INFORMATICA, 2015, Vol. 26, No. 4, 663–684
© 2015 *Vilnius University*
DOI: http://dx.doi.org/10.15388/Informatica.2015.70

Certificateless Signature with Strong Unforgeability in the Standard Model

Ying-Hao HUNG, Sen-Shan HUANG, Yuh-Min TSENG*,
Tung-Tso TSAI

Department of Mathematics, National Changhua University of Education
Jin-De Campus, Chang-Hua City 500, Taiwan
e-mail: ymtseng@cc.ncue.edu.tw

Received: November 2014; accepted: January 2015

Abstract. Certificateless public-key systems (CL-PKS) were introduced to simultaneously solve two critical problems in public-key systems. One is the key escrow problem in ID-based public-key systems and the other is to eliminate the presence of certificates in conventional public-key systems. In the last decade, several certificateless signature (CLS) schemes have been proposed in the random oracle model. These CLS schemes possess existential unforgeability against adaptive chosen-message attacks, and only few of them possess strong unforgeability. A CLS scheme with strong unforgeability plays an important role in the construction of certificateless cryptographic schemes. Unfortunately, all the existing CLS schemes in the standard model (without random oracles) have been shown insecure to provide existential unforgeability under a generally adopted security model. In the article, we propose a strongly secure CLS scheme in the standard model under the generally adopted security model. Our scheme possesses not only existential unforgeability but also strong unforgeability, and turns out to be the *first* strongly secure CLS scheme in the standard model. Under the collision resistant hash (CRH) and computational Diffie–Hellman (CDH) assumptions, we prove that our CLS scheme possesses strong unforgeability against both Type I (outsiders) and Type II (key generation center) adversaries.

Key words: certificateless signature, strong unforgeability, random oracle.

1. Introduction

Shamir (1984) proposed a prominent opinion for public-key cryptography, called identity (ID)-based public-key cryptography (ID-PKC), to simplify public-key management. The first practical ID-based cryptographic scheme (i.e., ID-based encryption) with bilinear maps is constructed by Boneh and Franklin (2001). Afterward, the design of ID-based cryptographic mechanisms has undergone quite rapid progress, and enormous literatures have been presented such as (Waters, 2005; Tseng and Tsai, 2012; Tseng *et al.*, 2014; Tsai *et al.*, 2012, 2014b, 2014c). In ID-PKC setting, the public key of a user is the combination of her/his name, e-mail address, social security number, IP address or other identity information while the private key of the user is generated and issued securely by a trusted

[*] Corresponding author.

third party called private key generator (PKG). ID-PKC eliminates certificate management needed in conventional public-key cryptography. However, ID-PKC suffers from the key escrow problem in the sense that the PKG knows the private key of every user so that the PKG can decrypt ciphertexts or sign messages on behalf of any user.

To resolve the key escrow problem in ID-PKC, Al-Riyami and Paterson (2003) devised a new paradigm called certificateless public-key cryptography (CL-PKC). In CL-PKC setting, a semi-trusted third party, called key generation center (KGC), generates the partial private keys of users. The full private key of a user consists of a partial private key generated by the KGC and a secret key chosen randomly by the user. Meanwhile, the public key of the user is generated by using the secret key, and is published. The KGC has no access to the full private key of any user since the secret key is generated randomly by the user herself/himself. Hence, the key escrow problem is resolved. Subsequently, enormous CL-PKC schemes have been proposed such as certificateless public-key encryption (CL-PKE) (Libert and Quisquater, 2006; Dent, 2008; Yang and Tan, 2011) and certificateless signature (CLS) (Yum and Lee, 2004; Huang *et al.*, 2005, 2007; Hu *et al.*, 2006; Zhang and Zhang, 2008; He *et al.*, 2012; Tso *et al.*, 2012; Tsai *et al.*, 2014a).

1.1. *Related Work*

Al-Riyami and Paterson (2003) presented a security model for CL-PKC. The model has two types of adversaries: Type I (outsiders) and Type II (KGC) adversaries. Type I adversary represents a malicious outsider and Type II adversary represents an honest-but-curious KGC. Al-Riyami and Paterson also proposed the first concrete CLS scheme in the random oracle model (Bellare and Rogaway, 1993) but did not present the security notions for CLS schemes. In 2005, Huang *et al.* (2005) pointed out that Al-Riyami and Paterson's CLS scheme is insecure against Type I adversary, and presented security notions for CLS schemes. Hu *et al.* (2006) enhanced the security notions of Huang *et al.* (2005) to permit adversaries more query capabilities. Since then, Hu *et al.*'s security model is generally adopted to formalize the security notions for CLS schemes. To improve the performance of signing and verifying, several CLS schemes (Gorantla and Saxena, 2005; Cao *et al.*, 2006; Zhang and Zhang, 2008; Zhang and Mao, 2007) were constructed and analyzed. For reducing communication cost, Huang *et al.* (2007) proposed a certificateless short signature scheme, but Shim (2009) proved that their scheme is insecure against key replacement attacks. Cheng *et al.* (2013) wrote a survey article on security models for CLS schemes and presented eight potential security models according to activities and behaviors of adversaries. In particular, strong unforgeability is included in some of the eight potential security models.

The security proofs of these CLS schemes mentioned above must rely on the usage of the random oracle model (Bellare and Rogaway, 1993). However, when random oracles in real implementation are adopted with some hash functions such as SHA-1, these CLS schemes could be insecure. To overcome this problem, Liu *et al.* (2007) proposed the first CLS scheme without random oracles based on the ID-based signature proposed by

Paterson and Schuldt (2006). Unfortunately, Xiong *et al.* (2008) pointed out that the Liu *et al.*'s scheme is insecure against the attacks of Type II adversary, and proposed an improved scheme. In addition, Yuan *et al.* (2009) also proposed a new CLS scheme in the standard model. Later, Xia *et al.* (2012) presented that both schemes of Xiong *et al.* and Yuan *et al.* are vulnerable to key replacement attacks. Quite recently, Yu *et al.* (2012) proposed a new CLS scheme in the standard model. However, Cheng *et al.* (2013) presented that Yu *et al.*'s scheme is still insecure against both the KGC and the key replacement attacks under the generally adopted security model of Hu *et al.* (2006).

1.2. *Contributions*

The CLS schemes in the standard model mentioned above have been shown to be insecure under Hu *et al.*'s security model. In addition, these schemes did not concern with strongly unforgeable property. A signature scheme is said to be strongly unforgeable (Boneh *et al.*, 2006) if the signature is existentially unforgeable and, given a signature on some message m, an adversary cannot generate a new signature on m. Indeed, CLS schemes with strong unforgeability are important for constructing certificateless cryptographic schemes such as chosen-ciphertext secure certificateless cryptosystems, certificateless signcryption certificateless group signatures and so forth. In the article, we propose a strongly secure CLS scheme in the standard model. Our scheme possesses not only existential unforgeability but also strong unforgeability, while retaining effiency when compared with previously proposed CLS schemes in the standard model. Our scheme turns out to be the first CLS scheme with strong unforgeability in the standard model. Under the collision resistant hash (CRH) and computational Diffie–Hellman (CDH) assumptions, we prove that our CLS scheme possesses strong unforgeability against both Type I (outsiders) and Type II (KGC) adversaries.

1.3. *Organization*

The remainder of the article is organized as follows. Preliminaries are given in Section 2. In Section 3, we present the framework and security notions for strongly secure CLS schemes. Our concrete scheme is given in Section 4. In Section 5, we analyze the security of our scheme. Comparisons are presented in Section 6. Conclusions are given in Section 7.

2. Preliminaries

In the section, we briefly review fundamental facts of bilinear pairings and two related security assumptions. A bilinear pairing is a map $\hat{e}: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, where $\mathbb{G}_1$ and $\mathbb{G}_2$ are two multiplicative cyclic groups of large prime order p. Additionally, a bilinear pairing is an admissible bilinear map if it possesses three properties, namely, bilinear, non-degeneracy and computable (Boneh and Franklin, 2001; Tsai *et al.*, 2014b). In the following, we first present a mathematical problem and its corresponding security assumption.

Definition 1. (Computational Diffie–Hellman (CDH) problem and assumption). Given a cyclic multiplicative group $\mathbb{G}_1$ of large prime order p with generator g and g^a, $g^b \in \mathbb{G}_1$ with unknown $a, b \in \mathbb{Z}_p^*$, the computational Diffie–Hellman (CDH) problem in $\mathbb{G}_1$ is to obtain g^{ab}. We say that the (ϵ, t)-CDH assumption holds in the group $\mathbb{G}_1$ if no probabilistic polynomial-time (PPT) adversary $\mathcal{A}$ running in time at most t can solve the CDH problem in $\mathbb{G}_1$ with probability at least ϵ. The advantage of $\mathcal{A}$ is denoted as $\Pr[\mathcal{A}(g, g^a, g^b) = g^{ab}]$, where the advantage is based on the random instances chosen by $\mathcal{A}$.

In our scheme, we use collision-resistant hash (CRH) functions to construct our strongly secure CLS scheme, in which the CRH functions can be easily obtained based on the CDH assumption (Boneh *et al.*, 2006). So, the usage of the CRH functions does not strengthen the security assumption of our scheme.

Definition 2. (Collision-resistant hashing (CRH) assumption.) Let $H_k : \{0, 1\}^* \rightarrow \{0, 1\}^n$ be a family of CRH functions, where k is an index and n is a fixed bit length. We say that the (ϵ, t)-CRH assumption holds if no PPT adversary $\mathcal{A}$ running in time at most t can break the collision resistance of H_k with probability at least ϵ. The advantage of $\mathcal{A}$ is denoted as $\Pr[\mathcal{A}(k) = (m_1, m_2) : m_1 \neq m_2, H_k(m_1) = H_k(m_2)]$.

3. Framework and Adversarial Model of CLS with Strong Unforgeability

3.1. *Framework*

We present the framework of CLS schemes with strong unforgeability (or called strongly secure CLS schemes), which is identical to that of the CLS schemes in Hu *et al.* (2006), Yu *et al.* (2012), Cheng *et al.* (2013). A strongly secure CLS scheme consists of two kinds of entities, namely, users and key generation center (KGC). A strongly secure CLS is specified by five algorithms, namely, the *system setup*, the *partial private key extract*, the *user key generation*, the *signing* and the *verifying* algorithms.

- *System setup*: On input a security parameter, the KGC runs this algorithm to return the master secret key and public parameters PP. PP is available for all the other algorithms.
- *Partial private key extract*: This algorithm, run by the KGC, takes as input the master secret key and a user's identity ID, and returns the user's partial private key D_{ID} to the user by way of a secure channel.
- *User key generation*: This algorithm, run by a user, takes as input the user's identity ID, and outputs the secret key SK_{ID} and the public key PK_{ID}. Note that the full private key of a user consists of a partial private key D_{ID} and a secret key SK_{ID}.
- *Signing*: This algorithm, run by a user (signer), takes as input the user's partial private key D_{ID}, secret key SK_{ID} and a message M, and returns a signature σ.
- *Verifying*: This algorithm, run by a user (verifier), takes as input a signature σ, a message M, a user identity ID with the public key PK_{ID}, the algorithm outputs either "accept" or "reject".

3.2. *Adversarial Model*

Based on the security models in Huang *et al.* (2005), Hu *et al.* (2006), Cheng *et al.* (2013), we present the security notions for strongly secure CLS schemes. We present two types of adversaries, namely, Type I and Type II adversaries. A Type I adversary acts as a dishonest user (outsider) who can replace the public key of any entity with a value of her/his choice, but has no access to the master secret key. A Type II adversary represents an honest-but-curious KGC that owns the master secret key, but cannot perform public key replacement. The security notions for strongly secure CLS schemes are modeled using the following games (Games I and II) between a challenger $\mathcal{B}$ and two types of adversaries.

Game 1 (for Type I Adversary $\mathcal{A}$, Outsider)

- *Setup*. The challenger $\mathcal{B}$ takes a security parameter ψ and runs the *system setup* algorithm to produce the master secret key and public parameters PP. PP is given to $\mathcal{A}$ and the master secret key is kept by $\mathcal{B}$.
- *Queries*. The Type I adversary $\mathcal{A}$ performs the following queries adaptively:
  - *Public key retrieve* (ID). When $\mathcal{A}$ requests the public key of an entity ID, the challenger $\mathcal{B}$ runs the *user key generation* algorithm to obtain the public key PK_{ID} and returns it to $\mathcal{A}$.
  - *Public key replace* (ID, PK'_{ID}). The adversary $\mathcal{A}$ replaces the public key of a user with identity ID by PK_{ID}. $\mathcal{B}$ records this replacement.
  - *Partial private key extract* (ID). When $\mathcal{A}$ requests the partial private key of an entity ID, $\mathcal{B}$ runs the *partial private key extract* algorithm to obtainD_{ID} and returns it to $\mathcal{A}$.
  - *Secret key extract* (ID). When $\mathcal{A}$ requests the secret key of an entity ID, $\mathcal{B}$ runs the *user key generation* algorithm to obtain the secret key SK_{ID} and returns it to $\mathcal{A}$. Here, $\mathcal{B}$ returns the symbol $\perp$ if the identity ID has already appeared in the *public key replace* query.
  - *Signing* (ID, M). When $\mathcal{A}$ requests a signature on the message M for an entity ID, $\mathcal{B}$ uses the current partial private key D_{ID} and secret key SK_{ID} to run the *signing* algorithm to obtain a signature on the message M. Note that, no matter whether the public key of the identity ID has not been replaced or not, $\mathcal{B}$ then returns to $\mathcal{A}$.
- *Forgery*. $\mathcal{A}$ generates a signature tuple (ID^*, M^*, σ^*). We say that $\mathcal{A}$ win the game if the following conditions holds:
  (1) (ID^*, M^*, σ^*) can pass the *verifying* algorithm.
  (2) (ID^*, M^*, σ^*) has never appeared during the *signing* query.
  (3) ID^* has never been submitted in the *partial private key extract* query.

Game 2 (for Type II Adversary, KGC)

- *Setup*. The challenger $\mathcal{B}$ takes a security parameter ψ and runs the *system setup* algorithm to produce the master secret key and public parameters PP. The master secret key and PP are given to the adversary $\mathcal{A}$.

- *Queries*. The adversary $\mathcal{A}$ may issue queries defined in Game 1, except for the *public key replace* query, in an adaptive manner. $\mathcal{A}$ has no need to request the *partial private key extract* query since it owns the master secret key. Note that it is unreasonable to ask $\mathcal{B}$ to respond the *signing* queries if the public key of the entity ID has been replaced.
- *Forgery*. $\mathcal{A}$ generates a signature tuple (ID^*, M^*, σ^*). We say that $\mathcal{A}$ win the game if the following conditions holds:
  1. (ID^*, M^*, σ^*) can pass the *verifying* algorithm.
  2. (ID^*, M^*, σ^*) has never been appeared during the *signing* query.
  3. ID^* has never been submitted in the *secret key extract* query.

Definition 3. A CLS scheme with strong unforgeability is said to be strongly secure against adaptive chosen-message attacks if no PPT adversary $\mathcal{A}$ has a non-negligible advantage in Games 1 and 2.

Remark 1. Note that, for existential unforgeability in CLS schemes, the condition (2) in the *Forgery phase* of both Games I and II is, instead, weakened as that (ID^*, M^*) has never been submitted during the *signing* query. Hence, strong unforgeability offers adversaries more capabilities than existential unforgeability does.

4. Strongly Secure CLS Scheme

In this section, we present a concrete CLS scheme with strong unforgeability in the standard model that consists of the following algorithms:

- *Setup*: Given a security parameter ψ, the KGC selects two cyclic groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of a prime order $p > 2^{\psi}$. Let g be a generator of $\mathbb{G}_1$ and $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be an admissible bilinear map. The KGC selects $\alpha \in \mathbb{Z}_p^*$ and $g_2 \in \mathbb{G}_1$ at random, computes $g_1 = g^{\alpha} \in \mathbb{G}_1$, and sets the master secret key as g_2^{α}. The KGC also selects five collision-resistant hash functions $H_1 : \{0,1\}^* \rightarrow \{0,1\}^m$, $H_2, H_3 : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \{0,1\}^n$, $H_4 : \{0,1\}^* \rightarrow \{0,1\}^l$ and $H_5 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$, where m, n and l are fixed lengths. Furthermore, the KGC randomly selects u_i, u', s_j, s', t_j, t', w_k, $w' \in \mathbb{G}_1$ and four vectors $\vec{u} = (u_i)$, $\vec{s} = (s_j)$, $\vec{t} = (t_j)$ and $\vec{w} = (w_k)$ of length m, n, n and l, respectively. The KGC publishes $PP = \langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, H_1, H_2, H_3, H_4, H_5, g, g_1, g_2, u', \vec{u}, s', \vec{s}, t', \vec{t}, w', \vec{w} \rangle$.
- *Partial private key extract*: Let $\vec{v} = H_1(ID) = (v_1, v_2, \ldots, v_m)$ be a bit string of length m representing a user's identity $ID \in \{0,1\}^*$. To construct the user's partial private key D_{ID}, the KGC selects a random value $r_v \in \mathbb{Z}_p^*$ and computes $D_{ID} = (D_1, D_2) = (g_2^{\alpha} U^{r_v}, g^{r_v})$, where $U = u' \prod_{i=1}^{m} u_i^{v_i}$. The KGC transmits D_{ID} to the user by way of a secure channel.
- *User key generation*: On input a user's identity ID, this algorithm randomly chooses two secret values $\theta_1, \theta_2 \in \mathbb{Z}_p^*$, and computes the user's public key $PK_{ID} = (PK_1, PK_2) = (g^{\theta_1}, g^{\theta_2})$, $\overrightarrow{vs} = H_2(PK_1, PK_2) = (vs_1, \ldots, vs_n)$ and $\overrightarrow{vt} =$

$H_3(PK_1, PK_2) = (vt_1, \ldots, vt_n)$, where $\overrightarrow{vs}$ and $\overrightarrow{vt}$ are two bit strings of length n. Finally, the user's secret key is $SK_{ID} = g_2^{\theta_1} S^{\theta_1} T^{\theta_2}$, where $S = s' \prod_{j=1}^{n} s_j^{vs_j}$ and $T = t' \prod_{j=1}^{n} t_j^{vt_j}$.

- *Signing*: Given a user's D_{ID}, SK_{ID} and M, the signer selects a random number $r_m \in \mathbb{Z}_p^*$, and computes $\overrightarrow{vm} = H_4(M) = (vm_1, vm_2, \ldots, vm_l)$ and $h = H_5(M \| g^{r_m})$. A signature σ on the message M is constructed by computing

$$\sigma = (\sigma_1, \sigma_2, \sigma_3) = \big(D_1^h (SK_{ID})^h W^{r_m}, D_2^h, g^{r_m}\big),$$

where $W = (w' \prod_{k=1}^{l} w_k^{vm_k})$.

- *Verifying*: Given a signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$, the message M, identity ID and its associated public key $PK_{ID} = (PK_1, PK_2)$, a verifier accepts the signature if the following equality holds:

$$\hat{e}(g, \sigma_1) = \hat{e}(g_1, g_2)^h \hat{e}(\sigma_2, U) \hat{e}(PK_1, g_2 S)^h \hat{e}(PK_2, T)^h \hat{e}(\sigma_3, W).$$

Correctness. We present that the equality in the verifying algorithm is correct as follows:

$$\begin{aligned}
\hat{e}(g, \sigma_1) &= \hat{e}\big(g, D_1^h (SK_{ID})^h W^{r_m}\big) \\
&= \hat{e}\big(g, g_2^{\alpha h} U^{h r_v} g_2^{\theta_1 h} S^{\theta_1 h} T^{\theta_2 h} W^{r_m}\big) \\
&= \hat{e}\big(g, g_2^{\alpha}\big)^h \hat{e}\big(g, U^{h r_v}\big) \hat{e}\big(g, g_2^{\theta_1 h} S^{\theta_1 h}\big) \hat{e}\big(g, T^{\theta_2 h}\big) \hat{e}\big(g, W^{r_m}\big) \\
&= \hat{e}\big(g^{\alpha}, g_2\big)^h \hat{e}\big(g^{h r_v}, U\big) \hat{e}\big(g^{\theta_1}, g_2 S\big)^h \hat{e}\big(g_2^{\theta}, T\big)^h \hat{e}\big(g^{r_m}, W\big) \\
&= \hat{e}(g_1, g_2)^h \hat{e}(\sigma_2, U) \hat{e}(PK_1, g_2 S)^h \hat{e}(PK_2, T)^h \hat{e}(\sigma_3, W).
\end{aligned}$$

5. Security Analysis

In this section, we establish two theorems to prove that our CLS scheme possesses strong unforgeability against adaptive chosen-message attacks under the CRH and CDH assumptions for both Type I (in Game 1) and Type II (in Game 2) adversaries defined in Section 3.

Theorem 1. *Under the CDH and CRH assumptions, our CLS scheme is strongly secure against Type I adversary. Concretely, suppose that a Type I adversary $\mathcal{A}$ with an advantage ϵ can break our CLS scheme within a running time τ. In the meantime, $\mathcal{A}$ can make at most q_E* partial private key extract *queries, q_S* signing queries *and q_K public key replace and* secret key extract *queries combined. Then there exists an algorithm $\mathcal{B}$ who has an advantage*

$$\epsilon' \geqslant \frac{\epsilon}{16 q_K q_S (q_E + q_S)(m+1)(n+1)(l+1)}$$

to violate the CDH assumption or a advantage $\epsilon'' \geqslant \frac{\epsilon}{4}$ to violate the CRH assumption within a running time $\tau' = \tau + O(m q_E + n q_K + (m + n + l) q_S)\tau_1 + O(q_E + q_S + q_K)\tau_2$, *where τ_1 and τ_2 are the computational costs of a scalar multiplication and an exponentiation in $\mathbb{G}_1$, respectively.*

Proof. Suppose that a Type I adversary $\mathcal{A}$ may forge a valid signature to our CLS scheme, then we can establish an algorithm $\mathcal{B}$ to resolve the CDH problem or find a collision pair for the CRH assumption. We assume that $\mathcal{B}$ is given an instance of the CDH problem with $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, g, g^a, g^b \rangle$. The algorithm $\mathcal{B}$ simulates the challenger in Game 1 to respond $\mathcal{A}$ as follows.

Setup. The challenger $\mathcal{B}$ chooses five CRH functions $H_1 : \{0,1\}^* \to \{0,1\}^m$, $H_2, H_3 : \mathbb{G}_1 \times \mathbb{G}_1 \to \{0,1\}^n$, $H_4 : \{0,1\}^* \to \{0,1\}^l$ and $H_5 : \{0,1\}^* \to \mathbb{Z}_p^*$, where m, n and l are fixed lengths. The adopted CRH functions do not act as random oracles in the following proof. $\mathcal{B}$ sets $l_v = 2(q_E + q_S)$, $l_s = q_K$ and $l_m = 2q_S$, and choose three random integers k_v, k_s and k_m, where $0 \leqslant k_v \leqslant m$, $0 \leqslant k_s \leqslant n$ and $0 \leqslant k_m \leqslant l$. For the given values of q_E, q_S, m, n and l, the following inequalities $l_v(m+1) < p$, $l_s(n+1) < p$ and $l_m(l+1) < p$ must hold. $\mathcal{B}$ selects the following random integers: $x', x_1, \dots, x_m \in \mathbb{Z}_{l_v}$, $y', y_1, \dots, y_m \in \mathbb{Z}_p$, $r', r_1, \dots, r_n \in \mathbb{Z}_{l_s}$, $z', z_1, \dots, z_n \in \mathbb{Z}_p$, $c', c_1, \dots, c_l \in \mathbb{Z}_{l_m}$, and d', $d_1, \dots, d_l \in \mathbb{Z}_p$. As in our scheme, we have $\vec{v} = H_1(ID) = (v_1, \dots, v_m)$ for an identity ID, $\overrightarrow{vs} = H_2(PK_1, PK_2) = (vs_1, \dots, vs_n)$ and $\overrightarrow{vt} = H_3(PK_1, PK_2) = (vt_1, \dots, vt_n)$ for a public key $PK_{ID} = (PK_1, PK_2)$, and $\overrightarrow{vm} = H_4(M) = (vm_1, \dots, vm_l)$ for a message M. We then construct six functions F, J, Q, E, K and L as follows:

$$F(\vec{v}) = -l_v k_v + x' + \sum_{i=1}^{m} v_i x_i, \qquad J(\vec{v}) = y' + \sum_{i=1}^{m} v_i y_i,$$

$$Q(\overrightarrow{vs}) = -l_s k_s + r' + \sum_{j=1}^{n} vs_j r_j, \qquad E(\overrightarrow{vt}) = z' + \sum_{j=1}^{n} vt_j z_j,$$

$$K(\overrightarrow{vm}) = -l_m k_m + c' + \sum_{k=1}^{l} vm_k c_k, \qquad L(\overrightarrow{vm}) = d' + \sum_{k=1}^{l} vm_k d_k.$$

$\mathcal{B}$ constructs public parameters PP by computing $g_1 = g^a$, $g_2 = g^b$; $u' = g_2^{-l_v k_v + x'} g^{y'}$, $u_i = g_2^{x_i} g^{y_i}$ for $1 \leqslant i \leqslant m$; $s' = g_2^{-1-l_s k_s + r'}$, $s_j = g_2^{r_j}$ for $1 \leqslant j \leqslant n$; $t' = g^{z'}$, $t_j = g^{z_j}$ for $1 \leqslant j \leqslant n$; $w' = g_2^{-l_m k_m + c'} g^{d'}$, $w_k = g_2^{c_k} g^{d_k}$ for $1 \leqslant k \leqslant l$. $\mathcal{B}$ publishes $PP = \langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, H_1, H_2, H_3, H_4, H_5, g, g_1, g_2, u', \vec{u}, s', \vec{s}, t', \vec{t}, w', \vec{w} \rangle$. For reducing complexity of the cumbersome notations mentioned above, we also conclude with four relations which will be frequently used in the sequel, namely,

$$U = u' \prod_{i=1}^{m} u_i^{v_i} = g_2^{F(\vec{v})} g^{J(\vec{v})}, \qquad S = s' \prod_{j=1}^{n} s_j^{vs_j} = g_2^{Q(\overrightarrow{vs})-1},$$

$$T = t' \prod_{j=1}^{n} t_j^{vt_j} = g^{E(\overrightarrow{vt})}, \qquad W = w' \prod_{k=1}^{l} w_k^{vm_k} = g_2^{K(\overrightarrow{vm})} g^{L(\overrightarrow{vm})}.$$

Queries. To avoid collision and consistently respond to queries, $\mathcal{B}$ maintains a list L of tuples $\langle ID, \theta_1, \theta_2, PK_{ID}, SK_{ID}\rangle$, which is initially empty. The challenger $\mathcal{B}$ responds to $\mathcal{A}$'s queries in an adaptive manner as follows:

- *Public key retrieve* (ID): When $\mathcal{A}$ makes this query on ID, the challenger $\mathcal{B}$ does as follows:
  (1) If the list L contains ID, $\mathcal{B}$ returns the corresponding PK_{ID} to $\mathcal{A}$.
  (2) Otherwise, $\mathcal{B}$ chooses two secret values $\theta_1, \theta_2 \in \mathbb{Z}_p^*$ and computes the public key $PK_{ID} = (PK_1, PK_2) = (g^{\theta_1}, g^{\theta_2})$, $\overrightarrow{vs} = H_2(PK_1, PK_2) = (vs_1, \ldots, vs_n)$, $\overrightarrow{vt} = H_3(PK_1, PK_2) = (vt_1, \ldots, vt_n)$ and the secret key $SK_{ID} = g_2^{\theta_1} S^{\theta_1} T^{\theta_2}$. $\mathcal{B}$ then adds the tuple $\langle ID, \theta_1, \theta_2, PK_{ID}, SK_{ID}\rangle$ in L and returns PK_{ID} to $\mathcal{A}$.
- *Public key replace* (ID, PK'_{ID}): When $\mathcal{A}$ makes this query on ID, $\mathcal{B}$ looks up L for the tuple $\langle ID, \theta_1, \theta_2, PK_{ID}, SK_{ID}\rangle$. If the list L contains ID, $\mathcal{B}$ replaces PK_{ID} with PK'_{ID}. Otherwise, $\mathcal{B}$ adds the tuple $\langle ID, \perp, \perp, PK'_{ID}, \perp\rangle$ in L.
- *Partial private key extract* (ID): $\mathcal{B}$ computes $\vec{v} = H_1(ID)$, $F(\vec{v})$ and $J(\vec{v})$. If $F(\vec{v}) \neq 0 \bmod p$, $\mathcal{B}$ selects a random value $r_v \in \mathbb{Z}_p$ and responds with the partial private key D_{ID} computed by

$$D_{ID} = (D_1, D_2) = \Big(\big(g^a\big)^{\frac{-J(\vec{v})}{F(\vec{v})}} U^{r_v}, \big(g^a\big)^{\frac{-1}{F(\vec{v})}} g^{r_v}\Big).$$

It is convinced that D_{ID} is a valid partial private key since

$$\begin{aligned} D_1 &= \big(g^a\big)^{\frac{-J(\vec{v})}{F(\vec{v})}} U^{r_v} = \big(g^a\big)^{\frac{-J(\vec{v})}{F(\vec{v})}} \big(g_2^{F(\vec{v})} g^{J(\vec{v})}\big)^{r_v} \\ &= g_2^a \big(g_2^{F(\vec{v})} g^{J(\vec{v})}\big)^{\frac{-a}{F(\vec{v})}} \big(g_2^{F(\vec{v})} g^{J(\vec{v})}\big)^{r_v} \\ &= g_2^a \big(g_2^{F(\vec{v})} g^{J(\vec{v})}\big)^{r_v - \frac{a}{F(\vec{v})}} = g_2^a U^{r'_v} \end{aligned}$$

and

$$D_2 = \big(g^a\big)^{\frac{-1}{F(\vec{v})}} g^{r_v} = g^{r_v - \frac{a}{F(\vec{v})}} = g^{r'_v},$$

where $r'_v = r_v - \frac{a}{F(v)}$. Otherwise, if $F(\vec{v}) = 0 \bmod p$, $\mathcal{B}$ aborts.
- *Secret key extract* (ID): When $\mathcal{A}$ makes this query on ID, $\mathcal{B}$ looks up L for the tuple $\langle ID, \theta_1, \theta_2, PK_{ID}, SK_{ID}\rangle$. If the list L contains ID, $\mathcal{B}$ returns SK_{ID} to $\mathcal{A}$. Otherwise, $\mathcal{B}$ runs the user key generation algorithm to generate θ_1, θ_2, PK_{ID} and SK_{ID}. $\mathcal{B}$ adds the tuple $\langle ID, \theta_1, \theta_2, PK_{ID}, SK_{ID}\rangle$ in L and returns SK_{ID} to $\mathcal{A}$.
- *Signing* (ID, M): When $\mathcal{A}$ makes this query on (ID, M), $\mathcal{B}$ first computes $\vec{v} = H_1(ID)$, $\overrightarrow{vs} = H_2(PK_1, PK_2) = (vs_1, vs_2, \ldots, vs_n)$, $\overrightarrow{vt} = H_3(PK_1, PK_2) = (vt_1, vt_2, \ldots, vt_n)$, $\overrightarrow{vm} = H_4(ID) = (vm_1, vm_2, \ldots, vm_l)$ and $h = H_5(M \parallel g^{r_m})$. $\mathcal{B}$ then computes $F(\vec{v})$, $J(\vec{v})$, $Q(\overrightarrow{vs})$, $E(\overrightarrow{vt})$, $K(\overrightarrow{vm})$ and $L(\overrightarrow{vm})$. If $K(\overrightarrow{vm}) = 0 \bmod p$, $\mathcal{B}$ reports failure and terminates. Otherwise, if $K(\overrightarrow{vm}) \neq 0$, $\mathcal{B}$ considers two cases as follows.

Case 1: Assume that ID has been replaced with $PK_{ID} = (PK_1, PK_2)$. If $F(\vec{v}) \neq 0 \bmod l_v$, $\mathcal{B}$ can compute the partial private key $D_{ID} = (D_1, D_2)$ as in the *partial private key extract* query, and $\mathcal{B}$ then creates a signature

$$\begin{aligned}
\sigma &= (\sigma_1, \sigma_2, \sigma_3) \\
&= \Bigl(D_1^h (PK_1)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})} Q(\overrightarrow{vs})} (PK_2)^{hE(\overrightarrow{vt})} W^{r_m}, D_2^h, (PK_1)^{\frac{-hQ(\overrightarrow{vs})}{K(\overrightarrow{vm})}} g^{r_m}\Bigr).
\end{aligned}$$

Let $r'_m = r_m - \frac{\theta_1 h Q(\overrightarrow{vs})}{K(\overrightarrow{vm})}$. Then σ is a valid signature since

$$\begin{aligned}
\sigma_1 &= D_1^h \bigl(g^{\theta_1}\bigr)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})} Q(\overrightarrow{vs})} \bigl(g^{\theta_2}\bigr)^{hE(\overrightarrow{vt})} \bigl(g_2^{K(\overrightarrow{vm})} g^{L(\overrightarrow{vm})}\bigr)^{r_m} \\
&= D_1^h g_2^{\theta_1 h} g_2^{\theta_1 h(Q(\overrightarrow{vs})-1)} \bigl(g^{E(\overrightarrow{vt})}\bigr)^{\theta_2 h} \bigl(g_2^{K(\overrightarrow{vm})} g^{L(\overrightarrow{vm})}\bigr)^{r_m - \frac{\theta_1 h Q(\overrightarrow{vs})}{K(\overrightarrow{vm})}} \\
&= D_1^h g_2^{\theta_1 h} S^{\theta_1 h} T^{\theta_2 h} W^{r'_m}, \\
\sigma_2 &= D_2^h, \\
\sigma_3 &= \bigl(g^{\theta_1}\bigr)^{\frac{-hQ(\overrightarrow{vs})}{K(\overrightarrow{vm})}} g^{r_m} = g^{r'_m}.
\end{aligned}$$

On the other hand, if $F(\vec{v}) = 0 \bmod p$, $\mathcal{B}$ selects two values $r_v, r_m \in \mathbb{Z}_p^*$ at random and responds with the signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$, where

$$\begin{aligned}
\sigma_1 &= \bigl(g^a\bigr)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})}} U^{hr_v} (PK_1)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})} Q(\overrightarrow{vs})} (PK_2)^{hE(\overrightarrow{vt})} W^{r_m}, \\
\sigma_2 &= g^{hr_v}, \\
\sigma_3 &= g^{r'_m}.
\end{aligned}$$

Let $r'_m = r_m - \frac{ah + \theta_1 h Q(\overrightarrow{vs})}{K(\overrightarrow{vm})}$, it is obvious that σ is a valid signature since

$$\begin{aligned}
\sigma_1 &= \bigl(g^a\bigr)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})}} U^{hr_v} \bigl(g^{\theta_1}\bigr)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})} Q(\overrightarrow{vs})} \bigl(g^{\theta_2}\bigr)^{hE(\overrightarrow{vt})} \bigl(g_2^{K(\overrightarrow{vm})} g^{L(\overrightarrow{vm})}\bigr)^{r_m} \\
&= g_2^{ah} U^{hr_v} g_2^{\theta_1 h} S^{\theta_1 h} T^{\theta_2 h} W^{r_m - \frac{ah + \theta_1 h Q(\overrightarrow{vs})}{K(\overrightarrow{vm})}} \\
&= g_2^{ah} U^{hr_v} g_2^{\theta_1 h} S^{\theta_1 h} T^{\theta_2 h} W^{r'_m}, \\
\sigma_3 &= \bigl(g^a\bigr)^{\frac{-h}{K(\overrightarrow{vm})}} \bigl(g^{\theta_1}\bigr)^{\frac{-hQ(\overrightarrow{vs})}{K(\overrightarrow{vm})}} g^{r_m} = g^{r'_m}.
\end{aligned}$$

Case 2: Let us consider the case that ID has not appeared in the *public key replace* query. If $F(\vec{v}) \neq 0$, $\mathcal{B}$ can compute the partial private key $D_{ID} = (D_1, D_2)$ as in the *partial key extract query*, and accesses the list L to obtain the secret key SK_{ID}. $\mathcal{B}$ randomly selects a value $r_m \in \mathbb{Z}_p^*$ and returns the signature

$$\sigma = (\sigma_1, \sigma_2, \sigma_3) = \big(D_1^h (SK_{ID})^h W^{r_m}, D_2^h, g^{r_m}\big).$$

If $F(\vec{v}) = 0 \bmod p$, then $\mathcal{B}$ randomly selects two values $r_v, r_m \in \mathbb{Z}_p^*$ and returns the signature

$$\sigma = (\sigma_1, \sigma_2, \sigma_3) = \Big(\big(g^a\big)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})}} U^{hr_v} (SK_{ID})^h W^{r_m}, g^{hr_v}, (g_1)^{\frac{-h}{K(\overrightarrow{vm})}} g^{r_m}\Big).$$

Forgery. Suppose that $\mathcal{A}$ forges a valid signature $\sigma^* = (\sigma_1, \sigma_2, \sigma_3)$ for ID^* on M^*, where ID^* and M^* are the target identity and message, respectively. We discuss two cases.

Case 1: If (ID^*, M^*) does not appear in the *signing* query, $\mathcal{B}$ accesses the list L to get $PK_{ID^*} = (PK_1, PK_2)$ and computes $\vec{v}^* = H_1(ID^*)$, $\overrightarrow{vs}^* = H_2(PK_1^*, PK_2^*)$, $\overrightarrow{vt}^* = H_3(PK_1^*, PK_2^*)$, $\overrightarrow{vm}^* = H_4(M^*)$, $h = H_5(M \parallel \sigma_3)$, $F(\vec{v}^*)$, $J(\vec{v}^*)$, $Q(\overrightarrow{vs}^*)$, $E(\overrightarrow{vt}^*)$, $L(\overrightarrow{vm}^*)$ and $K(\overrightarrow{vm}^*)$. If $F(\vec{v}^*) \neq 0 \bmod p$, $Q(\overrightarrow{vs}^*) \neq 0 \bmod p$ or $K(\overrightarrow{vm}^*) \neq 0$, mod p, then $\mathcal{B}$ aborts. Otherwise, if $F(\vec{v}^*) = Q(\overrightarrow{vs}^*) = K(\overrightarrow{vm}^*) = 0 \bmod p$, $\mathcal{B}$ computes g^{ab} as follows.

$$\begin{aligned}
&\frac{\sigma_1^{h^{-1}}}{(\sigma_2^{J(\vec{v}^*)})^{h^{-1}} (PK_2^{E(\overrightarrow{vt}^*)})(\sigma_3^{L(\overrightarrow{vm}^*)})^{h^{-1}}}\\
&= \frac{g_2^a U^{r_v} g_2^{\theta_1} S^{\theta_1} T^{\theta_2} W^{r_m h^{-1}}}{g^{r_v \cdot J(\vec{v}^*)} g^{\theta_2 \cdot E(\overrightarrow{vt}^*)} g^{r_m \cdot L(\overrightarrow{vm}^*) h^{-1}}}\\
&= \frac{g_2^a (g_2^{F(\vec{v}^*)} g^{J(\vec{v}^*)})^{r_v} g_2^{\theta_1} (g_2^{Q(\overrightarrow{vs}^*)-1})^{\theta_1} (g^{E(\overrightarrow{vt}^*)})^{\theta_2} (g_2^{K(\overrightarrow{vm}^*)} g^{L(\overrightarrow{vm}^*)})^{r_m h^{-1}}}{g^{r_v J(\vec{v}^*)} g^{\theta_2 E(\overrightarrow{vt}^*)} g^{r_m L(\overrightarrow{vm}^*) h^{-1}}}\\
&= \frac{g_2^a (g_2^0 g^{J(\vec{v}^*)})^{r_v} g_2^{\theta_1} (g_2^{0-1})^{\theta_1} (g^{E(\overrightarrow{vt}^*)})^{\theta_2} (g_2^0 g^{L(\overrightarrow{vm}^*)})^{r_m h^{-1}}}{g^{r_v J(\vec{v}^*)} g^{\theta_2 E(\overrightarrow{vt}^*)} g^{r_m L(\overrightarrow{vm}^*) h^{-1}}}\\
&= g_2^a = g^{ab}.
\end{aligned}$$

This solves the CDH problem.

Case 2: If (ID^*, M^*) has appeared in the *signing* query, $\mathcal{A}$ owned a previously queried signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$ of ID^* on M^*. If $\sigma_2 \neq \sigma_2^*$, the challenge $\mathcal{B}$ can output g^{ab} as in Case 1. Otherwise, if $\sigma_2 = \sigma_2^*$, then $g^{hr_v} = g^{h^* r_v}$ and so $h^* = h$. Namely, $H_5(M \parallel g^{r_m^*}) = H_5(M \parallel g^{r_m})$, where $\sigma_3^* = g^{r_m^*}$ and $\sigma_3 = g^{r_m}$. This causes a collision of H_5 which violates the CRH assumption.

In the following, we analyze the probabilities of the events that the challenger $\mathcal{B}$ does not abort. In the *partial partial key extract* query, if $F(\vec{v}) \neq 0 \bmod p$, $\mathcal{B}$ may respond to queries without aborting. In the *signing* query, if $K(\overrightarrow{vm}) \neq 0 \bmod p$, $\mathcal{B}$ may respond to queries without aborting. In the*forgery phase*, if $F(\vec{v}^*) = Q(\overrightarrow{vs}^*) = K(\overrightarrow{vm}^*) = 0 \bmod p$, $\mathcal{B}$ completes the simulation without aborting. We denote that q_I represents the number of the identities queried in *partial private key extract* and *signing* queries not involving *ID**. Meanwhile, q_M represents the number of the messages queried in the *signing* involving *ID**. It is obvious that we have $q_I < q_E + q_S$ and $q_M < q_S$. Here, we define several events as follows: $X_i : F(\vec{v}) \neq 0 \bmod l_v$; $X^* : F(\vec{v}^*) = 0 \bmod p$; $Y_k : K(\overrightarrow{vm}) \neq 0 \bmod l_m$; $Y^* : K(vm^*) = 0 \bmod p$; $Z^* : Q(\overrightarrow{vs}^*) = 0 \bmod p$, where $1 < i \leqslant q_I$ and $1 < k \leqslant q_M$, Hence, the probabilities of $\mathcal{B}$ not aborting in Case 1 and Case 2, respectively, are

$$\begin{aligned}
\Pr[\neg\text{abortCase1}] &\geqslant \Pr\left[\bigwedge_{i=1}^{q_I} X_i \wedge X^* \wedge \bigwedge_{k=1}^{q_M} Y_k \wedge Y^* \wedge Z^*\right] \\
&= \Pr\left[X^*\right] \cdot \Pr\left[\bigwedge_{i=1}^{q_I} X_i | X^*\right] \cdot \Pr\left[Y^*\right] \cdot \Pr\left[\bigwedge_{k=1}^{q_M} Y_k | Y^*\right] \cdot \Pr\left[Z^*\right]
\end{aligned}$$

and

$$\Pr[\neg\text{abortCase2}] \geqslant \Pr\left[\bigwedge_{i=1}^{q_I} X_i \wedge \bigwedge_{k=1}^{q_M} Y_k\right] = \Pr\left[\bigwedge_{i=1}^{q_I} X_i\right] \cdot \Pr\left[\wedge \bigwedge_{k=1}^{q_M} Y_k\right].$$

Since $l_v(m+1) < p$, $l_s(n+1) < p$ and $l_m(l+1) < p$, we have that $F(\vec{v}) = 0 \bmod p$ implies $F(\vec{v}) = 0 \bmod l_v$, $Q(\overrightarrow{vs}) = 0 \bmod p$ implies $Q(\overrightarrow{vs}) = 0 \bmod l_s$ and $K(\overrightarrow{vm}) = 0 \bmod p$ implies $K(\overrightarrow{vm}) = 0 \bmod l_m$. Furthermore, $F(\vec{v}) = 0 \bmod l_v$, $Q(\overrightarrow{vs}) = 0 \bmod l_s$ and $K(\overrightarrow{vm}) = 0 \bmod l_m$, there will be a unique choice of k_v with $0 \leqslant k_v \leqslant m$, k_s with $0 \leqslant k_s \leqslant n$ and k_m with $0 \leqslant k_m \leqslant l$ such that $F(\vec{v}) = 0 \bmod p$, $Q(\overrightarrow{vs}) = 0 \bmod p$ and $K(\overrightarrow{vm}) = 0 \bmod p$. Since k_v, x', $x_1, \ldots, x_m$, k_s, r', $r_1, \ldots, r_n, k_m$ and c', $c_1, \ldots, c_l$ are randomly chosen, we have the probabilities of the events X^*, Y^* and Z^* as follows.

$$\begin{aligned}
\Pr[X^*] &= \Pr\left[F\left(\vec{v}^*\right) = 0 \bmod p\right] \\
&= \Pr\left[F\left(\vec{v}^*\right) = 0 \bmod p \wedge F\left(\vec{v}^*\right) = 0 \bmod l_v\right] \\
&= \Pr\left[F\left(\vec{v}^*\right) = 0 \bmod l_v\right] \cdot \Pr\left[F\left(\vec{v}^*\right) = 0 \bmod p | F\left(\vec{v}^*\right) = 0 \bmod l_v\right] \\
&= \frac{1}{l_v} \frac{1}{m+1}, \\
\Pr\left[Y^*\right] &= \Pr\left[K\left(\overrightarrow{vm}^*\right) = 0 \bmod p\right] \\
&= \Pr\left[K\left(\overrightarrow{vm}^*\right) = 0 \bmod p \wedge K\left(\overrightarrow{vm}^*\right) = 0 \bmod l_m\right] \\
&= \Pr\left[K\left(\overrightarrow{vm}^*\right) = 0 \bmod l_m\right] \cdot \Pr\left[K\left(\overrightarrow{vm}^*\right) = 0 \bmod p | K\left(\overrightarrow{vm}^*\right) = 0 \bmod l_m\right] \\
&= \frac{1}{l_m} \frac{1}{l+1}, \\
\Pr[Z^*] &= \Pr\left[Q\left(\overrightarrow{vs}^*\right) = 0 \bmod p\right] \geqslant \frac{1}{l_s} \frac{1}{n+1}.
\end{aligned}$$

We then have that

$$\begin{aligned}\Pr\left[\bigwedge_{i=1}^{q_I} X_i|X^*\right] &= 1-\Pr\left[\bigvee_{i=1}^{q_I}\neg X_i|X^*\right]\geqslant 1-\sum_{i=1}^{q_I}\Pr\left[\neg X_i|X^*\right]\\ &= 1-\frac{q_I}{l_v}\geqslant 1-\frac{q_E+q_S}{l_v}\end{aligned}$$

and

$$\begin{aligned}\Pr\left[\bigwedge_{k=1}^{q_M} Y_k|Y^*\right] &= 1-\Pr\left[\bigvee_{k=1}^{q_M}\neg Y_k|Y^*\right]\geqslant 1-\sum_{k=1}^{q_M}\Pr\left[\neg Y_k|Y^*\right]\\ &= 1-\frac{q_M}{l_m}\geqslant 1-\frac{q_S}{l_m}.\end{aligned}$$

We also have

$$\Pr\left[\bigwedge_{i=1}^{q_I} X_i\right]=\Pr\left[\bigwedge_{i=1}^{q_I} X_i|X^*\right]\quad\text{and}\quad\Pr\left[\bigwedge_{k=1}^{q_M} Y_k\right]=\Pr\left[\bigwedge_{k=1}^{q_M} Y_k|Y^*\right],$$

by independency, hence we can obtain that

$$\Pr\left[\bigwedge_{i=1}^{q_I} X_i\wedge X^*\right] = \Pr\left[X^*\right]\cdot\Pr\left[\bigwedge_{i=1}^{q_I} X_i|X^*\right]\geqslant\left(\frac{1}{l_v}\frac{1}{m+1}\right)\left(1-\frac{q_E+q_S}{l_v}\right)$$

and

$$\Pr\left[\bigwedge_{k=1}^{q_M} Y_k\wedge Y^*\right] = \Pr\left[Y^*\right]\cdot\Pr\left[\bigwedge_{k=1}^{q_M} Y_k|Y^*\right]\geqslant\left(\frac{1}{l_m}\frac{1}{l+1}\right)\left(1-\frac{q_S}{l_m}\right).$$

As mentioned earlier, we have set $l_v=2(q_E+q_S)$, $l_s=q_K$ and $l_m=q_S$. Hence, the probabilities of $\mathcal{B}$ not aborting in Case 1 and Case 2, respectively,

$$\begin{aligned}\Pr[\neg\text{abortCase1}] &\geqslant \Pr\left[\bigwedge_{i=1}^{q_I} X_i\wedge X^*\wedge\bigwedge_{k=1}^{q_M} Y_k\wedge Y^*\wedge Z^*\right]\\ &= \Pr\left[X^*\right]\cdot\Pr\left[\bigwedge_{i=1}^{q_I} X_i|X^*\right]\cdot\Pr\left[Y^*\right]\cdot\Pr\left[\bigwedge_{k=1}^{q_M} Y_k|Y^*\right]\cdot\Pr\left[Z^*\right]\\ &\geqslant \frac{1}{4(q_E+q_S)(m+1)4q_S(l+1)q_K(n+1)}\\ &= \frac{1}{16q_Kq_S(q_E+q_S)(m+1)(n+1)(l+1)}\end{aligned}$$

and

$$\Pr[\neg\text{abortCase2}] \geqslant \Pr\left[\bigwedge_{i=1}^{q_I} X_i \wedge \bigwedge_{k=1}^{q_M} Y_k\right] = \Pr\left[\bigwedge_{i=1}^{q_I} X_I\right] \cdot \Pr\left[\bigwedge_{k=1}^{q_M} Y_K\right] \geqslant \frac{1}{4}.$$

Hence, if $\mathcal{A}$ with an advantage ϵ can break the proposed CLS scheme, $\mathcal{B}$ has an advantage

$$\epsilon' \geqslant \frac{\epsilon}{16q_K q_S(q_E + q_S)(m+1)(n+1)(l+1)}$$

to violate the CDH assumption or a advantage $\epsilon'' \geqslant \frac{\epsilon}{4}$ to violate the CRH assumption.

According to the descriptions above, $\mathcal{B}$ requires $O(m)$ scalar multiplications and $O(1)$ exponentiations in the *private partial key extract* queries. In both *public key replace* and *secret key extract* queries, $O(n)$ scalar multiplications and $O(1)$ exponentiations are required. In the *signing* queries, $\mathcal{B}$ requires $O(m+n+l)$ scalar multiplications and $O(1)$ exponentiations. So, the total running time required for $\mathcal{B}$ is $\tau' = \tau + O(mq_E + nq_K + (m+n+l)q_S)\tau_1 + O(q_E + q_K + q_S)\tau_2$, where τ, τ_1 and τ_2 are $\mathcal{A}$'s running time, the computational costs of a scalar multiplication and an exponentiation, respectively. □

Theorem 2. *Under the CDH and CRH assumptions, our CLS scheme is strongly secure against Type II adversary. Concretely, suppose that a Type II adversary $\mathcal{A}$ with an advantage ϵ can break our CLS scheme within a running time τ. In the meantime, $\mathcal{A}$ can make at most q_K secret key extract queries and q_S signing queries. Then there exists an algorithm $\mathcal{B}$ who has an advantage*

$$\epsilon' \geqslant \frac{\epsilon}{4q_S q_K(l+1)}$$

to violate the CDH assumption or a advantage $\epsilon'' \geqslant \frac{\epsilon}{2}$ to violate the CRH assumption within a running time $\tau' = \tau + O(nq_K + (m+n+l)q_S)\tau_1 + O(q_K + q_S)\tau_2$, where τ_1 and τ_2 are the computational costs of a scalar multiplication and an exponentiation in $\mathbb{G}_1$, respectively.

Proof. Suppose that a Type II adversary $\mathcal{A}$ may forge a valid signature to our CLS scheme, then we can establish an algorithm $\mathcal{B}$ to resolve the CDH problem or find a collision pair for the CRH assumption. We assume that $\mathcal{B}$ is given an instance of the CDH problem with $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, g, g^a, g^b\rangle$. The algorithm $\mathcal{B}$ simulates the challenger in Game 2 to respond $\mathcal{A}$ as follows.

Setup. As the *Setup* phase in Theorem 1, the challenger $\mathcal{B}$ first chooses five CRH functions. $\mathcal{B}$ then sets $l_m = 2q_S$, and selects a random integer k_m, where $0 \leqslant k_m \leqslant l$. We suppose that $l_m(l+1) < p$ for the given values of q_S and l. $\mathcal{B}$ selects the following random integers: $y', y_1, \dots, y_m \in \mathbb{Z}_p$, $r', r_1, \dots, r_n \in \mathbb{Z}_p$, $z', z_1, \dots, z_n \in \mathbb{Z}_p$, $c', c_1, \dots, c_l \in \mathbb{Z}_{l_m}$ and $d', d_1, \dots, d_l \in \mathbb{Z}_p$, and the following vectors $\vec{u} = (u_i)$, $\vec{s} = (s_j)$, $\vec{t} = (t_j)$ and

$\vec{w} = (w_k)$ of the length m, n, n and l, respectively. Then, as the *Setup* phase in Theorem 1, $\mathcal{B}$ constructs $\vec{v} = H_1(ID) = (v_1, \dots, v_m)$, $\overrightarrow{vs} = H_2(PK_1, PK_2) = (vs_1, \dots, vs_n)$ and $\overrightarrow{vt} = H_3(PK_1, PK_2) = (vt_1, \dots, vt_n)$, and $\overrightarrow{vm} = H_4(M) = (vm_1, \dots, vm_l)$. We constructs five functions J, Q, E, K and L as follows:

$$J(\vec{v}) = y' + \sum_{i=1}^{m} v_i y_i, \qquad Q(\overrightarrow{vs}) = r' + \sum_{j=1}^{n} vs_j r_j, \qquad E(\overrightarrow{vt}) = z' + \sum_{j=1}^{n} vt_j z_j,$$

$$K(\overrightarrow{vm}) = -l_m k_m + c' + \sum_{k=1}^{l} vm_k c_k, \qquad L(\overrightarrow{vm}) = d' + \sum_{k=1}^{l} vm_k d_k.$$

The challenger $\mathcal{B}$ chooses a random value $\alpha \in \mathbb{Z}_p$, and sets $g_1 = g^{\alpha}$ and $g_2 = g^{b}$. Furthermore, $\mathcal{B}$ computes and sends the master secret key g_2^{α} to the adversary $\mathcal{A}$. $\mathcal{B}$ constructs public parameters PP by computing $g_1 = g^{\alpha}$, $g_2 = g^{b}$; $u' = g^{y'}$, $u_i = g^{y_i}$ for $1 \leqslant i \leqslant m$; $s' = g_2^{r'}$, $s_j = g_2^{r_j}$ for $1 \leqslant j \leqslant n$; $t' = g^{z'}$, $t_j = g^{z_j}$ for $1 \leqslant j \leqslant n$; $w' = g_2^{-l_m k_m + c'} g^{d'}$, $w_k = g_2^{c_k} g^{d_k}$ for $1 \leqslant k \leqslant l$. $\mathcal{B}$ publishes $PP = \langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, H_1, H_2, H_3, H_4, H_5, g, g_1, g_2, u', \vec{u}, s', \vec{s}, t', \vec{t}, w', \vec{w} \rangle$. For the cumbersome notations defined above, as the *Setup* phase in Theorem 1, we also have four notations U, S, T and W.

Queries. To avoid collision and consistently respond to queries, the challenger $\mathcal{B}$ maintains an initially empty list L of tuples $\langle ID, \theta_1, \theta_2, PK_{ID}, SK_{ID} \rangle$. Moreover, $\mathcal{B}$ chooses a target identity ID' and a random value $\theta_2 \in \mathbb{Z}_p^*$. $\mathcal{B}$ computes the public key $PK'_{ID} = (g^{a}, g^{\theta_2})$. $\mathcal{B}$ adds $\langle ID', \perp, \theta_2, PK'_{ID}, \perp \rangle$ in L. $\mathcal{B}$ responds to $\mathcal{A}$'s queries in an adaptive manner as follows:

- *Public key retrieve* (*ID*): As the *public key retrieve* query in Theorem 1, $\mathcal{B}$ responds to $\mathcal{A}$'s queries.
- *Secret key extract* (*ID*): Upon receiving a query on *ID*, $\mathcal{B}$ responds to the query as follows:
  (1) If $ID = ID'$, $\mathcal{B}$ reports failure and terminates.
  (2) If $ID \neq ID'$, $\mathcal{B}$ accesses the tuple $\langle ID, \theta_1, \theta_2, PK_{ID}, SK_{ID} \rangle$ in the list L. If *ID* appears in L, $\mathcal{B}$ returns SK_{ID} to $\mathcal{A}$. If *ID* does not appear in L, $\mathcal{B}$ runs the *user key generation* algorithm to generate θ_1, θ_2, PK_{ID}, and SK_{ID}. $\mathcal{B}$ adds $\langle ID, \theta_1, \theta_2, PK_{ID}, SK_{ID} \rangle$ in L and returns SK_{ID} to $\mathcal{A}$.
- *Signing* (ID, M): When $\mathcal{A}$ makes this query on (ID, M), the challenger $\mathcal{B}$ first computes $\vec{v} = H_1(ID)$, $\overrightarrow{vs} = H_2(PK_1, PK_2) = (vs_1, vs_2, \dots, vs_n)$, $\overrightarrow{vt} = H_3(PK_1, PK_2) = (vt_1, vt_2, \dots, vt_n)$, $\overrightarrow{vm} = H_4(ID) = (vm_1, vm_2, \dots, vm_l)$ and $h = H_5(M \parallel g^{r_m})$. $\mathcal{B}$ then computes $J(\vec{v})$, $Q(\overrightarrow{vs})$, $E(\overrightarrow{vt})$, $K(\overrightarrow{vm})$ and $L(\overrightarrow{vm})$. If $K(\overrightarrow{vm}) \neq 0 \bmod p$, $\mathcal{B}$ considers the following two cases.

Case 1: If $ID = ID'$, $\mathcal{B}$ performs the *partial private key extract* algorithm to obtain the partial private key $D_{ID} = (D_1, D_2)$. $\mathcal{B}$ randomly selects a value $r_m \in \mathbb{Z}_p^*$ and computes the signature

$$\begin{aligned}\sigma &= (\sigma_1, \sigma_2, \sigma_3)\\ &= \Big(D_1^h (PK_1)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})}(1+Q(\overrightarrow{vs}))} (PK_2)^{hE(\overrightarrow{vt})} W^{r_m},\ D_2^h,\ (PK_1)^{\frac{-h(1+Q(\overrightarrow{vs}))}{K(\overrightarrow{vm})}} g^{r_m}\Big).\end{aligned}$$

Let $r'_m = r_m - \frac{ah(1+Q(\overrightarrow{vs}))}{K(\overrightarrow{vm})}$. Then σ is a valid signature since

$$\begin{aligned}\sigma_1 &= D_1^h \big(g^a\big)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})}(1+Q(\overrightarrow{vs}))} \big(g^{\theta_2}\big)^{hE(\overrightarrow{vt})} \big(g_2^{K(\overrightarrow{vm})} g^{L(\overrightarrow{vm})}\big)^{r_m}\\ &= D_1^h g_2^{ah} \big(g_2^{Q(\overrightarrow{vs})}\big)^{ah} \big(g^{E(\overrightarrow{vt})}\big)^{\theta_2 h} \big(g_2^{K(\overrightarrow{vm})} g^{L(\overrightarrow{vm})}\big)^{r_m - \frac{ah(1+Q(\overrightarrow{vs}))}{K(\overrightarrow{vm})}}\\ &= D_1^h g_2^{ah} S^{ah} T^{\theta_2 h} W^{r'_m},\\ \sigma_2 &= D_2^h,\\ \sigma_3 &= \big(g^a\big)^{\frac{-hL(\overrightarrow{vm})}{K(\overrightarrow{vm})}(1+Q(\overrightarrow{vs}))} g^{r_m} = g^{r'_m}.\end{aligned}$$

Case 2: If $ID \neq ID'$, $\mathcal{B}$ performs the *partial private key extract* algorithm to get the partial private key $D_{ID} = (D_1, D_2)$, and accesses the list L to obtain the secret key SK_{ID}. $\mathcal{B}$ randomly selects a value $r_m \in \mathbb{Z}_p^*$ and computes the signature

$$\sigma = (\sigma_1, \sigma_2, \sigma_3) = \big(D_1^h (SK_{ID})^h W^{r_m}, D_2^h, g^{r_m}\big).$$

Forgery. Suppose that the adversary $\mathcal{A}$ generates a valid signature $\sigma^* = (\sigma_1, \sigma_2, \sigma_3)$ for ID^* on M^*, where ID^* and M^* are the target identity and message, respectively. We discuss two cases.

Case 1: If (ID^*, M^*) does not appear in the *signing query*. If $ID^* \neq ID'$, $\mathcal{B}$ reports failure and terminates. If $ID^* = ID'$, $\mathcal{B}$ accesses the list L to obtain $PK_{ID^*} = (PK_1, PK_2)$, and computes $\vec{v}^* = H_1(ID^*)$, $\overrightarrow{vs}^* = H_2(PK_1^*, PK_2^*)$, $\overrightarrow{vt}^* = H_3(PK_1^*, PK_2^*)$, $\overrightarrow{vm}^* = H_4(M^*)$, $h = H_5(M \parallel \sigma_3)$, $J(\vec{v}^*)$, $Q(\overrightarrow{vs}^*)$, $E(\overrightarrow{vt}^*)$, $L(\overrightarrow{vm}^*)$ and $K(\overrightarrow{vm}^*)$. If $K(\overrightarrow{vm}^*) \neq 0$, $\mathcal{B}$ aborts. If $K(\overrightarrow{vm}^*) = 0 \bmod p$, $\mathcal{B}$ computes $(g^{ab})^{1+Q(\overrightarrow{vs}^*)}$ as follows.

$$\begin{aligned}V &= \frac{\sigma_1^{h^{-1}}}{g_2^\alpha (\sigma_2^{J(\vec{v}^*)})^{h^{-1}} (PK_2^{E(\overrightarrow{vt}^*)})(\sigma_3^{L(\overrightarrow{vm}^*)})^{h^{-1}}}\\ &= \frac{g_2^\alpha U^{r_v} g_2^a S^a T^{\theta_2} W^{r_m h^{-1}}}{g_2^\alpha g^{r_v \cdot J(\vec{v}^*)} g^{\theta_2 \cdot E(\overrightarrow{vt}^*)} g^{r_m \cdot L(\overrightarrow{vm}^*) h^{-1}}}\\ &= \frac{g_2^\alpha (g^{J(\vec{v}^*)})^{r_v} g_2^a (g_2^{Q(\overrightarrow{vs}^*)})^a (g^{E(\overrightarrow{vt}^*)})^{\theta_2} (g_2^{K(\overrightarrow{vm}^*)} g^{L(\overrightarrow{vm}^*)})^{r_m h^{-1}}}{g_2^\alpha g^{r_v J(\vec{v}^*)} g^{\theta_2 E(\overrightarrow{vt}^*)} g^{r_m L(\overrightarrow{vm}^*) h^{-1}}}\end{aligned}$$

$$= g_2^a \big(g_2^{Q(\overrightarrow{vs}^*)}\big)^a = \big(g_2^a\big)^{1+Q(\overrightarrow{vs}^*)}$$
$$= \big(g^{ab}\big)^{1+Q(\overrightarrow{vs}^*)}.$$

By computing $V^{(1+Q(\overrightarrow{vs}^*))^{-1}}$, we obtain the value g^{ab}. This solves the CDH problem.

Case 2: If (ID^*, M^*) has appeared in the *signing* query, $\mathcal{A}$ owned a previously queried signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$ of ID^* on M^*. If $\sigma_2 \neq \sigma_2^*$, the challenge $\mathcal{B}$ is able to output g^{ab} as in Case 1. Otherwise, if $\sigma_2 = \sigma_2^*$, then $g^{hr_v} = g^{h^* r_v}$ and so $h^* = h$. Namely, $H_5(M \parallel g^{r_m^*}) = H_5(M \parallel g^{r_m})$, where $\sigma_3^* = g^{r_m^*}$ and $\sigma_3 = g^{r_m}$. This causes a collision of H_5 which violates the CRH assumption.

In the following, we analyze the probabilities of the events that the challenger $\mathcal{B}$ does not abort. In the *signing* query, if $K(\overrightarrow{vm}) \neq 0 \bmod p$, $\mathcal{B}$ may respond to queries without aborting. In the *forgery* phase, if $ID^* = ID'$ and $K(\overrightarrow{vm}^*) = 0 \bmod p$, $\mathcal{B}$ completes the simulation without aborting. We denote that q_M represents the number of the messages in *signing* queries not involving ID^*. It is obvious that we have $q_M < q_S$. Here, we define several events as follows: $Y_k : K(\overrightarrow{vm}) \neq 0 \bmod l_m$; $Y^* : K(vm^*) = 0 \bmod p$; $Z^* : ID^* = ID'$, where $1 < k < q_M$. Hence, the probabilities of the challenger $\mathcal{B}$ not aborting in Case 1 and Case 2, respectively, are

$$\begin{aligned}
\Pr[\neg\text{abortCase1}] &\geqslant \Pr\left[\wedge \bigwedge_{k=1}^{q_M} Y_k \wedge Y^* \wedge Z^*\right] \\
&= \Pr\big[Y^*\big] \cdot \Pr\left[\bigwedge_{k=1}^{q_M} Y_k | Y^*\right] \cdot \Pr[Z^*]
\end{aligned}$$

and

$$\Pr[\neg\text{abortCase2}] \geqslant \Pr\left[\bigwedge_{k=1}^{q_M} Y_k\right].$$

By the assumption in the setup, we have that $0 \leqslant l_m k_m \leqslant l_m l < p$ and $0 \leqslant c' + \sum_{k=1}^{l} vm_k c_k < l_m(l+1) < p$, where $0 \leqslant k_m \leqslant l$ and $c', c_1, \ldots, c_l \in \mathbb{Z}_{l_m}$. This implies that $-p \leqslant l_m k_m + c' + \sum_{k=1}^{l} vm_k c_k < p$, i.e. $-p < K(\overrightarrow{vm}) < p$. Thus, $k(\overrightarrow{vm}) = 0 \bmod p$ implies $k(\overrightarrow{vm}) = 0 \bmod l_m$. With similar to the probability analysis in Theorem 1, we have

$$\Pr\left[\bigwedge_{k=1}^{q_M} Y_k\right] = \Pr\big[Y^*\big] \cdot \Pr\left[\bigwedge_{k=1}^{q_M} Y_k | Y^*\right] \geqslant \left(\frac{1}{l_m}\frac{1}{l+1}\right)\left(1 - \frac{q_s}{l_m}\right).$$

By $l_m = 2q_S$, the probabilities of $\mathcal{B}$ not aborting in Case 1 and Case 2 are, respectively,

$$\begin{aligned}\Pr[\neg\text{abortCase1}] &\geqslant \Pr\Bigg[\wedge \bigwedge_{k=1}^{q_M} Y_k \wedge Y^* \wedge Z^*\Bigg]\\ &= \Pr\big[Y^*\big]\cdot\Pr\Bigg[\bigwedge_{k=1}^{q_M} Y_k|Y^*\Bigg]\cdot\Pr\big[Z^*\big]\\ &\geqslant \frac{1}{4q_Sq_K(l+1)}\end{aligned}$$

and

$$\Pr[\neg\text{abortCase2}] \geqslant \Pr\Bigg[\wedge \bigwedge_{k=1}^{q_M} Y_k\Bigg] \geqslant \frac{1}{2}.$$

Hence, if $\mathcal{A}$ with an advantage ϵ can break the proposed CLS scheme, $\mathcal{B}$ has an advantage

$$\epsilon' \geqslant \frac{\epsilon}{4q_Sq_K(l+1)}$$

to violate the CDH assumption or a probability $\epsilon'' \geqslant \frac{\epsilon}{2}$ to violate the CRH assumption.

According to the descriptions above, $\mathcal{B}$ requires $O(n)$ scalar multiplications and $O(1)$ exponentiations in *secret key extract* queries. In the *signing* queries, $\mathcal{B}$ requires $O(m+n+l)$ scalar multiplications and $O(1)$ exponentiations. Therefore, the total time required for $\mathcal{B}$ is $\tau' = \tau + O((nq_K + (m+n+l)q_S)\tau_1 + O(q_K + q_S)\tau_2$, where τ, τ_1 and τ_2 are $\mathcal{A}$'s running time, the computational costs of a scalar multiplication and an exponentiation, respectively. □

6. Comparisons

To analyze and compare the computational cost, we consider two time-consuming operations T_p and T_e, which, respectively, denote the time of executing a bilinear pairing operation $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and the time of executing an exponentiation operation in $\mathbb{G}_1$ or $\mathbb{G}_2$. Table 1 lists the comparisons among the schemes of Liu *et al.* (2007), Yuan *et al.* (2009), Yu *et al.* (2012) and ours in terms of computational cost, security assumption and security property. All the signing phases of the schemes above require no pairing operation to sign a message. For the verifying phase, Yu *et al.* (2012) require five pairing operations but their scheme has been shown insecure against the Type I adversary.

For security analysis, the schemes of Liu *et al.* (2007), Yuan *et al.* (2009) and Yu *et al.* (2012) have been shown insecure against the Type I adversary or Type II adversary. For the security assumption, our scheme is based on the CDH and CRH assumptions, but the others are based on non-pairing-based generalized bilinear Diffie–Hellman (NGBDH) (Liu

Table 1
Comparisons between previously proposed CLS schemes and ours.

	Liu *et al.*'s scheme (2007)	Yuan *et al.*'s scheme (2009)	Yu *et al.*'s scheme (2012)	Our scheme
Computational cost for signing	$5T_e$	$9T_e$	$6T_e$	$5T_e$
Computational cost for verifying	$6T_p$	$6T_p$	$5T_p + T_e$	$6T_p + 3T_e$
Against Type I adversary	Yes	No	No	Yes
Against Type II adversary	No	Yes	No	Yes
Security assumption	NGBDH Many-DH	AC-DH 2-Many-DH	NGBDH Many-DH	CDH CRH
Security property	Existential unforgeability	Existential unforgeability	Existential unforgeability	Strong unforgeability

et al., 2007), many Diffie–Hellman (Many-DH) assumptions (Lysyanskaya, 2002), augmented computational Diffie–Hellman (AC-DH) and 2-many Diffie–Hellman (2-Many-DH) assumptions (Yuan *et al.*, 2009). For Type I and Type II adversaries, we emphasize that our scheme possesses strong unforgeability and is the first secure CLS scheme in the standard model.

7. Conclusions

Strongly secure CLS schemes are important for constructing certificateless cryptographic schemes such as chosen-ciphertext secure certificateless cryptosystems and certificateless group signatures. In this article, we proposed the first strongly secure CLS scheme in the standard model. Comparisons with previously proposed schemes were made to demonstrate the advantages of our scheme in terms of security property while retaining efficiency. For security analysis, under the CDH and CRH assumptions, we demonstrate that the proposed CLS scheme possesses strong unforgeability against adaptive chosen-message attacks under a generally adopted security model.

Acknowledgments. This research was partially supported by the Ministry of Science and Technology, Taiwan, R.O.C., under contract No. MOST103-2221-E-022-MY2.

References

Al-Riyami, S.S., Paterson, K.G. (2003). Certificateless public key cryptography. In: *Proceedings of Asiacrypt'03*, *LNCS*, Vol. 2894, pp. 452–473.

Bellare, M., Rogaway, P. (1993). Random oracles are practical: a paradigm for designing efficient protocols. In: *Proceedings of CCS'93*. ACM, pp. 62–73.

Boneh, D., Franklin, M. (2001). Identity-based encryption from the Weil pairing. In: *Proceedings of Crypto'01*, *LNCS*, Vol. 2139, pp. 213–229.

Boneh, D., Shen, E., Waters, B. (2006). Strongly unforgeable signatures based on computational Diffie–Hellman. In: *Proceedings of PKC'06*, *LNCS*, Vol. 3958, pp. 229–240.

Cao, X., Paterson, K.G., Kou, W. (2006). An attack on a certificateless signature scheme. *Cryptology ePrint Archive, Report 2006/367*. http://eprint.iacr.org/2006/367.

Chen, Y.C., Tso, R., Susilo, W., Huang, X., Horng, G. (2013). Certificateless signatures: structural extensions of security models and new provably secure schemes. *Cryptology ePrint Archive, Report 2013/193*. Available at http://eprint.iacr.org/2013/193.pdf.

Cheng, L., Wen, Q., Jin, Z.P., Zhang, H. (2013). On the security of a certificateless signature scheme in the standard model. *Cryptology ePrint Archive, Report, 2013/153*. http://eprint.iacr.org/2013/153.

Dent, A.W. (2008). A survey of certificateless encryption schemes and security models. *International Journal of Information Security*, 7(5), 349–377.

Gorantla, M.C., Saxena, A. (2005). An efficient certificateless signature scheme. In: *Proceedings of CIS'05*, *LNCS*, Vol. 3802, pp. 110–116.

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.

Hu, B., Wong, D., Zhang, Z., Deng, X. (2006). Key replacement attack against a generic construction of certificateless signature. In: *Proceedings of ACISP'06*, *LNCS*, Vol. 4058, pp. 235–346.

Huang, X., Susilo, W., Mu, Y., Zhang, F. (2005). On the security of a certificateless signature scheme from Asiacrypt 2003. In: *Proceedings of CANS'05*, *LNCS*, Vol. 3810, pp. 13–25.

Huang, X., Mu, Y., Susilo, W., Wong, D., Wu, W. (2007). Certificateless signature revisited. In: *Proceedings of ACISP'06*, *LNCS*, Vol. 4586, pp. 308–322.

Libert, B., Quisquater, J.J. (2006). On constructing certificateless cryptosystems from identity based encryption. In: *Proceedings of PKC'06*, *LNCS*, Vol. 3958, pp. 474–490.

Liu, J.K., Au, M.H., Susilo, W. (2007). Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model. In: *Proceedings of ASIACCS'07*. ACM, New York, pp. 273–283.

Lysyanskaya, A. (2002). Unique signatures and verifiable random functions from the DH-DDH separation. In: *Proceedings of Crypto'02*, *LNCS*, Vol. 2442, pp. 597–612.

Paterson, K.G., Schuldt, J.C.N. (2006). Efficient identity-based signatures secure in the standard model. In: *Proceedings of ACISP'06*, *LNCS*, Vol. 4058, pp. 207–222.

Shamir, A. (1984). Identity-based cryptosystems and signature schemes. In: *Proceedings of Crypto'84*, *LNCS*, Vol. 196, pp. 47–53.

Shim, K. (2009). Breaking the short certificateless signature scheme. *Information Sciences*, 179(3), 303–306.

Tsai, T.T., Tseng, Y.M., Wu, T.Y. (2012). A full secure revocable ID-based encryption in the standard model. *Informatica*, 23(3), 487–505.

Tsai, J.L., Lo, N.W., Wu, T.C. (2014a). Weaknesses and improvements of an efficient certificateless signature scheme without using bilinear pairings. *International Journal of Communication Systems*, 27(7), 1083–1090.

Tsai, T.T., Tseng, Y.M., Huang, S.S. (2014b). Efficient strongly unforgeable ID-based signature without random oracles. *Informatica*, 25(3), 505–521.

Tsai, T.T., Tseng, Y.M., Wu, T.Y. (2014c). RHIBE: Constructing revocable hierarchical ID-based encryption from HIBE. *Informatica*, 25(2), 299–326.

Tseng, Y.M., Tsai, T.T. (2012). Efficient revocable ID-based encryption with a public channel. *The Computer Journal*, 55(4), 475–486.

Tseng, Y.M., Huang, Y.H., Chang, H.J. (2014). Privacy-preserving multireceiver ID-based encryption with provable security. *International Journal of Communication Systems*, 27(7), 1034–1050.

Tso, R., Huang, X., Susilo, W. (2012). Strongly secure certificateless short signatures. *The Journal of Systems and Software*, 85(6), 1409–1417.

Waters, B. (2005). Efficient identity-based encryption without random oracles. In: *Proceedings of Eurocrypt'05*, *LNCS*, Vol. 3494, pp. 114–127.

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

Xia, Q., Xu, C.X., Yu, Y. (2012). Key replacement attack on two certificateless signature schemes without random oracles. *Key Engineering Materials*, 439–440, 1601–1611.

Yang, G., Tan, C.H. (2011). Certificateless public key encryption: a new generic construction and two pairing-free schemes. *Theoretical Computer Science*, 412(8–10), 662–674.

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 oracle. In: *Proceedings of ISA'09*, *LNCS*, Vol. 5576, pp. 31–40.

Yum, D., Lee, P. (2004). Generic construction of certificateless signature. In: *Proceedings of ACISP'04*, *LNCS*, Vol. 3108, pp. 200–211.

Zhang, J., Mao, J. (2007). Security analysis of two signature schemes and their improved schemes. In: *Proceedings of ICCSA'07*, *LNCS*, Vol. 4705, pp. 589–602.

Zhang, L., Zhang, F. (2008). A new provably secure certificateless signature scheme. In: *IEEE ICC'08*, pp. 1685–1689.

Y.-H. Hung received the BS degree from the Department of Mathematics, National Changhua University of Education, Taiwan, in 1999. He received the MS degree from the Department of Applied Mathematics, National Hsinchu University of Education, Taiwan, in 2008. He is currently a PhD candidate in the Department of Mathematics, National Changhua University of Education, Taiwan. His research interests include applied cryptography and pairing-based cryptography.

S.-S. Huang is currently a Professor in the Department of Mathematics, National Changhua University of Education, Taiwan. His research interests include number theory, cryptography, and network security. He received his PhD from the University of Illinois at Urbana-Champaign in 1997 under the supervision of Professor Bruce C. Berndt.

Y.-M. Tseng is currently a Professor in the Department of Mathematics, National Changhua University of Education, Taiwan. He is a member of IEEE Computer Society, IEEE Communications Society and the Chinese Cryptology and Information Security Association (CCISA). In 2006, his paper received the Wilkes Award from The British Computer Society. He has published over one hundred scientific journal and conference papers on various research areas of cryptography, security and computer network. His research interests include cryptography, network security, computer network and mobile communications. He serves as an editor of several international journals.

T.-T. Tsai is currently a senior engineer in software system develop department of innovation Digital System Business Group (iDSBG) of HON HAL Precision IND. CO., LTD, Taiwan. His research interests include applied cryptography and pairing-based cryptography. He received his PhD from the University of the Department of Mathematics, National Changhua University of Education, Taiwan, in 2014.

Standartinio modelio visiškai nesuklastojamas parašas be sertifikato

Ying-Hao HUNG, Sen-Shan HUANG, Yuh-Min TSENG, Tung-Tso TSAI

Viešojo rakto šifravimo sistemos be sertifikatų (CL-PKS) buvo pasiūlytos dviejų svarbių problemų sprendimui: pirma, pašalinio asmens, kuris žino slaptąjį raktą, problema ir, antra, sertifikavimo eliminavimas įprastose viešojo rakto sistemose. Pasiūlyta kelios schemos be sertifikatų (CLS), kurios yra dalinai nepažeidžiamos prieš adaptyviąsias pasirinktų pranešimų atakas ir tik kai kurios iš jų yra visiškai nepažeidžiamos. Šio straipsnio autoriai siūlo saugią standartinio šifravimo modelio CLS schemą, kuri yra nepažeidžiama prieš adaptyviąsias pasirinktų pranešimų atakas. Kai maišos funkcija yra kolizijoms atspari (CRF) ir galioja Diffie–Hellmano prielaidos (CDF), tuomet pasiūlyta schema yra visiškai nesuklastojama esant I tipo (pašalinio asmens) ir II tipo (raktų generavimo centro) grėsmėms.