J_3(m)}.$$

Query: Attacker $\mathfrak{A}_I$ performs queries adaptively as following:

- *Public-Key-Extract Query*: At first, $\mathfrak{B}$ maintains a list $L_{PK} = \{(ID, sv_{ID}, \mathsf{PK}_{ID})\}$ in order to respond to these queries. When an identity ID is supplied to this oracle, $\mathfrak{B}$ inspects the list L_{PK}:
  (1) If the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ exists in L_{PK}, $\mathfrak{B}$ answers to $\mathfrak{A}_I$ with PK_{ID}.
  (2) If the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ doesn't exist in L_{PK}, $\mathfrak{B}$ randomly picks $x_1, x_2 \in \mathcal{Z}_p^*$, calculates g^{x_1}, g^{x_2}, and sets $sv_{ID} = (x_1, x_2), \mathsf{PK}_{ID} = (g^{x_1}, g^{x_2})$. After that, $\mathfrak{B}$ answers to $\mathfrak{A}_I$ with PK_{ID} and inserts the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ into L_{PK}.
- *Partial-Private-Key-Extract Query*: At first, $\mathfrak{B}$ maintains a list $L_{PPK} = \{(ID, \mathsf{PPK}_{ID})\}$ in order to respond to these queries. When an identity ID is supplied to this oracle, $\mathfrak{B}$ inspects the list L_{PPK}:
  (1) If the tuple (ID, PPK_{ID}) exists in L_{PPK}, $\mathfrak{B}$ answers to $\mathfrak{A}_I$ with PPK_{ID}.
  (2) If the tuple (ID, PPK_{ID}) doesn't exist in L_{PPK}, and $F_1(u) \neq 0 \bmod p$, $\mathfrak{B}$ randomly picks $r_u \in \mathcal{Z}_p^*$ and calculates

$$\mathsf{PPK}_{ID} = \left(\mathsf{PPK}^{(1)}, \mathsf{PPK}^{(2)}\right) = \left(\left(\frac{g_1}{g^\alpha}\right)^{-\frac{J_1(u)}{F_1(u)}} f_1^{r_u}(\mathcal{U}), \left(\frac{g_1}{g^\alpha}\right)^{-\frac{1}{F_1(u)}} g^{r_u}\right).$$

  After that, $\mathfrak{B}$ answer to $\mathfrak{A}_I$ with PPK_{ID} and inserts the tuple (ID, PPK_{ID}) into L_{PPK}.
  (3) Otherwise, $\mathfrak{B}$ outputs "failure" and discontinues.
- *Time-Key-Update Query*: When a tuple (ID, T) is supplied to this oracle, $\mathfrak{B}$ randomly selects $r_t \in \mathcal{Z}_p^*$ and calculates the time key $\mathsf{TK}_T = (\mathsf{TK}^{(1)}, \mathsf{TK}^{(2)}) = (g_2^\alpha f_2^{r_t}(\mathcal{T}), g^{r_t})$.
- *Secret-Value-Extract Query*: When an identity ID is supplied to this oracle, $\mathfrak{B}$ inspects the list L_{PK}:
  (1) If the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ exists in L_{PK}, $\mathfrak{B}$ answers to $\mathfrak{A}_I$ with sv_{ID}.
  (2) If the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ doesn't exist in L_{PK}, $\mathfrak{B}$ makes a *Public-Key-Extract Query* with ID to obtain $(\mathsf{PK}_{ID}, sv_{ID})$. After that, $\mathfrak{B}$ updates $(\mathsf{PK}_{ID}, sv_{ID})$ into L_{PK} and answers to $\mathfrak{A}_I$ with sv_{ID}.
- *Public-Key-Replace Query*: When an identity (ID, PK'_{ID}) is supplied to this oracle, $\mathfrak{B}$ inspects the list L_{PK}:

(1) If the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ exists in L_{PK}, $\mathfrak{B}$ sets $\mathsf{PK}_{ID} = \mathsf{PK}'_{ID}, sv_{ID} = sv'_{ID}$ and then updates $(\mathsf{PK}_{ID}, sv_{ID})$ into L_{PK}.

(2) If the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ doesn't exist in L_{PK}, $\mathfrak{B}$ first makes a *Public-Key-Extract Query* with ID to obtain $(\mathsf{PK}_{ID}, sv_{ID})$. And then $\mathfrak{B}$ sets $\mathsf{PK}_{ID} = \mathsf{PK}'_{ID}, sv_{ID} = sv'_{ID}$. After that, $\mathfrak{B}$ updates $(\mathsf{PK}_{ID}, sv_{ID})$ into L_{PK}.

- *RCL-Sign Query*: When the tuple (M, ID, T) is supplied to this oracle, $\mathfrak{B}$ inspects the list L_{PK}:

(1) If the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ exists in L_{PK}, $\mathfrak{B}$ retrieves the list L_{PPK}:

(i) If the tuple (ID, PPK_{ID}) exists in L_{PPK}, $\mathfrak{B}$ produces a signature $\sigma \leftarrow \mathsf{RCL\text{-}Sign}(sv_{ID}, \mathsf{PPK}_{ID}, M)$ by running the algorithm RCL-Sign.

(ii) If the tuple (ID, PPK_{ID}) doesn't exist in L_{PPK}, and $F_1(u) \neq 0 \bmod l_u$, $\mathfrak{B}$ makes a *Partial-Private-Key-Extract Query* to get (ID, PPK_{ID}), and then produces a signature by running the algorithm RCL-Sign.

(iii) If the tuple (ID, PPK_{ID}) doesn't exist in L_{PPK}, and $F_1(u) = 0 \bmod l_u$, $\mathfrak{B}$ calculates $F_2(m) \bmod l_m$:

① If $F_2(m) \neq 0 \bmod l_m$, $\mathfrak{B}$ selects $\lambda, \mu, \nu \in \mathcal{Z}_p^*$ and calculates

$$\begin{aligned}\sigma &= (\sigma_1, \sigma_2, \sigma_3, \sigma_4)\\ &= \left(\left[g_2^{\alpha} f_1^{\lambda}(\mathcal{U}) f_2^{\mu}(\mathcal{T}) \left(\frac{g_1}{g^{\alpha}}\right)^{-\frac{J_3(m)}{F_2(m)}} f_3^{\nu}(\mathcal{M}) g_3^{x_2}\right]^{x_1^{-1}},\right.\\ &\quad \left. g^{\lambda}, g^{\mu}, g_1^{-\frac{1}{F_2(m)}} g^{\nu}\right)\\ &= \left(\left[g_2^{a+\alpha} f_1^{\lambda}(\mathcal{U}) f_2^{\mu}(\mathcal{T}) f_3^{\nu'}(\mathcal{M}) g_3^{x_2}\right]^{x_1^{-1}}, g^{\lambda}, g^{\mu}, g^{\nu'}\right),\end{aligned}$$

where $\nu' = \nu - \frac{a}{F_2(m)}$.

② If $F_2(m) = 0 \bmod l_m$, $\mathfrak{B}$ outputs "failure" and discontinues.

(iv) Otherwise, $\mathfrak{B}$ outputs "failure" and discontinues.

(2) If the tuple $(ID, sv_{ID}, \mathsf{PK}_{ID})$ doesn't exist in L_{PK}, $\mathfrak{B}$ makes a *Public-Key-Extract Query* with ID and then repeats step **(1)**.

Forgery: After finishing all queries and $\mathfrak{B}$ doesn't discontinue, $\mathfrak{A}_I$ outputs $u^* = \mathcal{H}_0(ID^*), ut^* = \mathcal{H}_1(ID^*, t^*), m^* = \mathcal{H}_2(M^*, ID^*, \mathsf{PK}_{ID^*})$, and generates a forgery $\sigma^* = (\sigma_1^*, \sigma_2^*, \sigma_3^*, \sigma_4^*)$. Iff $F_1(u^*) = 0 \bmod p$ and $F_2(m^*) = 0 \bmod p$, $\mathfrak{B}$ calculates

$$\begin{aligned}&\frac{(\sigma_1^*)^{x_1^*}}{(\sigma_2^*)^{J_1(u^*)}(\sigma_3^*)^{J_2(ut^*)}(\sigma_4^*)^{J_3(m^*)} g_2^{\alpha} g_3^{x_2^*}} = \frac{([g_2^{a+\alpha} f_1^{\lambda}(\mathcal{U}) f_2^{\mu}(\mathcal{T}) f_3^{\nu'}(\mathcal{M}) g_3^{x_2^*}]^{x_1^{*-1}})^{x_1^*}}{g^{J_1(u^*)\lambda} g^{J_2(ut^*)\mu} g^{J_3(m^*)\nu'} g_2^{\alpha} g_3^{x_2^*}}\\ &= \frac{g_2^{a+\alpha}(g_2^{F_1(u^*)} g^{J_1(u^*)})^{\lambda}(g^{J_2(ut^*)})^{\mu}(g_2^{F_2(m^*)} g^{J_3(m^*)})^{\nu'} g_3^{x_2^*}}{g^{J_1(u^*)\lambda} g^{J_2(ut^*)\mu} g^{J_3(m^*)\nu'} g_2^{\alpha} g_3^{x_2^*}}\\ &= g_2^a = g^{ab},\end{aligned}$$

g^{ab} is the solution of the CDH problem.

Now we analyse the probability that $\mathfrak{B}$ can solve the given CDH problem instance. At first, let $u_1, \ldots, u_{q_U}$ be the $\mathcal{H}_0$'s result that appears in either *Partial-Private-Key-Extract Query* or in *RCL-Sign Query* but not including the algorithm's identity ID^*. Let $ut_1, \ldots, ut_{q_T}$ be the $\mathcal{H}_1$'s result that appears in *Time-Key-Update Query*. Let $m_1, \ldots, m_{q_M}$ be the $\mathcal{H}_2$'s result that appears in *RCL-Sign Query* including all identity ID. Obviously, there are $q_U \leqslant q_{PPK} + q_S, q_T \leqslant q_{TK}$ and $q_M \leqslant q_S$. Next, we define the following events for simplifying the probability analysis.

(1) E_i $(i = 1, \ldots, q_U)$: $F_1(u_i) \neq 0 \bmod l_u$, in other words, $\mathfrak{B}$ does not discontinue in the $\mathfrak{A}_I$'s *Partial-Private-Key-Extract Query*.
(2) E^*: $F_1(u^*) = 0 \bmod p$.
(3) E'_i $(i = 1, \ldots, q_M)$: $F_2(m_i) \neq 0 \bmod l_m$, in other words, $\mathfrak{B}$ does not discontinue in the $\mathfrak{A}_I$'s *RCL-Sign Query*.
(4) E'^*: $F_2(m^*) = 0 \bmod p$.
(5) E^*_S: $F_1(u^*) = 0 \bmod p$, $F_2(m^*) = 0 \bmod p$, in other words, $\mathfrak{A}_I$ produces a valid signature.

The probability that $\mathfrak{B}$ does not discontinue is

$$\begin{aligned}\Pr[\text{success}] &\geqslant \Pr\left[\left(\bigwedge_{i=1}^{q_U} E_i \wedge E^*\right) \wedge \left(\bigwedge_{i=1}^{q_M} E'_i \wedge E'^*\right) \wedge E^*_S\right] \\ &= \Pr[E^*] \cdot \Pr\left[\bigwedge_{i=1}^{q_U} E_i | E^*\right] \cdot \Pr[E'^*] \cdot \Pr\left[\bigwedge_{i=1}^{q_M} E'_i | E'^*\right] \cdot \Pr[E^*_S],\end{aligned}$$

$\because l_u = 2(q_{PPK} + q_S),\ l_m = 2q_S,\ l_u(n_u + 1) \leqslant p,\ l_m(n_m + 1) \leqslant p$

$$\begin{aligned}\therefore \Pr[E^*] &= \Pr[F_1(u^*) = 0 \bmod p \wedge F_1(u^*) = 0 \bmod l_u] \\ &= \Pr[F_1(u^*) = 0 \bmod l_u] \cdot \Pr[F_1(u^*) = 0 \bmod p | F_1(u^*) = 0 \bmod l_u] \\ &= \frac{1}{l_u} \cdot \frac{1}{n_u + 1}\end{aligned}$$

$$\begin{aligned}\Pr[E'^*] &= \Pr[F_2(m^*) = 0 \bmod p \wedge F_2(m^*) = 0 \bmod l_m] \\ &= \Pr[F_2(m^*) = 0 \bmod l_m] \cdot \Pr[F_2(m^*) = 0 \bmod p | F_2(m^*) = 0 \bmod l_m] \\ &= \frac{1}{l_m} \cdot \frac{1}{n_m + 1},\end{aligned}$$

$$\begin{aligned}\Pr\left[\bigwedge_{i=1}^{q_U} E_i | E^*\right] &= 1 - \Pr\left[\bigvee_{i=1}^{q_U} \neg E_i | E^*\right] \geqslant 1 - \sum_{i=1}^{q_U} \Pr\left[\neg E_i | E^*\right] \\ &= 1 - \frac{q_U}{l_u} \geqslant 1 - \frac{q_{PPK} + q_S}{l_u}, \\ \Pr\left[\bigwedge_{i=1}^{q_M} E'_i | E'^*\right] &= 1 - \Pr\left[\bigvee_{i=1}^{q_M} \neg E'_i | E'^*\right] \geqslant 1 - \sum_{i=1}^{q_M} \Pr\left[\neg E'_i | E'^*\right]\end{aligned}$$

$$= 1 - \frac{q_M}{l_m} \geqslant 1 - \frac{q_S}{l_m}$$

$$\Longrightarrow \Pr[\text{success}] \geqslant \frac{\epsilon}{16(q_{PPK} + q_S)(n_u + 1)(n_m + 1)q_S}.$$

Therefore, the probability that $\mathfrak{B}$ can solve the given CDH problem instance is

$$\epsilon' \leqslant \frac{\epsilon}{16(q_{PPK} + q_S)(n_u + 1)(n_m + 1)q_S}. \qquad \square$$

Theorem 2. *The proposed scheme is* $(t, q_{PPK}, q_{TK}, q_S, \epsilon)$*-EUF-CMA-secure in* ***Game II*** *defined in Section* 2 *if the* (t', ϵ')*-CDH assumption holds in* $\mathcal{G}$*, where*

$$\epsilon' \leqslant \frac{\epsilon}{4(n_m + 1)q_S},$$

$$t' \approx t + O\big(n_u \cdot q_{PPK} + n_t \cdot q_{TK} + (n_u + n_t + n_m) \cdot q_S\big).$$

Proof. The proof of **Theorem 2** is similar to that of **Theorem 1** and the detail of the proof is omitted here. □

Theorem 3. *The proposed scheme is* $(t, q_{PPK}, q_{TK}, q_S, \epsilon)$*-EUF-CMA-secure in* ***Game III*** *defined in Section* 2 *if the* (t', ϵ')*-CDH assumption holds in* $\mathcal{G}$*, where*

$$\epsilon' \leqslant \frac{\epsilon}{16(q_{PPK} + q_S)(n_u + 1)(n_m + 1)q_S},$$

$$t' \approx t + O\big(n_u \cdot q_{PPK} + n_t \cdot q_{TK} + (n_u + n_t + n_m) \cdot q_S\big).$$

Proof. The proof of **Theorem 3** is similar to that of **Theorem 1** and the detail of the proof is omitted here. □

5. Performance Evaluation

This section investigates the property of the proposed scheme and its performance in terms of computational and communication overhead.

Table 1 demonstrates the properties of different schemes in respect of the security level, revocation mechanism, security model and security assumption for the existing RCLS schemes (Tsai *et al.*, 2014; Xiong and Qin, 2015; Hung *et al.*, 2016; Zheng *et al.*, 2017) and our proposed scheme. Here, the symbol "✓" represents that the scheme satisfies the property and "×" represents that the scheme does not satisfy the property. "ROM" and "SM" denote that the security model is the random oracle model and the standard model, respectively. Obviously, our proposed scheme holds all properties. Especially, our proposed RCLS scheme is provably secure in the standard model and resists the attack mounted by the malicious-but-passive KGC.

Table 2 shows theoretical evaluation of the signature size, signing and verification cost. In the computational overhead comparison, there are five operations that are considered:

Table 1
Property comparisons of different RCLS schemes.

Scheme	Tsai et al.	Xiong and Qin	Huang et al.	Zheng et al.	Our scheme
Security against $\mathfrak{A}_I$	✓	×	✓	✓	✓
Security against $\mathfrak{A}_{II}$	×	✓	✓	✓	✓
Security against $\mathfrak{A}_{ru}$	✓	×	✓	×	✓
Revocable	✓	✓	✓	✓	✓
Security model	SM	ROM	ROM	ROM	SM
Security assumption	CDH	CDH	CDH	CDH	CDH

Table 2
Computation overhead comparisons of different RCLS schemes.

Scheme	Signature size	Signing cost	Verification cost
Tsai et al.	$4\|\mathcal{G}\|$	$1SM + (\frac{n_m}{2} + 1)M$	$7P + (\frac{n_u+n_t+n_k+n_s+n_m}{5} + 5)M$
Xiong and Qin	$3\|\mathcal{G}\|$	$5SM$	$5P + 3H$
Hung et al.	$1\|\mathcal{G}\|$	$2SM + 2M$	$4P + 3H$
Zheng et al.	$6\|\mathcal{G}\|$	$4SM + 2M$	$9P + 6H + 2SM + 3E$
Our scheme	$4\|\mathcal{G}\|$	$3SM + (\frac{n_m}{2} + 1)M$	$5P + (\frac{n_u+n_t+n_m}{3} + 3)M$

pairing, scalar multiplication, multiplication in $\mathcal{G}$, exponentiation in $\mathcal{G}_T$ and hash operations, which are denoted by P, SM, M, E and H respectively. Especially, it is known that the pairing operation and the scalar multiplication on a curve make up the major part of the computational complexity. In the communication overhead comparison, signature size is measured with respect to the number of group elements. From Table 2, it is readily to observe that Hung *et al.'* scheme (Hung *et al.*, 2016) has better performance than the schemes in Tsai *et al.* (2014), Xiong and Qin (2015), Zheng *et al.* (2017) and our presented scheme in terms of computation and communication overhead. However, the scheme of Hung *et al.* (2016) was constructed in the random oracle model. Although the scheme of Tsai *et al.* (2014) was constructed without random oracles same as our scheme, their scheme needs more verification cost and cannot resist the malicious-but-passive attacks. Therefore, our proposed scheme is more secure and efficient in actual life.

6. Conclusion

This paper first analyses a certificateless signature scheme with revocation of Tsai *et al.* (2014), which was proved to be secure without random oracles. However, this paper demonstrates that their scheme is insecure against the malicious-but-passive attacks. Considering that there does not exist a secure RCLS scheme under the standard model at present, this paper constructs a new and provably secure RCLS scheme without random oracles.

To explain readily the proposed revocable certificateless signature scheme, this paper formalizes the RCLS scheme's definition and security model. Furthermore, a concrete RCLS construction scheme is given, whose security analysis is proved in the standard model with CDH assumption. Compared to the existing solutions, the RCLS proposed in this paper is more efficient and secure.

Acknowledgements. This work was supported in part by the 13th Five-Year Plan of National Cryptography Development Fund for Cryptographic Theory of China under Grant MMJJ20170204, in part by the Fundamental Research Funds for the Central Universities under Grant ZYGX2016J091, the Guangxi Colleges and Universities Key Laboratory of Cloud Computing and Complex Systems, the Sichuan Science and Technology Project under Grant 2018KZ0007 and in part by the Natural Science Foundation of China under Grants U1401257, 61472064 and 61602096.

References

Al-Riyami, S.S., Paterson, K.G. (2003). Certificateless public key cryptography. *Advances in Cryptology – ASIACRYPT 2003, 9th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 452–473.

Bao, F., Deng, R.H., Zhu, H. (2003). Variations of diffie-hellman problem. In: *International Conference on Information and Communications Security*. Springer, pp. 301–312.

Boneh, D., Ding, X., Tsudik, G., Wong, C.M. (2001). A method for fast revocation of public key certificates and security capabilities. In: *USENIX Security Symposium*, pp. 22–22.

Canetti, R., Goldreich, O., Halevi, S. (2004). The random oracle methodology, revisited. *Journal of the ACM*, 51(4), 557–594.

Chow, S.S., Boyd, C., Nieto, J.M.G. (2006). Security-mediated certificateless cryptography. In: *Public Key Cryptography – PKC 2006, 9th International Conference on Theory and Practice of Public-Key Cryptography,2006, Proceedings*. Springer, pp. 508–524.

ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4), 469–472.

He, D., Chen, J., Zhang, R. (2012). An efficient and provably-secure certificateless signature scheme without bilinear pairings. *International Journal of Communication Systems*, 25(11), 1432–1442.

Housley, R., Polk, W., Ford, W., Solo, D. (2002). *Internet X. 509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*.

Huang, X., Mu, Y., Susilo, W., Wong, D.S., Wu, W. (2007). Certificateless signature revisited. In: *Proceedings of Information Security and Privacy, 12th Australasian Conference, ACISP 2007*, pp. 308–322.

Huang, X., Susilo, W., Mu, Y., Zhang, F. (2005). On the security of certificateless signature schemes from asiacrypt 2003. In: *Proceedings of Cryptology and Network Security, 4th International Conference, CANS 2005*, pp. 13–25.

Hung, Y.H., Tseng, Y.M., Huang, S.S. (2016). A revocable certificateless short signature scheme and its authentication application. *Informatica*, 27(3), 549–572.

Jia, X., He, D., Liu, Q., Choo, K.K.R. (2018). An efficient provably-secure certificateless signature scheme for internet-of-things deployment. *Ad Hoc Networks*, 71(5), 78–87.

Ju, H.S., Kim, D.Y., Lee, D.H., Lim, J., Chun, K. (2005). Efficient revocation of security capability in certificateless public key cryptography. In: *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer, pp. 453–459.

Karati, A., Islam, S.H., Biswas, G.P. (2018a). A pairing-free and provably secure certificateless signature scheme. *Information Sciences*, 450, 378–391.

Karati, A., Islam, S.H., Karuppiah, M. (2018b). Provably secure and lightweight certificateless signature scheme for IIoT environments. *IEEE Transactions on Industrial Informatics*, 14(8), 3701–3711.

Liu, J.K., Au, M.H., Susilo, W. (2007). Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model: extended abstract. In: *ACM Symposium on Information, Computer and Communications Security*, pp. 273–283.

Shamir, A. (1984). Identity-based cryptosystems and signature schemes. In: *Workshop on the Theory and Application of Cryptographic Techniques*. Springer, pp. 47–53.

Shen, L., Zhang, F., Sun, Y. (2013). Efficient revocable certificateless encryption secure in the standard model. *The Computer Journal*, 57(4), 592–601.

Shim, K.A. (2018a). A new certificateless signature scheme provably secure in the standard model. *IEEE Systems Journal*, 99, 1–10.

Shim, K.A. (2018b). Comments on "Revocable and scalable certificateless remote authentication protocol with anonymity for wireless body area networks". *IEEE Transactions on Information Forensics and Security*. doi:10.1109/TIFS.2018.2871761.

Sun, Y., Zhang, F., Fu, A. (2018). Revocable certificateless encryption with ciphertext evolution. *Australasian Conference on Information Security and Privacy*, 27(3), 741–749.

Sun, Y., Zhang, F., Shen, L., Deng, R.H. (2014). A revocable certificateless signature scheme. *JCP*, 9(8), 1843–1850.

Tsai, T.T., Huang, S.S., Tseng, Y.M. (2014). Secure certificateless signature with revocation in the standard model. *Mathematical Problems in Engineering*, 2014(11), 1–16.

Tsai, T.T., Tseng, Y.M. (2015). Revocable certificateless public key encryption. *Information Sciences*, 9(4), 824–833.

Xia, Q., Xu, C.X., Yu, Y. (2010). Key replacement attack on two certificateless signature schemes without random oracles. *Key Engineering Materials*, 439, 1606–1611.

Xiong, H., Mei, Q., Zhao, Y. (2019). Efficient and provably secure certificateless parallel key-insulated signature without pairing for IIoT environments. *IEEE Systems Journal*. doi:10.1109/JSYST.2018.2890126.

Xiong, H., Qin, Z. (2015). Revocable and scalable certificateless remote authentication protocol with anonymity for wireless body area networks. *IEEE Transactions on Information Forensics and Security*, 10(7), 1442–1455.

Xiong, H., Qin, Z., Li, F. (2008). An improved certificateless signature scheme secure in the standard model. *Fundamenta Informaticae*, 88(1–2), 193–206.

Yap, W.S., Chow, S.S., Heng, S.H., Goi, B.M. (2007). Security mediated certificateless signatures. In: *Applied Cryptography and Network Security*. Springer, pp. 459–477.

Yap, W.S., Heng, S.H., Goi, B.M. (2006). An efficient certificateless signature scheme. In: *Proceedings of Emerging Directions in Embedded and Ubiquitous Computing, EUC 2006*, pp. 322–331.

Yu, Y., Mu, Y., Wang, G., Xia, Q., Yang, B. (2012). Improved certificateless signature scheme provably secure in the standard model. *IET Information Security*, 6(2), 102–110.

Yuan, Y., Li, D., Tian, L., Zhu, H. (2009). Certificateless signature scheme without random oracles. In: *International Conference on Information Security and Assurance*. Springer, pp. 31–40.

Zhang, Z., Wong, D.S., Xu, J., Feng, D. (2006). Certificateless public-key signature: security model and efficient construction. iN: *Proceedings of Applied Cryptography and Network Security, 4th International Conference, ACNS 2006*, pp. 293–308.

Zhang, J., Zhao, X. (2015). An efficient revocable certificateless signature scheme. In: *Fuzzy Systems and Knowledge Discovery , 12th International Conference, FSKD 2015*. IEEE, pp. 1852–1857.

Zheng, Q., Li, Q., Azgin, A., Weng, J. (2017). Data verification in information-centric networking with efficient revocable certificateless signature. In: *2017 IEEE Conference on Communications and Network Security, CNS*. IEEE, pp. 1–9.

Q. Mei is currently pursuing her PhD degree from the School of Information and Software Engineering, University of Electronic Science and Technology of China. She received her BS degree from Jiangxi University of Science and Technology, in 2017. Her research interests include certificateless public key cryptography and key-insulated mechanism.

Y. Zhao is currently pursuing her MS degree from the School of Information and Software Engineering, University of Electronic Science and Technology of China. She received her BS degree from Jiangxi University of Science and Technology, in 2017. Her research interests include identity-based public key cryptography.

H. Xiong received his PhD degree from University of Electronic Science and Technology of China (UESTC), in 2009. He is now a full-time professor in the UESTC. His research interests include public key cryptography and networks security.