Provably secure and efficient certificateless signature in the standard model

Lin Cheng*, Qiaoyan Wen, Liming Zhou

State Key Laboratory of Networking and Switch Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract

Certificateless public key cryptography eliminates inherent key escrow problem in identity-based cryptography, and does not yet requires certificates as in the traditional public key infrastructure. However, most of certificateless signature schemes without random oracles have been demonstrated to be insecure. In this paper, we propose a new certificateless signature scheme and prove that our new scheme is existentially unforgeable against adaptively chosen message attack in the standard model. Performance analysis shows that our new scheme has shorter system parameters, shorter length of signature, and higher computational efficiency than the previous schemes in the standard model.

Keywords: Cryptography; Signature; Certificateless signature; Bilinear parings

1. Introduction

Digital signature plays a crucial role to provide integrity, authentication and non-repudiation of data. In traditional public key signature algorithms the public key of the user (signer or verifier) is essentially a random bit ring, it needs a certificate issued by a certification authority (CA) to achieve authentication of the user's public key. To conquer the problem of costly certificate management, Shamir [8] proposed the notion of identity - based cryptography, in which, the user's public key is derived directly from its name, email-address or other identity information, the user's private key is generated by a trusted third party called Key Generation Center (KGC). Such cryptosystem eliminates the need for public key certificate. But, it suffers from the key escrow problem, i.e., the KGC knows the user's private key. A malicious KGC can decrypt any ciphertext and forge the signature of any user. To overcome the drawback of key escrow in IBC, Al-Riyami and Paterson [1] introduced certificateless public cryptography (CL-PKC) in 2003. In CL-PKC, the user's private key is a combination partial private key computed by KGC and some user-chosen secret value, the user's public key is computed from the KGC's public parameters and the secret value of the user. Hence, CL-PKC avoids usage of certificates and resolves the key escrow problem.

Since Al-Riyami and Paterson's certificateless signature scheme [1], many CLS schemes such as [5–7, 10, 11, 13–15] have been proposed. However, most of these certificateless signature schemes are provably secure in the random oracle model [3], which can only be considered as a heuristic argument [4]. It has been shown in [2] that the security of schemes may not preserve when the random oracle is instantiated with a particular hash function such as SHA-1. The fist certificateless signature scheme in the standard model is proposed by Liu et al.[7] in 2007. Unfortunately, in 2008, Xiong et al. [10] showed that Liu et ai.'s scheme [7] is insecure against a "malicious-but-passive" KGC attack and proposed an improved scheme. In 2009, Yuan [13] presented another provably secure CLS scheme against "malicious-but-passive" KGC attack in the standard model. However, Xia et al. [9] showed that both Xiong et al.'s improved scheme [10] and Yuan et al.'s scheme [13] are vulnerable to key replacement attack. Recently, Yu et al. [12] proposed a certificateless signature scheme which is an improved version of the existing certificateless signature schemes [7, 10, 13]. In this paper, we propose a new certificateless signature scheme and prove that our new scheme is existentially unforgeable against adaptively chosen message attack in the standard model. Compared with the existing schemes [10, 12, 13] in the standard model, our scheme offers shorter system parameters, shorter length of signature, and higher computational efficiency.

*Corresponding author.

Email address: stonewoods302@163.com, Tel:18710065389 (Lin Cheng)

The rest of paper is organized as follows. In Section 2, some preliminaries are introduced. In Section 3, we construct a new certificateless signature scheme in the standard model. In Section 4, we give the security proof and the performance analysis of our signature scheme. Finally, a concluding remark is given in Section 5.

2. Preliminaries

2.1. Security model

Generally, two types of attackers should be considered in a certificateless cryptosystem. The Type I attacker $\mathcal{A}_I$ models an "outsider" adversary, who can compromise user's secret value or replace user's public key, but neither compromise master secret key nor get access to partial private key. We call this attack launched by the type I adversary as the key replacement attack. The Type II attacker $\mathcal{A}_{II}$ models a malicious KGC who knows the master secret key, and can derive partial private key, but cannot compromise user's secret value nor replace any public key.

A certificateless signature scheme is existentially unforgeable against chosen message attack if no adversaries $\mathcal{A}_I$ and $\mathcal{A}_{II}$ have a non-negligible advantage in the following game played between a challenger and an adversary.
Game1. We illustrate the first game played between a challenger and a Type I adversary $\mathcal{A}_I$ as below.
Initialization. Challenger runs algorithm **Setup** to generate the master key msk and the master public key mpk. Challenger then gives mpk to $\mathcal{A}_I$ and keeps msk secret.
Phase 1. In this phase, $\mathcal{A}_I$ adaptively performs a polynomially bounded number of oracle queries as below:
Public-Key-Broadcast-Oracle. When $\mathcal{A}_I$ requests the public key for any identity ID, challenger computes the corresponding public key pk_{ID} and returns pk_{ID} to $\mathcal{A}_I$.
Partial-Private-Key-Oracle. When $\mathcal{A}_I$ requests ID's partial private key, challenger computes the corresponding partial private key psk_{ID} for this identity and returns psk_{ID} to $\mathcal{A}_I$.
Public-Key-Replacement-Oracle. When $\mathcal{A}_I$ supplies an identity ID and a new valid public key value pk'_{ID}, challenger replaces the current public key with pk'_{ID}.
Private-Key-Extract-Oracle. When $\mathcal{A}_I$ requests the private key of an identity ID whose public key was not replaced, challenger computes the private key sk_{ID} for this identity and returns sk_{ID} to $\mathcal{A}_I$.
Sign-Oracle. When $\mathcal{A}_I$ supplies an identity ID, and a message m, challenger C responds with a valid signature δ. It is possible for the challenger not to be aware of the signer's secret value when the associated public key has been replaced. In this case, we require $\mathcal{A}_I$ to provide the signer's secret value.
Output. Eventually, $\mathcal{A}_I$ outputs (ID^*, m^*, δ^*), where ID^* is the identity of a target user, m^* is a message, and δ^* is a signature for m^*. $\mathcal{A}_I$ wins the game if
(1) ID^* has not been submitted to **Partial-Private-Key-Oracle**.
(2) (ID^*, m^*) has not been submitted to the **Sign-Oracle**.
(3) $1 \leftarrow$ **Verify**(params, $ID^*, PK_{ID^*}, m^*, \delta^*$).

Game II. We illustrate the second game played between a challenger and a Type II adversary $\mathcal{A}_{II}$ as below.
Initialization. Challenger runs algorithm **Setup** to generate the master secret key msk and the master public key mpk. Challenger then gives mpk and msk to adversary $\mathcal{A}_{II}$.
Phase 1. In this phase, $\mathcal{A}_{II}$ adaptively issues a polynomially bounded number of queries as in game I. The difference in this phase is that $\mathcal{A}_{II}$ cannot replace any public key and $\mathcal{A}_{II}$ can compute the partial private key of any identity by itself.
Output. Eventually, $\mathcal{A}_{II}$ outputs (ID^*, m^*, δ^*), where ID^* is the identity of a target user, m^* is a message, and δ^* is a signature for m^*. $\mathcal{A}_{II}$ wins the game if
(1) ID^* has not been submitted to **Private-Key-Extract-Oracle**.
(2) (ID^*, m^*) has not been submitted to the **Sign-Oracle**.
(3) $1 \leftarrow$ **Verify**(params, $ID^*, PK_{ID^*}, m^*, \delta^*$).

2.2. Complexity assumptions

2.2.1. Non-pairing-based generalised bilinear DH (NGBDH) assumption

Given a group G of prime order p with a generator g and elements $g^a, g^b \in G$ where a, b are selected randomly from Z_p^*, the NGBDH problem in G is to output the pair (g^{abc}, g^c). An algorithm $\mathcal{B}$ has at least ε advantage in solving the NGBDH problem if $Pr[\mathcal{B}(g, g^a, g^b) = (g^{abc}, g^c)] \geq \varepsilon$.

The (ε, t)-NGBDH assumption is said to hold if no algorithm running in time at most t can solve the NGBDH problem in G with an advantage at least ε.

2.2.2. Many-DH assumption

Given a group G of prime order p with generator g and elements $(g^a, g^b, g^c, g^{ab}, g^{ac}, g^{bc}) \in G$ where a, b, c are selected randomly from Z_p^*, the Many-DH problem is to output g^{abc}. An algorithm $\mathcal{B}$ has at least an ε advantage in solving the Many-DH problem if $Pr[\mathcal{B}(g, g^a, g^b, g^c, g^{ab}, g^{ac}, g^{bc}) = g^{abc}] \geq \varepsilon$.

The (ε, t)-Many-DH assumption is said to hold if no algorithm running in time at most t can solve the Many-DH problem in G with an advantage at least ε.

3. New certificateless signature scheme

Our certificateless signature scheme consists of the following five algorithms:

Setup: Let (G, G_T) be bilinear groups where $|G| = |G_T| = p$ for a large prime p. g is a generator of G. Randomly select $\alpha \in Z_p$, $g_2 \in G$ and compute $g_1 = g^\alpha$. $e : G \times G \to G_T$ denotes an admissible pairing. Select $u', v', v \in G$ and vector $\mathbf{u} = (u_i)$ of length n , where all the entries are random elements of G. $H_0 : \{0,1\}^* \to \{0,1\}^n$ and $H : \{0,1\}^* \times G^2 \to Z_p$ are two collision-resistant hash functions. The public parameters are $\{G, G_T, e, g, g_1, g_2, u', v', v, \mathbf{u}, H_0, H\}$ and the master secret key is g_2^α.

PartialPrivateKeyGen. Let ID be a bit string of length n and $ID[i]$ be the i-th bit. Define $\mathcal{U} \subset \{1, \ldots, n\}$ to be the set of indices i such that $ID[i] = 1$. The KGC picks a random value $r \in Z_p^*$ and computes partial private key

$$psk_{ID} = (psk_1, psk_2) = \left(g_1^\beta \cdot F_u(ID)^r, g^r\right)$$

where $F_u(ID) = u' \prod_{i \in \mathcal{U}} u_i$.

UserkeyGen. Pick a secret value $x_{ID} \in Z_p^*$ and compute public key $pk_{ID} = (X_{ID}, \delta_{ID})$ where $X_{ID} = h^{x_{ID}}$ and δ_{ID} is the Schnorr one-time signature using x_{ID} as the signing key and $(h, X_{ID} = h^{x_{ID}})$ as the verification. The message can be any arbitrary string which can be included in mpk. Then user picks r' randomly from Z_p^* and computes sk_{ID} as

$$\begin{aligned}(sk_1, sk_2) &= \left(psk_1^{x_{ID}} \cdot F_u(ID)^{r'}, psk_2^{x_{ID}} \cdot g^{r'}\right) \\ &= \left(g_1^{\beta x_{ID}} \cdot F_u(ID)^{r x_{ID} + r'}, g^{r x_{ID} + r'}\right).\end{aligned}$$

Sign To sign a message $m \in \{0,1\}^*$, a signer with identity ID, picks a random $k \in Z_p$ and computes $h = H(m, ID, sk_2, g^k)$. The signer computes $\delta = (\delta_1, \delta_2, \delta_3)$ as follows.

$$\delta_1 = sk_1 \cdot (v' \cdot v^h)^k, \delta_2 = sk_2, \delta_3 = g^k.$$

Verify: Given a signature $\delta = (\delta_1, \delta_2, \delta_3)$ for an identity ID, public key $pk_{ID} = (X_{ID}, \delta_{ID})$ on a message m , a verifier checks the validity of the signature as follows.

1. Check whether the public key X_{ID} is the correctly formed.If not, output $\perp$ and abort the algorithm. Otherwise, compute $h = H(m, ID, \delta_2, \delta_3)$.
2. Verify $e(\delta_1, g) = X_{ID} \cdot e(\delta_2, F_u(ID)) \cdot e(\delta_3, v' \cdot v^h)$ holds with equality. Accept the signature and output true if the above verify equation holds, otherwise, output false and reject the signature.

4. Analysis of the proposed scheme

In this section, we give the security proof and the performance analysis of our certificateless signature scheme.

4.1. Security proof

We now prove that the above signature scheme is existentially unforgeable against adaptively chosen message attack in the standard model.

Theorem 1. Our scheme is $(\epsilon, t, q_k, q_r, q_{pp}, q_p, q_s)$ secure during Game 1, assuming the (ϵ', t')-NGBDH intractability assumption holds, where $\epsilon' = \frac{\epsilon}{2(q_{pp}+q_p)(n+1)}$ and $t' = t + O((q_{pp} + q_p)n + q_s)t_m + (q_{pp} + q_p + q_k + q_s)t_e)$, where q_k, q_{pp}, q_p and q_s respectively denote the number of queries made to the **Public-Key-Broadcast-Oracle**, the **Partial-Private-Key- Oracle**, the **Private-Key-Extract-Oracle** and the **Sign-Oracle**, t_m and t_e respectively denote the time for a multiplication and an exponentiation in G.

Proof. Assume there exists a type I adversary $\mathcal{A}_I$ against our scheme. We construct a PPT simulator $\mathcal{B}$ that makes use of $\mathcal{A}_I$ to solve the NGBDH problem with probability at least ϵ' and in time at most t'. $\mathcal{B}$ is given a a group G of prime order p with a generator g and elements $g^a, g^b \in G$ where a, b are selected randomly from Z_p^*, and $\mathcal{B}$ replies the queries of $\mathcal{A}_I$ as follows.

Setup. $\mathcal{B}$ sets an integer $l = 2(q_{pp} + q_p)$ and uniformly picks an integer k, such that $0 < k < n$. $\mathcal{B}$ then chooses a value x', and a random n-vector $\mathbf{x} = (x_1, \ldots, x_n)$ where $x', x_i \in Z_l$. $\mathcal{B}$ also picks a random value y', and a random n-vector $\mathbf{y} = (y_1, \ldots, y_n)$, where $y', y_i \in Z_p$ and $t, a, b, c \in Z_p^*$. Assume p be sufficiently bigger than $(n + 1)l$ for any p, n and l and ID be a bit string of length n and $ID[i]$ be the i-th bit. Define $\mathcal{U} \subset \{1, \ldots, n\}$ to be the set of indices i such that $ID[i] = 1$. $\mathcal{B}$ chooses a collision-resistant hash function $H : \{0,1\}^* \times G^2 \to Z_p$. For ease of analysis, we define two functions $F(ID), J(ID)$. $F(ID) = -lk + x' + \sum_i^n x_i t_i$, $J(ID) = y' + \sum_i^n y_i t_i$. Then $\mathcal{B}$ assigns the public parameters as follows

$g_1 = g^\alpha, g_2 = g^\beta, u' = g_2^{x'-lk} g^{y'}, u_i = g_2^{x_i} g^{y_i}, v' = g_2^t \cdot g^b, v = g^c$. At this moment, $u' \prod_{i \in \mathcal{U}} u_i = g_1^{F(ID)} g^{J(ID)}$. $\mathcal{B}$ sends all the public parameters to $\mathcal{A}_I$.

Phase 1. $\mathcal{A}_I$ can carry out the following queries.

Public-Key-Broadcast-Oracle. Upon receiving a query for a public key of an identity ID, if (ID, pk_{ID}) exists in PublicKeyList, $\mathcal{B}$ returns pk_{ID} as the answer. Otherwise, $\mathcal{B}$ picks a secret value $x_{ID} \in Z_p^*$ and computes public key $pk_{ID} = (X_{ID}, \delta_{ID})$ where $X_{ID} = h^{x_{ID}}$ and δ_{ID} is the Schnorr one-time signature using x_{ID} as the signing key. $\mathcal{B}$ adds (ID, x_{ID}) to SecretValueList and adds (ID, pk_{ID}) to PublicKeyList, then returns the public key pk_{ID} as the answer.

Partial-Private-Key-Extract-Oracle. Upon receiving a query for a partial private key of an identity ID, $\mathcal{B}$ first search PartialPrivateKeyList for a tuple (ID, psk_{ID}). If it exists, return (ID, psk_{ID}) as the answer. Otherwise, $\mathcal{B}$ can construct a partial private key by assuming $F(ID) \neq 0 \bmod p$. $\mathcal{B}$ randomly chooses $r \in Z_p$ and computes a partial private key

$$\begin{aligned} psk_{ID} &= (psk_{ID,1}, psk_{ID,2}) \\ &= (g_2^{-J(ID)/F(ID)} (u' \prod_{i \in \mathcal{U}} u_i)^r, g_2^{-1/F(ID)} g^r) \end{aligned}$$

psk_{ID} is a valid partial private key for the identity ID shown as follows.

$$\begin{aligned} psk_{ID,1} &= g_2^{-J(ID)/F(ID)} (u' \prod_{i \in \mathcal{U}} u_i)^r = g_1^b (g_1^{F(ID)} g^{J(ID)})^{-b/F(ID)} (g_1^{F(ID)} g^{J(ID)})^r \\ &= g_1^b (g_1^{F(ID)} g^{J(ID)})^{r - b/F(ID)} = g_1^b (u' \prod_{i \in \mathcal{U}} u_i)^{r'}, \\ psk_{ID,2} &= g_2^{-1/F(ID)} g^r = g^{r - b/F(ID)} = g^{r'} \end{aligned}$$

where $r' = r - b/F(ID)$. From $-p < F(ID) < p$, we conclude that $F(ID) = 0 \bmod p$ implies $F(ID) = 0 \bmod l$, so $F(ID) \neq 0 \bmod l$ suffices to have $F(ID) \neq 0 \bmod p$. $\mathcal{B}$ adds (ID, psk_{ID}) to its PartialPrivateKeyList and returns the partial private key psk_{ID} as the query output. If, on the other hand, $F(ID) = 0 \bmod p$, $\mathcal{B}$ abort.

Private-Key-Extract-Oracle. Upon receiving a query for a private key of an identity ID, if the PrivateKeyList contains (ID, sk_{ID}), $\mathcal{B}$ returns sk_{ID}. Otherwise, $\mathcal{B}$ can construct a private key by assuming $F(ID) \neq 0 \bmod p$. $\mathcal{B}$ searches SecretValueList to find out x_{ID}. If it does not exist, $\mathcal{B}$ runs the algorithm **UserKeyGen** to generate secret-public key pair (x_{ID}, pk_{ID}), and adds (ID, x_{ID}) to SecretValueList and adds (ID, pk_{ID}) to PublicKeyList, then $\mathcal{B}$ chooses $r \in Z_p$

randomly and computes

$$\begin{aligned} sk_{ID,1} &= (g_2^{x_{ID}})^{-J(ID)/F(ID)}(u'\prod_{i\in\mathcal{U}} u_i)^r \\ &= g_1^{bx_{ID}}(g_1^{F(ID)}g^{J(ID)})^{-ax_{ID}/F(ID)}(g_2^{F(ID)}g^{J(ID)})^r \\ &= g_1^{bx_{ID}}(g_1^{F(ID)}g^{J(ID)})^{r-bx_{ID}/F(ID)} \\ &= g_1^{bx_{ID}}(u'\prod_{i\in\mathcal{U}} u_i)^t, \\ sk_{ID,2} &= (g_2^{x_{ID}})^{-1/F(ID)}g^r = g^{r-bx_{ID}/F(ID)} = g^t \end{aligned}$$

where $t = r - bx_{ID}/F(ID)$. $\mathcal{B}$ adds (ID, sk_{ID}) to PrivateKeyList and returns the private key sk_{ID}. If, on the other hand, $F(ID) = 0 \bmod p$, $\mathcal{B}$ aborts.

Public-Key-Replacement-Oracle. When the adversary requests to replace the current public key pk_{ID} of an identity ID with a new and valid public key pk'_{ID} chosen by him, $\mathcal{B}$ finds out pk_{ID} in its PublicKeyList, and replace it with the new public key pk'_{ID}. If pk_{ID} does not exist, $\mathcal{B}$ directly sets $pk_{ID} = pk'_{ID}$, while the adversary delivers x'_{ID} to $\mathcal{B}$. Then $\mathcal{B}$ adds (ID, x_{ID}) to SecretValueList and adds (ID, pk_{ID}) to PublicKeyList.

Sign-Oracle: Receiving a signature query for an identity ID and a message m,

1. If $F(ID) \neq 0$, $\mathcal{B}$ first runs **Private-Key-Extract-Oracle** to generate a private key sk_{ID}, then performs the **Sign** algorithm to create a signature on m for the identity ID.

2. If $F(ID) = 0$, $\mathcal{B}$ first retrieves the secret value x s.t.$pk_{ID} = e(g_1, g_2)^x$ (or the adversary provides the secret value x if the public key of ID has been replaced), then $\mathcal{B}$ picks three random values $r_0, r_1, k \in Z_p$, and computes $h = H(m, ID, g^{r_0 x} \cdot g^{r_1}, g^k \cdot g_1^{(-x/t)})$ and generates a signature $\delta = (\delta_1, \delta_2, \delta_3)$ on m as follows

$\delta_1 = (g_1^{-(b+ch/t)})^x \cdot (u' \prod_{i\in\mathcal{U}} u_i)^{r_1} \cdot g^{r_0 \cdot J(ID)\cdot x} \cdot (v' \cdot v^h)^k$, $\delta_2 = g^{r_0 x} \cdot g^{r_1}$, $\delta_3 = g^k \cdot g_1^{-x/t}$.

It is a valid signature,this is because

$$\begin{aligned} \delta_1 &= (g_1^{-(b+ch/t)})^x \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_1} \cdot g^{r_0\cdot J(ID)\cdot x} \cdot (v' \cdot v^h)^k \\ &= (g_2^{\alpha} \cdot g_2^{-\alpha} \cdot g_1^{-(b+ch/t)})^x \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_1} \cdot g^{r_0\cdot J(ID)\cdot x} \cdot (v' \cdot v^h)^k \\ &= (g_2^{\alpha} \cdot (g_2^t \cdot g^{b+ch})^{-(\alpha/t)})^x \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_1} \cdot (g^{J(ID)})^{r_0\cdot x} \cdot (v' \cdot v^h)^k \\ &= (g_2^{\alpha} \cdot (g_2^t \cdot g^b \cdot (g^c)^h)^{-(\alpha/t)})^x \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_1} \cdot (g_2^{F(ID)}g^{J(ID)})^{r_0\cdot x} \cdot (v' \cdot v^h)^k \\ &= (g_2^{\alpha} \cdot (v' \cdot v^h)^{-(\alpha/t)})^x \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_1} \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_0\cdot x} \cdot (v' \cdot v^h)^k \\ &= (g_2^{\alpha} \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_0})^x \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_1} \cdot (v' \cdot v^h)^{k-(\alpha\cdot x/t)} \end{aligned}$$

furthermore,

$$\begin{aligned} e(\delta_1, g) &= e((g_2^{\alpha} \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_0})^x \cdot (u'\prod_{i\in\mathcal{U}} u_i)^{r_1} \cdot (v' \cdot v^h)^{k-(\alpha\cdot x/t)}, g) \\ &= e(g_2^{\alpha}, g)^x e(u'\prod_{i\in\mathcal{U}} u_i, g^{r_0 x + r_1}) e((v' \cdot v^h)^{k-(\alpha\cdot x/t)}, g) \\ &= X_{ID} \cdot e(\delta_2, F_u(ID)) \cdot e(\delta_3, v' \cdot v^h) \end{aligned}$$

Output: Eventually, $\mathcal{A}_I$ outputs a forgery signature δ^* on message m^* with respect to (ID^*, PK_{ID^*}). If $F(ID^*) \neq 0$, $\mathcal{B}$ will abort. Otherwise, $u' \prod_{i\in\mathcal{U}} u_i = g^{J(ID^*)}$, and $\mathcal{B}$ retrieves the secret value x^* of ID^* and computes $h^* =$

$H(m^*, ID^*, \delta_2^*, \delta_3^*)$. Since δ^* is a valid certificateless signature on message m^* with respect to ID^* and pk_{ID^*}, we have

$$\begin{aligned} e(\delta_1^*, g) &= X_{ID^*} \cdot e(\delta_2^*, ID^*) \cdot e(\delta_3^*, v^{h^*}) \\ &= e(g^\alpha, g^\beta)^{x^*} \cdot e(\delta_2^*, g^{J(ID^*)}) \cdot e(\delta_3^*, g^{ch^*}) \\ &= e(g, g^{\alpha\beta x^*}) \cdot e((\delta_2^*)^{J(ID^*)}, g) \cdot e((\delta_3^*)^{ch^*}, g) \end{aligned}$$

Therefore $\mathcal{B}$ can output $(\delta_1^*/(\delta_2^*)^{J(ID^*)}(\delta_3^*)^{ch^*}, g^{x^*})$ as the solution to the instance of NGBDH problem.

Now we evaluate the success probability of solving the NGBDH problem. For the simulation to be perfect, we require the following conditions satisfied:
(1) All partial private key extraction queries or all private key extraction queries on an identity ID have $F(ID) \neq 0 \bmod l$.
(2) $F(ID^*) = 0 \bmod p$.
Let $ID_1, ID_2, \ldots, ID_{q_I}$ be the identities appearing in these queries not involving any of the challenge identities. Clearly, $q_I \leq q_{pp} + q_p$. Define the events A^* and A_i as:
$A^* : F(ID^*) = 0 \bmod p$, $A_i : F(ID_i) \neq 0 \bmod l$, $i = 1, 2, \ldots, q_I$.
Thus, the probability of $\mathcal{B}$ not aborting is:
$Pr[\neg abort] \geq Pr[\bigwedge_{i=1}^{q_I} A_i \bigwedge A^*]$.
From $(n+1)l < p$, we conclude that if $F(ID) = 0 \bmod p$ we have $F(ID) = 0 \bmod l$, and if $F(ID) = 0 \bmod l$ there will be unique choice of k with $0 \leq k \leq n$. So we have

$$\begin{aligned} Pr[A^*] = Pr[F(ID^*) = 0 \bmod p] &= Pr[F(ID^*) = 0 \bmod p \bigwedge F(ID^*) = 0 \bmod l] \\ &= Pr[F(ID^*) = 0 \bmod l] Pr[F(ID^*) = 0 \bmod p | F(ID^*) = 0 \bmod l] \\ &= \frac{1}{l}\frac{1}{n+1}. \end{aligned}$$

Since $F(ID_{i_1}) = 0 \bmod l$ and $F(ID_{i_2}) = 0 \bmod l$ $(i_1 \neq i_2)$ are independent, and the events A_i and A^* are also independent, we have

$$\begin{aligned} Pr[\bigwedge_{i=1}^{q_I} A_i \bigwedge A^*] = Pr[A^*] Pr[\bigwedge_{i=1}^{q_I} A_i | A^*] &= Pr[A^*](1 - Pr[\bigvee_{i=1}^{q_I} \neg A_i | A^*] \\ &\geq Pr[A^*](1 - \sum_{i=1}^{q_I} Pr[\neg A_i | A^*)] = \frac{1}{l}\frac{1}{n+1}(1 - \frac{q_I}{l}). \end{aligned}$$

We get

$$\begin{aligned} Pr[\neg abort] &\geq Pr[\bigwedge_{i=1}^{q_I} A_i \bigwedge A^*] \\ &\geq \frac{1}{2(q_{pp} + q_p)(n+1)} \end{aligned}$$

If the simulation does not abort, adversary makes the correct guess with probability $\frac{1}{2} + \epsilon$. Thus, the advantage of $\mathcal{B}$ is at least $\frac{\epsilon}{2(q_{pp}+q_p)(n+1)}$.

The time complexity of the challenger $\mathcal{B}$ is dominated by the exponentiations and multiplications performed in queries. Both the partial private key extraction and private key extraction queries need to do $O(n)$ multiplications and $O(1)$ exponentiations. The computational costs for the signing query are $O(1)$ multiplications and $O(1)$ exponentiations. The public key query needs to carry out $O(1)$ exponentiations. Therefore, the time complexity of $\mathcal{B}$ is $t + O((q_{pp} + q_p)n + q_s)t_m + (q_{pp} + q_p + q_k + q_s)t_e)$.

Theorem 2. The above scheme is $(\epsilon, t, q_k, q_p, q_s)$ secure during Game 2, assuming the (ϵ', t')-Many-DH intractability assumption holds, where $\epsilon' = \frac{\epsilon}{2q_p(n+1)}$ and $t' = t + O(q_p n + q_s)t_m + (q_p + q_k + q_s)t_e)$, where q_k, q_p and q_s respectively

denote the number of queries made to the **Public-Key-Broadcast-Oracle**, the **Private-Key-Extract-Oracle** and the **Sign-Oracle**, t_m and t_e respectively denote the time for a multiplication and an exponentiation in G.

Proof:Assume that $\mathcal{B}$ receives a random instance of Many-DH problem. Given a group G, a generator $g \in G$, and elements $g, g^{\beta}, g^{\gamma}, g^{\alpha\beta}, g^{\alpha\gamma}, g^{\beta\gamma} \in G$, his goal is to output $g^{\alpha\beta\gamma}$. In order to use $\mathcal{A}_{II}$ to solve the problem, $\mathcal{B}$ needs to simulate a challenger and response all the queries for $\mathcal{A}_{II}$.

Setup. The Type II adversary $\mathcal{B}$ chooses $g^{\alpha\beta}$ as the master secret key, and other public parameters are identical to those of Theorem 1. Then $\mathcal{A}_{II}$ sends all public parameters and the master secret key to $\mathcal{A}_{II}$.

Phase 1. $\mathcal{A}_{II}$ can compute partial private key of any identity by itself and carry out the following queries.

Public-Key-Broadcast-Oracle. Upon receiving a query for a public key of an identity ID, if (ID, pk_{ID}) exists in PublicKeyList, $\mathcal{B}$ returns pk_{ID} as the answer. Otherwise, $\mathcal{B}$ performs as follows.

(1) If $ID \neq ID^*$, $\mathcal{B}$ runs the algorithms **UserkeyGen** to generate public key pk_{ID} and private key sk_{ID}. $\mathcal{B}$ adds (ID, pk_{ID}) to PublicKeyList and adds (ID, sk_{ID}) to PrivateKeyList, and returns the public key to $\mathcal{A}_{II}$.

(2) If $ID \doteq ID^*$, $\mathcal{B}$ sets the public key $pk_{ID} = (X_{ID}, \delta_{ID})$ where $X_{ID} = e(g^{\alpha}, g^{\beta\gamma})$, and δ_{ID} can be simulated in the signing oracle of the one-time signature.

Private-Key-Extract-Oracle. Upon receiving a query for a private key of an identity ID, if the PrivateKeyList contains (ID, sk_{ID}), $\mathcal{B}$ returns sk_{ID}. Otherwise, $\mathcal{B}$ performs as follows.

(1) If $ID \neq ID^*$, $\mathcal{B}$ runs the algorithms **UserkeyGen** to adds (ID, pk_{ID}) to PublicKeyList and adds (ID, sk_{ID}) to PrivateKeyList, then $\mathcal{B}$ returns the private key sk_{ID}.

(2) If $ID \doteq ID^*$, $\mathcal{B}$ aborts.

Sign-Oracle: Receiving a signature query for an identity ID and a message m, $\mathcal{B}$ responses the adversary's signature queries as follows :

1. If $ID \neq ID^*$, $\mathcal{B}$ first runs **Private-Key-Extract-Oracle** to generate a private key sk_{ID}, then performs the **Sign** algorithm to create a signature on m for the identity ID.

2. If $ID = ID^*$, $\mathcal{B}$ picks three random values $r_0, r_1, k \in Z_p$, computes $h = H(m, ID, (g^{\gamma})^{r_0} \cdot g^{r_1}, g^{k} \cdot (g^{\alpha\gamma})^{(-1/t)})$ and generates a signature on m as follows

$\delta_1 = (g^{\alpha\gamma})^{-(b+ch/t)} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot (g^{\gamma})^{r_0 \cdot J(ID)} \cdot (v' \cdot v^h)^k$, $\delta_2 = (g^{\gamma})^{r_0} \cdot g^{r_1}$, $\delta_3 = g^k \cdot g_1^{-\gamma/t}$.

It is a valid signature,this is because

$$
\begin{aligned}
\delta_1 &= (g^{\alpha\gamma})^{-(b+ch/t)} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot (g^{\gamma})^{r_0 \cdot J(ID)} \cdot (v' \cdot v^h)^k \\
&= (g_1^{-(b+ch/t)})^{\gamma} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot g^{r_0 \cdot J(ID) \cdot \gamma} \cdot (v' \cdot v^h)^k \\
&= (g_2^{\alpha} \cdot g_2^{-\alpha} \cdot g_1^{-(b+ch/t)})^{\gamma} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot g^{r_0 \cdot J(ID) \cdot \gamma} \cdot (v' \cdot v^h)^k \\
&= (g_2^{\alpha} \cdot (g_2^{t} \cdot g^{b+ch})^{-(\alpha/t)})^{\gamma} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot (g^{J(ID)})^{r_0 \cdot \gamma} \cdot (v' \cdot v^h)^k \\
&= (g_2^{\alpha} \cdot (g_2^{t} \cdot g^{b} \cdot (g^{c})^{h})^{-(\alpha/t)})^{\gamma} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot (g_2^{F(ID)} g^{J(ID)})^{r_0 \cdot \gamma} \cdot (v' \cdot v^h)^k \\
&= (g_2^{\alpha} \cdot (v' \cdot v^h)^{-(\alpha/t)})^{\gamma} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_0 \cdot \gamma} \cdot (v' \cdot v^h)^k \\
&= (g_2^{\alpha} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_0})^{\gamma} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot (v' \cdot v^h)^{k-(\alpha \cdot \gamma/t)}
\end{aligned}
$$

furthermore,

$$
\begin{aligned}
e(\delta_1, g) &= e((g_2^{\alpha} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_0})^{\gamma} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_1} \cdot (v' \cdot v^h)^{k-(\alpha \cdot \gamma/t)}, g) \\
&= e(g_2^{\alpha}, g)^{\gamma} e(u' \prod_{i \in \mathcal{U}} u_i, g^{r_0 \gamma + r_1}) e((v' \cdot v^h)^{k-(\alpha \cdot \gamma/t)}, g) \\
&= X_{ID^*} \cdot e(\delta_2, F_u(ID^*)) \cdot e(\delta_3, v' \cdot v^h)
\end{aligned}
$$

Table 1: Comparison of certificateless signature schemes in the standard model

Schemes	Length	Sign	Verify	Size
Scheme [10]	$3\|G\|$	$6E + (\frac{n_u+n_m}{2} + 3)M_G$	$3P + (\frac{n_u+n_m}{2})M_G + 1E + 2M_{G_T}$	$(n_u + n_m + 5)\|G\|$
Scheme [12]	$4\|G\|$	$6E + (\frac{n_u}{2} + 3)M_G$	$5P + (\frac{n_u}{2} + 1)M_G + 1E + 2M_{G_T}$	$(n_u + 7)\|G\|$
Scheme [13]	$3\|G\|$	$9E + (\frac{n_u+n_m}{2} + 3)M_G$	$5P + (\frac{n_u+n_m}{2})M_G + 2M_{G_T}$	$(n_u + n_m + 4)\|G\|$
Ours scheme	$3\|G\|$	$2E + 2M_G$	$3P + (\frac{n_u}{2} + 1)M_G + 1E + 2M_{G_T}$	$(n_u + 6)\|G\|$

Output: Eventually, $\mathcal{A}_{II}$ outputs a forgery signature δ^* on message m^* with respect to (ID^*, PK_{ID^*}). If $F(ID^*) \neq 0$, $\mathcal{B}$ will abort. Otherwise, $u' \prod_{i \in \mathcal{U}} u_i = g^{J(ID^*)}$. Since δ^* is a valid certificateless signature on message m^* with respect to ID^* and pk_{ID^*}, we have

$$\begin{aligned} e(\delta_1^*, g) &= X_{ID^*} \cdot e(\delta_2^*, ID^*) \cdot e(\delta_3^*, v^{h^*}) \\ &= e(g^\alpha, g^{\beta\gamma}) \cdot e(\delta_2^*, g^{J(ID^*)}) \cdot e(\delta_3^*, g^{ch^*}) \\ &= e(g, g^{\alpha\beta\gamma}) \cdot e((\delta_2^*)^{J(ID^*)}, g) \cdot e((\delta_3^*)^{ch^*}, g) \end{aligned}$$

Therefore $\mathcal{B}$ can output $\delta_1^*/(\delta_2^*)^{J(ID^*)}(\delta_3^*)^{ch^*}$ as the solution to the instance of Many-DH problem.

Now we evaluate the success probability of solving the Many-DH problem. $\mathcal{B}$ will not abort during the proof if the following conditions hold simultaneously.
(1) All private key extraction queries on an identity ID have $ID \neq ID^*$.
(2) $F(ID^*) = 0 \bmod p$.

We can see the probability that $ID \neq ID^*$ during private key extraction queries is $1 - \frac{1}{q_p}$. $Pr[F(ID^*) = 0 \bmod p] = \frac{1}{l}\frac{1}{n+1}$. Let $l = 2q_p - 1$. We get

$$\begin{aligned} Pr[\neg abort] &\geq (1 - \frac{1}{q_p}) \cdot \frac{1}{l}\frac{1}{n+1} \\ &\geq \frac{1}{2q_p(n+1)} \end{aligned}$$

If the simulation does not abort, adversary makes the correct guess with probability $\frac{1}{2} + \epsilon$. Thus, the advantage of $\mathcal{B}$ is at least $\frac{\epsilon}{2q_p(n+1)}$.

The time complexity of the challenger $\mathcal{B}$ is dominated by the exponentiations and multiplications performed in queries. The private key extraction queries need to do $O(n)$ multiplications and $O(1)$ exponentiations. The computational costs for the signing query are $O(1)$ multiplications and $O(1)$ exponentiations. The public key query needs to carry out $O(1)$ exponentiations. Therefore, the time complexity of $\mathcal{B}$ is $t + O(q_p n + q_s)t_m + (q_p + q_k + q_s)t_e)$.

4.2. Performance analysis

To evaluate the performance of different schemes, we use the simple method from [12]. Performance comparison of our scheme and the previous schemes [10, 12, 13] in the standard model is summarized in Table 1, where M_G denotes the multiplication in G, M_{G_T} denotes the multiplication in G_T, E denotes the exponentiation in G and P denotes the pairing computation, 'length' denotes the signature length, 'Size' denotes the number of group elements in G to be included in system parameters. From Table 1, we can get that our scheme has smaller size of system parameters, shorter length of signature, higher computational efficiency than the previous schemes [10, 12, 13] in the standard model.

5. Conclusion

In this paper, we propose a new certificateless signature scheme and prove that our new scheme is existentially unforgeable against adaptively chosen message attack in the standard model. Compared with the previous schemes [10, 12, 13] in the standard model, our new scheme offers shorter system parameters, shorter length of signature, and higher computational efficiency.

References

[1] S. Al-Riyami, K. Paterson, Certificateless public key cryptography, Advances in Cryptology-ASIACRYPT 2003,Springer (2003) 452–473.

[2] M. Bellare, A. Boldyreva, A. Palacio, An uninstantiable random-oracle-model scheme for a hybrid-encryption problem, in: Advances in Cryptology-EUROCRYPT 2004, LNCS 3027, Springer-Verlag, pp. 171–188.

[3] M. Bellare, P. Rogaway, Random oracles are practical: A paradigm for designing efficient protocols, in: Proceedings of the 1st ACM conference on Computer and communications security,Fairfax, Virginia, USA, pp. 62–73.

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

[5] M. Gorantla, A. Saxena, An efficient certificateless signature scheme, in: ACIS 2005. LNCS, vol. 3802, Springer, pp. 110–116.

[6] X. Li, K. Chen, L. Sun, Certificateless signature and proxy signature schemes from bilinear pairings, Lithuanian Mathematical Journal 45 (2005) 76–83.

[7] J. Liu, M. Au, W. Susilo, Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model, in: Proceedings of the 2nd ACM symposium on Information, computer and communications security, ACM, pp. 273–283.

[8] A. Shamir, Identity-based cryptosystems and signature schemes, in: Advances in Cryptology-Crypto 1984, LNCS, vol. 196, Springer-Verlag, Berlin, pp. 47–53.

[9] Q. Xia, C. Xu, Y. Yu, Key replacement attack on two certificateless signature schemes without random oracles, Key Engineering Materials 439-440 (2010) 1606–1611.

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

[11] W.S. Yap, S.H. Heng, B.M. Goi, An efficient certificateless signature scheme, in: EUC workshops 2006, LNCS, vol.4097,Springer, pp. 322–331.

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

[13] Y. Yuan, D. Li, L. Tian, Z. H., Certificateless signature scheme without random oracles, in: ISA 2009, LNCS vol.5576, Springer, pp. 31–40.

[14] J. Zhang, J. Mao, An efficient rsa-based certificateless signature scheme, The Journal of Systems and Software 85 (2012) 638–642.

[15] Z. Zhang, D. Wong, J. Xu, D. Feng, Certificateless public-key signature: security model and efficient construction, in: ACNS'06, LNCS, vol. 3989, Springer, pp. 293–308.