Certificateless Ring Signcryption Scheme from Pairings

Hui Guo and Lunzhi Deng
(Corresponding author: Lunzhi Deng)

School of Mathematical Sciences, Guizhou Normal University
Guiyang 550001, China
(Email: denglunzhi@163.com)

(Received June 23, 2018; Revised and Accepted Nov. 22, 2018; First Online June 22, 2019)

Abstract

Signcryption is a useful primitive which simultaneously provides the functions of encryption and signature. Certificateless cryptography not only eliminates the key escrow property, but also removes certificates. In a ring signcryption scheme, an entity can anonymously signcrypt a message on behalf of ring members including himself. In this paper, a new certificateless ring signcryption (CLRSC) scheme is proposed, and it is proved to be secure in the random oracle model. In the scheme, it requires only one bilinear pairing operation in signcryption, and three bilinear pairing operations in unsigncryption. To the best of our knowledge, our scheme is more efficient than previous ones in computation.

Keywords: Certificateless Cryptography; Pairing; Random Oracle Model; Ring Signcryption

1 Introduction

Public key cryptography [16] is an important technique to realize network and information security. Traditional public key infrastructure (PKI) [1,3,8,20] needs a trusted certification authority (CA) to issue a certificate binding the identity and the public key of the user. Hence, the management problem of public key certificates arises. To solve the problem, Shamir [27] defined a idea of identity-based cryptography in 1984. In the identity-based cryptography [14,18], a trusted third party called the private key generator (PKG) generates all user's private keys, which bring a new problem of the key escrow.

In 2003, Al-Riyami *et al.* [2] introduced the concept of certificateless public key cryptography (CL-PKC). In CL-PKC, a user's private key is made up of partial private key generated by key generation center (KGC) [11,19,25] and a secret value selected by the user separately. So even if the malicious KGC leaks the partial private key created by KGC, the attacker also cannot get the entire private key to decrypt the associated ciphertext. Through this, certificateless cryptography not only eliminates the key escrow property, but also removes certificates.

Ring signature was first defined by Rivest *et al.* [23] in 2001. In a ring signature scheme, a signer can select some members to form a ring and produce a ring signature without the assitance of the other ring members. Any verifier can know that the message comes from a member of ring, but doesn't know exactly who the signer is. So it has a lot of important applications for revealing secrets. Some valuable information was found in the study of ring signature [4, 7, 10, 17, 21, 24]. Ring signcryption [15] is a cryptographic primitive motivated by ring signature. In a ring signcryption scheme, a user can anonymously signcrypt a message on behalf of ring members including himself. It is helpful for leaking secrets in an anonymous, authenticated and confidential way.

Huang *et al.* [13] extended ring signature to ring signcryption and proposed a concrete scheme in the identity-based cryptosystem, but the ciphertext of their scheme is too long. In 2009, Zhu *et al.* [33] proposed an efficient and provable secure identity based ring signcryption scheme. But Selvi *et al.* [26] pointed out that the scheme [33] is not semantically secure. Other schemes proposed including generalized ring signcryption [32], attribute-based ring signcryption [9,31], threshold ring signcryption [5], *etc.*

In 2007, Wang *et al.* [30] constructed a certificateless ring signcryption scheme, which is proved to be secure. Their scheme needs 3n+5 pairing operations. Zhu *et al.* [34] proposed a provably secure parallel certificateless ring signcryption scheme, but they did not give the concrete proof about security. In 2011, Qi *et al.* [22] proposed a provably secure certificateless ring signcryption scheme. In 2015, Sharma *et al.* [28] constructed a pairing-free certificateless ring signcryption scheme (PF-CLRSC). However, Shen *et al.* [29] pointed out that the scheme [28] is not secure in 2017.

In this paper, we propose a new certificateless ring signcryption scheme which has the following features:

1) The proposed scheme is proved to be secure in the random oracle model.

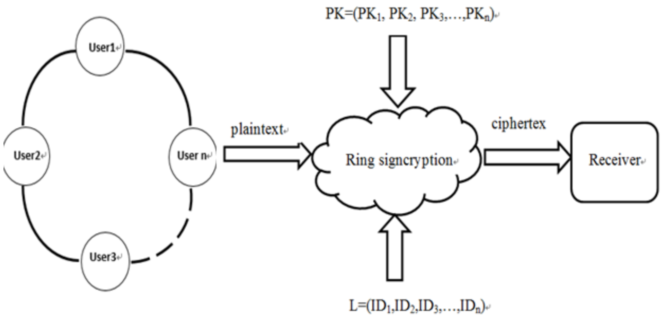

Figure 1: Process of a CLRSC scheme

2) The proposed scheme requires only 4 pairing operations and it is more efficient than the schemes [22,30,34] in computation.

2 Preliminaries

2.1 Bilinear Pairing

Let G_1 be an additive group of prime order q and G_2 be a multiplicative group of the same order. And P is a generator of G_1. Let $e : G_1 \times G_1 \rightarrow G_2$ be a map with the following properties:

- Bilinearity: $e(aP_1, bP_2) = e(P_1, P_2)^{ab}$ for all $P_1, P_2 \in G_1$ and $a, b \in Z_q^*$.
- Non-degeneracy: There exist $P_1, P_2 \in G_1$ such that $e(P_1, P_2) \neq 1_{G_2}$.
- Computability: There is an efficient algorithm to compute $e(P_1, P_2)$ for all $P_1, P_2 \in G_1$.

Definition 1. *Given a generator P of a group G_1 and a tuple $(aP, bP, cP, X \in G_2)$ for unknown $a, b, c \in Z_q^*$, the Decisional Bilinear Diffie-Hellman problem (DBDHP) is to decide whether $X = e(P, P)^{abc}$.*

Definition 2. *Given a generator P of group G_1 and a tuple (aP, bP) for unknown $a, b \in Z_q^*$, the computational Diffie-Hellman problem (CDHP) is to compute abP.*

Definition 3. *Given a generator Q of group G_3 with prime order p, and a tuple $(aQ, bQ, X \in G_3)$ for unknown $a, b \in Z_q^*$, the Decisional Diffie-Hellman problem (DDHP) is to decide whether $X = abQ$.*

Definition 4. *Given a generator Q of group G_3 with prime order p, and an elements aQ, the discrete logarithm problem (DLP) is to compute a.*

2.2 Model of Certificateless Ring Signcryption

A certificateless ring signcryption scheme (CLRSC) is composed of six polynomial time algorithms, it is defined as follows:

- Setup: Input a security parameter ν, KGC outputs the system parameters $params$ and a master secret key msk.
- Partial-Private-Key-Extract: Input the system parameters $params$, the master secret key msk and the identity $ID_i \in \{0,1\}^*$, KGC returns the user's partial private key D_i.
- Secret-Value-Set: The user ID_i randomly chooses a secret value $t_i \in Z_q^*$.
- User-Public-Key-Generate: Input the system parameters $params$, the user's secret value t_i and identity $ID_i \in \{0,1\}^*$, this algorithm outputs the public key T_i. It is run by user himself.
- Signcryption: To send the message m to the receiver ID_r, the actual signcrypter ID_s selects $n-1$ other users to form n users ring L including himself and represents members of the ring L to give a ciphertext σ on the message m.
- Unsigncryption: After receiving the ciphertext (σ, L), the receiver ID_r decrypts the ciphertext and obtains the message m or the symbol $\perp$ if σ was a invalid ciphertext.

Definition 5.

A CLRSC scheme is said to be indistinguishable under adaptive chosen ciphertext attacks (IND-CLRSC- CCA2) if the polynomial bounded adversary with a negligible advantage in the following game.

Game I. A challenger $\mathscr{C}$ and a Type I adversary $\mathscr{A}_1$ play the following game.

Initialization. $\mathscr{C}$ runs the setup algorithm to generate a master secret key msk and the public system parameters $params$. $\mathscr{C}$ sends $params$ to $\mathscr{A}_1$. ($\mathscr{A}_1$ does not know msk).

Phase 1. $\mathscr{A}_1$ makes a polynomially bounded number of adaptive queries to $\mathscr{C}$.

- Hash functions query: $\mathscr{A}_1$ can query the values of any hash functions.
- Partial private key query: $\mathscr{A}_1$ chooses a user's identity ID_i, $\mathscr{C}$ runs this algorithm to generate the corresponding partial private key D_i, and sends to $\mathscr{A}_1$.
- User public key query: $\mathscr{A}_1$ chooses an identity ID_i, $\mathscr{C}$ returns public key T_i generated by the public key algorithm.
- User public key replacement: $\mathscr{A}_1$ chooses an identity ID_i and a new public key value T_i', $\mathscr{A}_1$ replaces the current public key T_i of the user ID_i with T_i'.
- Secret value query: $\mathscr{A}_1$ chooses an identity ID_i, $\mathscr{C}$ returns the corresponding secret value t_i to $\mathscr{A}_1$. If public key of the user ID_i was replaced, $\mathscr{A}_1$ cannot ask for the secret value of the user ID_i.

- Signcryption query: $\mathscr{A}_1$ chooses a message m, a receiver ID_r and a set $R = L\bigcup\{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users' identities, and sends to $\mathscr{C}$. $\mathscr{C}$ returns the ciphtext σ to $\mathscr{A}_1$.
- Unsigncryption query: When $\mathscr{A}_1$ chooses a ciphertext σ, a receiver's identity ID_r and a set $L = \{ID_1, \cdots, ID_n\}$, $\mathscr{C}$ outputs plaintext m or the symbol $\perp$ if σ is an invalid ciphertext.

Challenge. $\mathscr{A}_1$ sends following information to the challenger: two equal length messages m_0, m_1, a specified receiver ID_r, a set $R = L\bigcup\{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users, and fulfills the following conditions:

1) $\mathscr{A}_1$ should not have queried the partial private key to ID_r in Phase 1.
2) There exists at least a member $ID_s \in L$ whose public key has not been replaced by $\mathscr{A}_1$.

$\mathscr{C}$ takes randomly a bit $\mu \in \{0, 1\}$ and computes the ciphertext σ^* on the message m_μ under the set R.

Phase 2. $\mathscr{A}_1$ performs a polynomially bounded number of queries just like in Phase 1, and fulfills the following restrictions:

1) $\mathscr{A}_1$ can not have requested the partial private key for ID_r.
2) $\mathscr{A}_1$ can not have made the unsigncryption queries for the ciphertext σ^*.

Response. $\mathscr{A}_1$ outputs a bit μ' and wins the game if $\mu' = \mu$.

The advantage of $\mathscr{A}_1$ is defined as : $Adv_{\mathscr{A}_1}^{IND-CLRSC}(\nu) = |2\Pr[\mu' = \mu] - 1|$.

Game II. A Type II adversary $\mathscr{A}_2$ for a CLRSC scheme plays the following game with a challenger $\mathscr{C}$.

Initialization. $\mathscr{C}$ runs the setup algorithm to generate the master secret key msk and public system parameters $params$, then sends $params$ and msk to $\mathscr{A}_2$.

Phase 1. Same as that in the Game I.

Challenge. $\mathscr{A}_2$ sends following information to the challenger: two equal length messages m_0, m_1, a specified receiver ID_r and a set $R = L\bigcup\{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users, and fulfills the following restrictions:

1) $\mathscr{A}_2$ can not have requested the secret value for ID_r in Phase 1.
2) $\mathscr{A}_2$ can not have replaced the user public key corresponding to ID_r in Phase 1.
3) There exists at least a member $ID_s \in L$ whose public key has not been replaced by $\mathscr{A}_2$.

$\mathscr{C}$ takes randomly a bit $\mu \in \{0, 1\}$ and computes the ciphertext σ^* on m_μ under the set R.

Phase 2. $\mathscr{A}_2$ performs a polynomially bounded number of queries just like in Phase 1, and fulfills the following conditions:

1) $\mathscr{A}_2$ can not have requested the secret value for ID_r.
2) $\mathscr{A}_2$ can not have made the unsigncryption queries for the ciphertext σ^*.

Response. $\mathscr{A}_2$ outputs a bit μ' and wins the game if $\mu' = \mu$.

The advantage of $\mathscr{A}_2$ is defined as: $Adv_{\mathscr{A}_2}^{IND-CLRSC}(\nu) = |2\Pr[\mu' = \mu] - 1|$.

Definition 6. *CLRSC is said to be unforgeable under adaptive chosen message attacks(EUF-CLRSC-CMA2) if the polynomial bounded adversary with a negligible advantage in the following game.*

Game III. Challenger $\mathscr{C}$ and type I adversary $\mathscr{A}_1$ play the following game:

Initialization, Query. Same as that in the Game I.

Forge. $\mathscr{A}_1$ produces a new ciphertext (σ, ID_r, R).

When the following conditions hold, $\mathscr{A}_1$ wins the game.

1) The symbol $\perp$ is not returned by unsigncryption query.
2) $\mathscr{A}_1$ cannot ask for the partial private keys of the users in L.
3) The forged ciphertext (σ, ID_r, R) is not obtained by signcryption query.

The advantage of $\mathscr{A}_1$ is defined as: $Adv_{\mathscr{A}_1}^{UNF-CLRSC}$ $=\Pr[\mathscr{A}_1\ win]$.

Game IV. Challenger $\mathscr{C}$ and type II adversary $\mathscr{A}_2$ play the following game:

Initialization, Query. Same as that in the Game II.

Forge. $\mathscr{A}_2$ produces a new ciphertext (σ, ID_r, R). When the following conditions hold, $\mathscr{A}_2$ wins the game.

1) The symbol $\perp$ is not returned by unsigncryption query.
2) $\mathscr{A}_2$ can not request the secret value of the users in L and replace the user public key of the members in L.
3) The forged ciphertext (σ, ID_r, R) is not obtained by signcryption query.

The advantage of $\mathscr{A}_2$ is defined as : $Adv_{\mathscr{A}_2}^{UNF-CLRSC}$ $=\Pr[\mathscr{A}_2\ win]$.

Definition 7. *A CLRSC scheme is anonymous if for any message m, any ring $L = \{ID_1, \cdots, ID_n\}$, receiver ID_r and ciphertext σ. The receiver ID_r ($ID_r \notin L$), even with unbounded computing resources, can identify the actual signcrypter with probability no better than $\frac{1}{n}$.*

3 Proposed Scheme

- Setup: Given the security parameter of the system ν, KGC chooses groups $G_1 = \langle P \rangle$, G_2 and $G_3 = \langle Q \rangle$ of prime order $q > 2^{\nu}$, and a bilinear pairing $e : G_1 \times G_1 \to G_2$. Then KGC chooses four hash function $H_1 : \{0,1\}^* \to G_1$, $H_2 : \{0,1\}^* \to \{0,1\}^l$, $H_3, H_4 : \{0,1\}^* \to Z_q^*$. The message space is $\Omega = \{0,1\}^l$. KGC randomly chooses its secret key $x \in Z_q^*$ and sets $P_{pub} = xP$ as its system public key. KGC publishes system parameters : $params = \{G_1, G_2, G_3, q, e, P, Q, P_{pub} = xP, H_1, H_2, H_3, H_4\}$.

- Partial-Private-Key-Extract: Given a user's identity $ID_i \in \{0,1\}^*$, KGC computes $E_i = H_1(ID_i), D_i = xE_i$ and sends D_i to the user via a secure channel.

- Secret value set: The user ID_i selects at random $t_i \in Z_q^*$ as his/her secret value.

- User public key generate: The user ID_i sets $T_i = t_iQ$ as his/her public key.

- Signcryption: Let $R = L \bigcup \{T_i, ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users' identities. The actual signcrypter $ID_s \in L$ outputs a ciphertext σ on the message m and sends it to the receiver ID_r as following:

  1) Randomly selects $\lambda_1, \lambda_2 \in Z_q^*$, computes $B_1 = \lambda_1 P$, $B_2 = \lambda_2 Q$, $U_1 = e(\lambda_1 P_{pub}, E_r)$, $U_2 = \lambda_2 T_r$, $C = H_2(R, U_1, U_2) \bigoplus m$.
  2) Randomly selects $A_i \in G_1, c_i \in Z_q^*$, computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, \cdots, s-1, s+1, \cdots, n$.
  3) Randomly selects $\delta_1 \in Z_q^*$, computes $A_s = \delta_1 E_s - \sum_{i=1, i\neq s}^{n}(A_i + h_iE_i)$.
  4) Randomly selects $\delta_2 \in Z_q^*$, computes $y = H_4(m, R, U_1, U_2, \delta_2 Q + \sum_{i=1,i\neq s}^{n} c_iT_i, \bigcup_{i=1}^{n}\{A_i\})$.
  5) Computes $c_s = y - \sum_{i=1,i\neq s}^{n} c_i \pmod q$, $h_s = H_3(m, R, U_1, U_2, T_s, ID_s, A_s, c_s)$.
  6) Computes $z = \delta_2 - c_s t_s \pmod q$, $V = (\delta_1 + h_s)D_s$.
  7) Outputs the ciphertext : $\sigma = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$.

- Unsigncryption: On receiving the ciphertext $\sigma = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$, the receiver ID_r decrypts the ciphertext as follows:

  1) Computes $U_1 = e(B_1, D_r)$, $U_2 = t_r B_2$, $m = C \bigoplus H_2(R, U_1, U_2)$.
  2) Checks if $\sum_{i=1}^{n} c_i = H_4(m, R, U_1, U_2, zQ + \sum_{i=1}^{n} c_iT_i, \bigcup_{i=1}^{n}\{A_i\})$. Proceed if the equality holds, reject otherwise.
  3) Computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, \cdots, n$.
  4) Checking whether $e(P, V) = e(P_{pub}, \sum_{i=1}^{n}(A_i + h_iE_i))$. If the equality holds, accepts m as a valid message. Otherwise, it returns $\perp$

4 Analysis of Proposed Scheme

4.1 Correctness Analysis

$$\begin{aligned}
e(P,V) &= e(P, (\delta_1 + h_s)D_s) \\
&= e(P, (\delta_1 + h_s)xE_s) \\
&= e(xP, (\delta_1 + h_s)E_s) \\
&= e(P_{pub}, \delta_1 E_s + h_sE_s) \\
&= e(P_{pub}, A_i + \sum_{i=1, i\neq s}^{n}(A_i + h_iE_i) + h_sE_s) \\
&= e(P_{pub}, \sum_{i=1}^{n}(A_i + h_iE_i)); \\
U_2 &= t_rB_2 = t_r\lambda_2 Q = \lambda_2 t_r Q = \lambda_2 T_r; \\
U_1 &= e(B_1, D_r) \\
&= e(\lambda_1 P, xE_r) \\
&= e(\lambda_1 xP, E_r) \\
&= e(\lambda_1 P_{pub}, E_r); \\
\sum_{i=1}^{n} c_i &= y \\
&= H_4(m, R, U_1, U_2, zQ \\
&\quad + \sum_{i=1}^{n} c_iT_i, \bigcup_{i=1}^{n}\{A_i\}); \\
\delta_2 Q + \sum_{i=1,i\neq s}^{n} c_iT_i &= (z + c_st_s)Q + \sum_{i=1,i\neq s}^{n} c_iT_i \\
&= zQ + c_sT_s + \sum_{i=1,i\neq s}^{n} c_iT_i \\
&= zQ + \sum_{i=1}^{n} c_iT_i.
\end{aligned}$$

4.2 Security Analysis

Theorem 1. *In random oracle model, the scheme is indistinguishable against IND-CLRSC-CCA2 adversary $\mathscr{A}_1$ if the DBDHP is hard.*

Proof. Assume that the challenger $\mathscr{C}$ receives an instance (P, aP, bP, cP, X) of the DBDHP, the goal of $\mathscr{C}$ is to determine whether $X = e(P,P)^{abc}$ or not. $\mathscr{C}$ runs $\mathscr{A}_1$ as a subroutine and plays the role of the challenger in Game I.

Initialization. $\mathscr{C}$ runs the setup algorithm to generate system parameters. Then $\mathscr{C}$ sends the system parameters $params = \{G_1, G_2, G_3, q, e, P, Q, P_{pub} = aP, H_1, H_2, H_3, H_4\}$ to $\mathscr{A}_1$. ($\mathscr{A}_1$ does not know the value a).

Phase 1.Without losing generality, assuming that each query is different. $\mathscr{A}_1$ will ask for $H_1(ID_i)$ before the identity ID_i is used in any other queries. $\mathscr{C}$ will maintain some lists to store the queries and answers, all of the lists are initially empty.

- H_1 queries: $\mathscr{C}$ maintains the list L_1 of tuple (ID_i, d_i). When $H_1(ID_i)$ is queried by $\mathscr{A}_1$, $\mathscr{C}$ answers the query H_1 as follows.

  At the j^{th} H_1 query, $\mathscr{C}$ sets $H_1(ID^*) = bP$. For $i \neq j$, $\mathscr{C}$ selects a random $d_i \in Z_q^*$ and sets $H_1(ID_i) = d_iP$, the query and the respond will be stored in the list L_1.

- H_2 queries: $\mathscr{C}$ maintains the list L_2 of tuple (α_i, h_i). When $H_2(\alpha_i)$ is queried by $\mathscr{A}_1$, $\mathscr{C}$ selects a random $h_i \in \{0,1\}^l$, sets $H_2(\alpha_i) = h_i$ and adds (α_i, h_i) to list L_2.

- H_3 queries: $\mathscr{C}$ maintains the list L_3 of tuple (β_i, c_i). When $H_3(\alpha_i)$ is queried by $\mathscr{A}_1$, $\mathscr{C}$ selects a random $c_i \in Z_q^*$, sets $H_3(\beta_i) = c_i$ and adds (β_i, c_i) to list L_3.

- H_4 queries: $\mathscr{C}$ maintains the list L_4 of tuple $(\beta_i^{'}, c_i^{'})$. When $H_4(\alpha_i)$ is queried by $\mathscr{A}_1$, $\mathscr{C}$ selects a random $c_i^{'} \in Z_q^*$, sets $H_4(\beta_i^{'}) = c_i^{'}$ and adds $(\beta_i^{'}, c_i^{'})$ to list L_4.

- User public key queries: $\mathscr{C}$ maintains the list L_U of tuple (ID_i, t_i). When $\mathscr{A}_1$ makes this query, $\mathscr{C}$ picks a random $t_i \in Z_q^*$, sets $T_i = t_iQ$ and adds (ID_i, t_i) to list L_U.

- User public key replacement requests: $\mathscr{C}$ maintains the list L_R of tuple (ID_i, T_i, T_i'). When $\mathscr{A}_1$ makes this query, $\mathscr{C}$ replaces the current public key value T_i with a new value T_i' and adds (ID_i, T_i, T_i') to list L_R.

- Partial private key queries: $\mathscr{C}$ maintains the list L_D of tuple (ID_i, D_i). When $\mathscr{A}_1$ makes this query, $\mathscr{C}$ does as follows:

  If $ID_i = ID^*$, $\mathscr{C}$ fails and stops. Otherwise $\mathscr{C}$ looks up the tuple (ID_i, d_i) in list L_1, responds with $D_i = d_i \cdot (aP)$ and adds (ID_i, D_i) to list L_D.

- Secret value queries: $\mathscr{C}$ maintains the list L_E of tuple (ID_i, t_i). When $\mathscr{A}_1$ makes this query, $\mathscr{C}$ checks list L_U. If there exists the tuple (ID_i, t_i) in list L_U, $\mathscr{C}$ answers with t_i. Otherwise, $\mathscr{C}$ selects a random $t_i \in Z_q^*$, answers with t_i and adds (ID_i, t_i) to lists L_E and L_U.

- Signcryption queries: $\mathscr{A}_1$ selects a message m, a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n users' identities and a receiver ID_r and sends them to $\mathscr{C}$. $\mathscr{C}$ returns a signcryption as follows:

  If there exists an identity $ID_s \in L$ such that $ID_s \neq ID^*$ and $ID_s \notin L_R$, $\mathscr{C}$ gives a signcryption σ by calling the signcryption algorithm to answer $\mathscr{A}_1$, where ID_s is the actual signer. Otherwise, $\mathscr{C}$ does the following steps:

  1) Randomly selects $\lambda_1, \lambda_2 \in Z_q^*$, computes $B_1 = \lambda_1 P$, $B_2 = \lambda_2 Q$, $U_1 = e(\lambda_1 P_{pub}, E_r)$, $U_2 = \lambda_2 T_r$, $C = H_2(R, U_1, U_2) \bigoplus m$.
  2) Randomly selects $A_i \in G_1, c_i \in Z_q^*$, computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, s-1, s+1, \cdots, n$.
  3) Randomly selects $z, c_s \in Z_q^*$, computes $T = zQ + \sum_{i=1}^{n} c_i T_i$.
  4) Randomly selects $r, h_s \in Z_q^*$, computes $A_s = rP - h_s E_s - \sum_{i=1, i\neq s}^{n}(A_i + h_i E_i), V = r(aP)$.
  5) Stores the relations: $\sum_{i=1}^{n} c_i = H_4(m,\ R,\ U_1,\ U_2,\ T,\ \bigcup_{i=1}^{n}\{A_i\})$, $h_s = H_3(m,\ R,\ U_1,\ U_2, T_s, ID_s, A_s, c_s)$.

     If collision occurs, repeats Steps (1)-(5).
  6) Outputs the ciphertext: $\sigma \doteq \{z,\ V, B_1,\ B_2,\ C,\ \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$.

- Unsigncryption queries: $\mathscr{A}_1$ picks ciphertext $\sigma = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$, a set $R = L \bigcup \{T_i : ID_i \in L\}$ and a receiver ID_r. If $ID_r \neq ID^*$ and $ID_r \notin L_R$, $\mathscr{C}$ gives a message m by calling the unsigncryption algorithm. Otherwise, $\mathscr{C}$ notifies that σ is an invaild ciphertext.

Challenge. $\mathscr{A}_1$ chooses two equal length messages m_0, m_1, a specified receiver ID_r, and a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of ring members, and sends them to the challenger $\mathscr{C}$. ($\mathscr{A}_1$ should not have queried the partial private key for ID_r in Phase 1). If $ID_r \neq ID^*$, $\mathscr{C}$ fails and stops. Otherwise, $\mathscr{C}$ picks $\mu \in \{0,1\}$, and computes ciphertext σ^* on the message M_μ under the set R as follows:

1) Randomly selects $c, \lambda_2 \in Z_q^*$, computes $B_1 = cP$, $B_2 = \lambda_2 Q$, $U_1 = X$, $U_2 = \lambda_2 T_r$, $C = H_2(R, X, U_2) \bigoplus m$.
2) Randomly selects $A_i \in G_1, c_i \in Z_q^*$, computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, \cdots, s-1, s+1, \cdots, n$.
3) Randomly selects $\delta_1 \in Z_q^*$, computes $A_s = \delta_1 E_s - \sum_{i=1, i\neq s}^{n}(A_i + h_i E_i)$.
4) Randomly selects $\delta_2 \in Z_q^*$, computes $y = H_4(m,\ R,\ U_1, U_2, \delta_2 Q + \sum_{i=1, i\neq s}^{n} c_i T_i, \bigcup_{i=1}^{n}\{A_i\})$.
5) Computes $c_s = y - \sum_{i=1, i\neq s}^{n} c_i \pmod q$. $h_s = H_3(m, R, U_1, U_2, T_s, ID_s, A_s, c_s)$.
6) Computes $z = \delta_2 - c_s t_s \pmod q$, $V = (\delta_1 + h_s) D_s$.
7) Outputs the ciphertext: $\sigma^* = \{z,\ V,\ B_1,\ B_2,\ C,\ \bigcup_{i=1}^{n}\{A_i\},\ \bigcup_{i=1}^{n}\{c_i\}\}$.

Phase 2. $\mathscr{A}_1$ makes a polynomially bounded number of queries just like in the Phase 1 (but $\mathscr{A}_1$ should not have queried the partial private key for ID_r and requested the plaintext corresponding to the ciphertext σ^*).

Response. $\mathscr{A}_1$ outputs $\mu' \in \{0,1\}$. If $\mu' \doteq \mu$, $\mathscr{C}$ outputs 1. Otherwise, $\mathscr{C}$ outputs 0. If $X = e(P,P)^{abc}$, σ^* is a valid ciphertext. Then $\mathscr{A}_1$ can distinguish μ with the advantage ε. So $\Pr[\mathscr{C} \longrightarrow 1 | X \doteq e(P,P)^{abc}] \doteq \Pr[\mu' \doteq \mu | X \doteq e(P,P)^{abc}] \doteq \frac{1}{2} + \varepsilon$.

If $X \neq e(P,P)^{abc}$, when $\mu = 0$ or $\mu = 1$, each part of the ciphertext has the same probability distribution, so $\mathscr{A}_1$ has no advantage to distinguishing μ. So

$\Pr[\mathscr{C} \longrightarrow 1 | X \neq e(P,P)^{abc}] \doteq \Pr[\mu' \doteq \mu | X \neq e(P,P)^{abc}] \doteq \frac{1}{2}$.

Probability. Let $q_{H_i}(i = 1,2,3,4)$, q_U, q_R, q_D and q_S be the number of $H_i(i = 1,2,3,4)$ queries, user public key queries, user public key replacement requests, partial private key queries and signcryption queries, respectively.

Without loss of generality, we may assume that $L_E \cap L_R = \emptyset$, and denote some events as follows: π_1: $\mathscr{C}$ does not fail in partial private key queries; π_2: $\mathscr{C}$ does not fail in unsigncryption queries; π_3: $\mathscr{C}$ does not fail in challenge stage. It is easy to get following results:
$\Pr[\pi_1]=1 - \frac{q_D}{q_{H_1}}$, $\Pr[\pi_2]=1 - \frac{q_U}{2^\nu}$, $\Pr[\pi_3]=\frac{1}{q_{H_1} - q_D}$.

$$\begin{aligned}\Pr[\mathscr{C}\ success] &= \Pr[\pi_1 \wedge \pi_2 \wedge \pi_3] \\ &= \Pr[\pi_1] \cdot \Pr[\pi_2] \cdot \Pr[\pi_3] \\ &= (1 - \frac{q_D}{q_{H_1}}) \cdot (1 - \frac{q_U}{2^\nu}) \cdot \frac{1}{q_{H_1} - q_D} \\ &\approx \frac{1}{q_{H_1}}\end{aligned}$$

Therefore, if $\mathscr{A}_2$ can succeed with the probability ε, then $\mathscr{C}$ can solve the DBDHP with probability $\frac{\varepsilon}{q_{H_1}}$. □

Theorem 2. *In the random oracle model, the scheme is indistinguishable against IND-CLRSC-CCA2 adversary $\mathscr{A}_2$ if the DDHP is hard.*

Proof. Assume that the challenger $\mathscr{C}$ receives an instance (aQ, bQ, Y) of the DDHP, the goal of $\mathscr{C}$ is to determine whether $Y = abQ$ or not. $\mathscr{C}$ runs $\mathscr{A}_2$ as a subroutine and plays the role of the challenger in Game II.

Initialization. $\mathscr{C}$ performs the setup algorithm with the parameter ν, then sends the system parameters $params = \{G_1, G_2, G_3, q, e, P, Q, P_{pub} = xP, H_1, H_2, H_3, H_4\}$ and master secret key $msk = \{x\}$ to $\mathscr{A}_2$.

Phase 1. Without losing generality, assuming that each query is different. $\mathscr{A}_1$ will ask for $H_1(ID_i)$ before the identity ID_i is used in any other queries. $\mathscr{C}$ will maintain some lists to store the queries and answers, all of the lists are initially empty.

- H_1 queries: $\mathscr{C}$ maintains the list L_1 of tuple (ID_i, d_i). When $\mathscr{A}_2$ makes a query $H_1(ID_i)$, $\mathscr{C}$ randomly picks $d_i \in Z_q^*$, sets $H_1(ID_i) = d_iP$ and adds (ID_i, d_i) to list L_1.

- H_2, H_3 and H_4 queries: Same as those in the proof of Theorem 1.

- User public key queries: $\mathscr{C}$ maintains the list L_U of tuple (ID_i, t_i). When $\mathscr{A}_2$ makes this query, $\mathscr{C}$ responds as follows:

  At the j^{th} query, $\mathscr{C}$ sets $ID_j = ID^*$, $T^* = aQ$. For $i \neq j$, $\mathscr{C}$ randomly picks $t_i \in Z_q^*$, returns $T_i = t_iQ$ and adds (ID_i, t_i) to list L_U.

- User public key replacement requests: Same as that in the proof of Theorem 1.

- Partial private key queries: $\mathscr{C}$ maintains the list L_D of tuple (ID_i, D_i). When $\mathscr{A}_2$ makes this query, $\mathscr{C}$ finds the tuple (ID_i, d_i) in list L_1, responds with $D_i = d_i(xP)$ and adds (ID_i, D_i) to list L_D.

- Secret value queries: $\mathscr{C}$ maintains the list L_E of tuple (ID_i, t_i). When $\mathscr{A}_2$ makes this query, $\mathscr{C}$ does as follows:

  If $ID_i = ID^*$, $\mathscr{C}$ fails and stops. Otherwise, $\mathscr{C}$ looks up (ID_i, t_i) in list L_U, responds with t_i and adds (ID_i, t_i) to list L_E.

- Signcryption, Unsigncryption queries: Same as that in the proof of Theorem 1.

Challenge. $\mathscr{A}_2$ chooses two equal length messages m_0, m_1, and a specified receiver ID_r, a set $R = L \bigcup \{T_i : ID_i \in L\}$, where $L = \{ID_1, \cdots, ID_n\}$ is the set of n ring members, and sends them to the challenger $\mathscr{C}$.($\mathscr{A}_2$ should not have queried the secret value for ID_r). if $ID_r \neq ID^*$, $\mathscr{C}$ fails and stops. Otherwise, $\mathscr{C}$ picks $\mu \in \{0,1\}$, and computes ciphertext σ^* on message M_μ under the set R as follows:

1) Randomly chooses $\lambda_1, b \in Z_q^*$, computes $B_1 = \lambda_1 P$, $B_2 = bQ$, $U_1 = e(\lambda_1 P_{pub}, E_r)$, $U_2 = Y$, $C = H_2(R, U_1, Y) \bigoplus m$.

2) Randomly chooses $A_i \in G_1, c_i \in Z_q^*$, computes $h_i = H_3(m, R, U_1, U_2, T_i, ID_i, A_i, c_i)$, $i = 1, 2, \cdots, s-1, s+1, \cdots, n$.

3) Randomly chooses $\delta_1 \in Z_q^*$, computes $A_s = \delta_1 E_s - \sum_{i=1, i \neq s}^n (A_i + h_i E_i)$.

4) Randomly chooses $\delta_2 \in Z_q^*$, computes $y = H_4(m, R, U_1, U_2, \delta_2 Q + \sum_{i=1, i\neq s}^n c_i T_i, \bigcup_{i=1}^n \{A_i\})$.

5) Computes $c_s = y - \sum_{i=1, i\neq s}^n c_i \pmod q$, $h_s = H_3(m, R, U_1, U_2, T_s, ID_s, A_s, c_s)$.

6) Computes $z = \delta_2 - c_s t_s \pmod q$, $V = (\delta_1 + h_s) D_s$.

7) Outputs the ciphertext:
$\sigma = \{z, V, B_1, B_2, C, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$.

Phase 2. $\mathscr{A}_2$ performs a polynomially bounded number of queries just like in Phase 1. ($\mathscr{A}_2$ should not have queried the secret value for ID_r and requested the plaintext corresponding to the ciphertext σ^*).

Response. $\mathscr{A}_2$ outputs $\mu' \in \{0,1\}$. If $\mu' \doteq \mu$, $\mathscr{C}$ outputs 1. Otherwise, $\mathscr{C}$ outputs 0. If $Y = abQ$, σ^* is a valid ciphertext. Then $\mathscr{A}_2$ distinguishes μ with the advantage ε. So

$$\Pr[\mathscr{C} \longrightarrow 1|Y = abQ] = \Pr[\mu' \doteq \mu|Y = abQ] = \frac{1}{2}+\varepsilon.$$

If $Y \neq abQ$, when $\mu = 0$ or $\mu = 1$, each part of the ciphertext has the same probability distribution, so $\mathscr{A}_2$ has no advantage to distinguishing μ. So

$$\Pr[\mathscr{C} \longrightarrow 1|Y \neq abQ] = \Pr[\mu' \doteq \mu|Y \neq abQ] = \frac{1}{2}.$$

Probability. Let $q_{H_i}(i = 1,2,3,4)$, q_U, q_R, q_D and q_S be the number of $H_i(i = 1,2,3,4)$ queries, user public key queries, user public key replacement requests, partial private key queries and signcryption queries, respectively.

Without loss of generality, we may assume that $L_E \cap L_R = \emptyset$, and denote some events as follows: π_1: $\mathscr{C}$ does not fail in secret value queries; π_2: $\mathscr{C}$ does not fail in unsigncryption queries; π_3: $\mathscr{C}$ does not fail in challenge stage. It is easy to get following results:

$\Pr[\pi_1]=1 - \frac{q_T}{q_Q}$, $\Pr[\pi_2]=1 - \frac{q_U}{2^\nu}$, $\Pr[\pi_3]=\frac{1}{q_Q - q_T}$.

$$\begin{aligned}\Pr[\mathscr{C}\ success] &= \Pr[\pi_1 \wedge \pi_2 \wedge \pi_3] \\ &= \Pr[\pi_1]\cdot\Pr[\pi_2]\cdot\Pr[\pi_3] \\ &= (1 - \frac{q_T}{q_Q})\cdot(1 - \frac{q_U}{2^\nu})\cdot\frac{1}{q_Q - q_T} \\ &\approx \frac{1}{q_Q}.\end{aligned}$$

Therefore, if $\mathscr{A}_2$ can succeed with the probability ε, then $\mathscr{C}$ can solve the DDHP with probability $\frac{\varepsilon}{q_Q}$. □

Theorem 3. *In random oracle model, the scheme is unforgeable against EUF-CLRSC-CMA2 adversary $\mathscr{A}_1$ if the CDHP is hard.*

Proof. Assume that the challenger $\mathscr{C}$ receives an instance (P, aP, bP) of the CDHP. The goal of $\mathscr{C}$ is to compute the value of abP. $\mathscr{C}$ will run $\mathscr{A}_1$ as a subroutine and play the role of challenger in Game III.

Initialization, Phase 1. Same as that in the Theorem 1.

Forge. $\mathscr{A}_1$ outputs a forged signcryption $\sigma = \{z, V, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$ on message m^* under the set $R = L\bigcup\{P_i : P_i \in L\}$, and fulfills the requirements as defined in Game III.

Solve CDHP. Using the forking lemma for ring signature schemes [6], after replays $\mathscr{A}_1$ with the same random tape except the λ^{th} result returned by H_2 query of the forged message, $\mathscr{C}$ gets two valid ring signncryptions with probability $\frac{\varepsilon^2}{66C_{q_{H_2}}^n}$: $\{z, V, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$ and $\{z, V', \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$. It follows that $h_\lambda \neq h'_\lambda$ and $h_i = h'_i$ for $i \neq \lambda$. If ID^* is the actual signer and $\lambda = s$, then $V = (r_1 + h_s)abP$ and $V' = (r_1 + h'_s)abP$, $\mathscr{C}$ solves CDHP by computing: $abP = (h'_s - h_s)^{-1}(V' - V)$.

Probability. Let $q_{H_i}(i = 1,2,3,4)$, q_U, q_D and q_S be the number of $H_i(i = 1,2,3)$ queries, user public key queries, partial private key queries and signcryption queries, respectively.

We denote some events as follows: π_1: $\mathscr{C}$ does not fail during the queries; π_2: $ID^* \in L$; π_3: ID^* is the actual signer; π_4: $\lambda = s$. It is easy to get following results:

$$\begin{aligned}\Pr[\pi_1] &= \frac{q_{H_1} - q_D}{q_{H_1}}, \\ Pr[\pi_2|\pi_1] &= \frac{n}{q_{H_1} - q_D}, \\ Pr[\pi_3|\pi_1 \wedge \pi_2] &= \frac{1}{n}, \\ Pr[\pi_4|\pi_1 \wedge \pi_2 \wedge \pi_3] &= \frac{1}{n}.\end{aligned}$$

$$\begin{aligned}\Pr[\mathscr{C}\ success] &= \Pr[\pi_1 \wedge \pi_2 \wedge \pi_3 \wedge \pi_4] \\ &= \Pr[\pi_1]\cdot Pr[\pi_2|\pi_1]\cdot\Pr[\pi_3|\pi_1 \wedge \pi_2] \\ &\quad \cdot\Pr[\pi_4|\pi_1 \wedge \pi_2 \wedge \pi_3] \\ &= \frac{q_{H_1} - q_D}{q_{H_1}}\cdot\frac{n}{q_{H_1} - q_D}\cdot\frac{1}{n}\cdot\frac{1}{n} \\ &= \frac{1}{n\cdot q_{H_1}}\end{aligned}$$

Therefore, if $\mathscr{A}_1$ can succeed with the probability ε, then $\mathscr{C}$ can solve CDHP with the probability $\frac{\varepsilon^2}{66C_{q_{H_3}}^n}\cdot\frac{1}{n\cdot q_{H_1}}$. □

Theorem 4. *In random oracle model, the scheme is unforgeable against the Type II adversary if the DLP is hard.*

Proof. Assume that the challenger $\mathscr{C}$ receives an instance (P, aP) of the DLP and the goal of $\mathscr{C}$ is to compute the value of a. $\mathscr{C}$ will run $\mathscr{A}_2$ as a subroutine and play the role of challenger in the Game IV.

Initialization, Phase 1. Same as that in the Theorem 2.

Forge. $\mathscr{A}_2$ outputs a forged signcryption $\sigma = \{z, V, \bigcup_{i=1}^{n}\{A_i\}, \bigcup_{i=1}^{n}\{c_i\}\}$ on message m^* under the set $R = L\bigcup\{P_i : P_i \in L\}$, and fulfills the requirements as defined in Game IV.

Solve DLP. Using the forking lemma for ring signature schemes [6], after replays $\mathscr{A}_2$ with the same random tape except the result returned by H_3 query of the forged message, $\mathscr{C}$ gets two valid ring signcryptions with probability $\frac{\varepsilon^2}{66C^n_{q_{H_3}}}$: $\{z, V, \bigcup_{i=1}^n \{A_i\}, \bigcup_{i=1}^n \{c_i\}\}$ and $\{z', V, \bigcup_{i=1}^n \{A_i\}, \bigcup_{i=1}^n \{c'_i\}\}$. It follows that $c_s \neq c'_s$, $c_i = c'_i$ for $i \neq s$. If ID^* is the actual signer, then $z = r_2 - c_s a \pmod q$ and $z' = r_2 - c'_s a \pmod q$, $\mathscr{C}$ solves DLP by computing: $a = (c'_s - c_s)^{-1}(z - z') \bmod q$.

Probability. Let $q_{H_i}(i = 1, 2, 3, 4)$, q_U, q_R, q_D and q_S be the number of $H_i(i = 1, 2, 3)$ queries, user public key queries, user public key replacement requests, partial private key queries and signcryption queries, respectively.

Without loss of generality, we may assume that $L_E \cap L_R = \emptyset$, and denote some events as follows: π_1: $\mathscr{C}$ does not fail during the queries; π_2: $ID^* \in L$; π_3: ID^* is the actual signer. It is easy to get following results:

$\Pr[\pi_1] = \frac{q_U - q_E}{q_U}$, $\Pr[\pi_2|\pi_1] = \frac{n}{q_U - q_E - q_R}$, $\Pr[\pi_3|\pi_1 \wedge \pi_2] = \frac{1}{n}$.

$$\begin{aligned} \Pr[\mathscr{C}\ success] &= \Pr[\pi_1 \wedge \pi_2 \wedge \pi_3] \\ &= \Pr[\pi_1] \cdot \Pr[\pi_2|\pi_1] \cdot \Pr[\pi_3|\pi_1 \wedge \pi_2] \\ &= \frac{q_U - q_E}{q_U} \cdot \frac{n}{q_U - q_E - q_R} \cdot \frac{1}{n} \\ &\geq \frac{1}{q_U} \end{aligned}$$

Therefore, if $\mathscr{A}_2$ can succeed with the probability ε, then $\mathscr{C}$ can solve the DLP with probability $\frac{\varepsilon^2}{66C^n_{q_{H_3}}} \cdot \frac{1}{q_U}$. □

Theorem 5. *The scheme is anonymous.*

Proof. In the scheme, because A_i, c_i are randomly selected from G_1 and Z_q^* for $i \neq s$, respectively. h_i are hash functions values for $i \neq s$, and δ_1 is randomly selected from Z_q^*, so $A_s = \delta_1 E_s - \sum_{i=1, i\neq s}^n (A_i + h_i E_i)$ is distributed uniformly. Since δ_2 is chosen uniformly at random from Z_q^* and y is the output of the random oracle, then $c_s = y - \sum_{i=1, i\neq s}^n c_i \pmod q$ is distributed uniformly. By h_s is the output of the random oracle, then h_s is distributed uniformly. Further, z and V are also distributed uniformly over Z_q^* and G_1, respectively. □

In conclusion, no matter who is the actual signer, all the mentioned parameters are independent and uniformly distributed for any message m, receiver and the user ring L. Therefore, even an adversary with all the private keys corresponding to the set of identities L and unbounded computing resources has no advantage in identifying the actual signer over random guessing.

5 Efficiency and Comparison

By using a famous encryption library (MIRACL) on a mobile device (Samsung Galaxy S5 with a Quad-core 2.45G processor, 2G bytes memory and the Google Android 4.4.2 operating system), He *et al.* [12] obtained the running time for cryptographic operations. The running time are listed in Table 1.

For the CLRSC scheme based on bilinear pairing, we use the Tate bilinear pairing $G_1 \times G_1 \longrightarrow G_2$, where G_1 with prime order $\hat{q}$ is an additive group defined on a super sigular elliptic curve $E/E_p : y^2 = x^3 + x$ over the finite field $F_{\hat{p}}$, and $\hat{p}$ and $\hat{q}$ are 512 bits and 160 bits, respectively. To achieve the same level of security, for the CLRSC based on the non-singular elliptic curve cryptography, we use an additive group G_3 with the prime order $\hat{q}$, which is defined on a non-sigular elliptic curve over the finite field $F_{\hat{p}}$, where both $\hat{p}$ and $\hat{q}$ are 160 bits. We define some notations as follows:

- P: a pairing operation.
- M_{G_1}: a scalar multiplication operation in G_1.
- M_{G_3}: a scalar multiplication operation in G_3.
- E_{G_2}: a exponentiation operation in G_2.
- n: the number of members in the ring.

We use a simple method to evaluate the computation efficiency of different schemes. For example, the scheme [30] needs $3n + 5$ pairing operations, $3n + 2$ scalar multiplication operation in G_1. Therefore, the resulting operation time is $(3n + 5) \times 32.713 + (3n + 2) \times 13.405 = 190.375 + 138.354n$. We now let n = 10, and then the computation time is $190.375 + 138.354 \times 10 = 1573.915$.

According to the above ways, the detailed comparisom results of other schemes [22, 34] are shown in Table 2.

Table 1: Cryptographic operation time (in milliseconds)

P	M_{G_1}	M_{G_3}	E_{G_2}
32.713	13.405	3.335	2.249

6 Conclusion

In recent years, some good results have been achieved in speeding up the computation of pairing function. However, the pairing operation is still relatively expensive. So it is still quite significant to design CLRSC scheme with less pairing operations. In this paper, we constructe a new CLRSC scheme and prove the security against the Type I/II adversary in the random oracle model.

Our proposed scheme is proved to be indistinguishable against adaptive chosen ciphertext attacks, existentially unforgeable against adaptive chosen message attacks and anonymous. The proposed scheme based on certificateless cryptography, it avoids the storage problem of public

Table 2: Comparison of several CLRSC schemes

Scheme	Signcryption	Unsigncryption	Time(n = 10)
Qi [22]	$P + (2n+3)M_{G_1} + E_{G_2}$	$3P + (n+1)M_{G_1}$	588.871
Wang [30]	$(n+2)P + (2n+2)M_{G_1}$	$(2n+3)P + nM_{G_1}$	1573.915
Zhu [34]	$3nP + (n+4)M_{G_1} + nE_{G_2}$	$(2n+1)P + M_{G_1} + nE_{G_2}$	1914.418
Our scheme	$P + (n+3)M_{G_1} + (n+2)M_{G_3}$	$3P + nM_{G_1} + (n+2)M_{G_3}$	519.207

key certificate of public key infrastructure and the key escrow problem in identity based system. Our scheme only requires four pairing operations. Compared with other schemes [22,30,34], our CLRSC scheme is more efficient in computation. Because of the good nature of our scheme, it should be useful for practical application in the ring signcryption.

Acknowledgments

The authors are grateful to the anonymous referees for their helpful comments and suggestions. The research is supported by the National Natural Science Foundation of China under Grants 61562012, the Innovation Group Major Research Projects of Department of Education of Guizhou Province under Grant No. KY[2016]026.

References

[1] R. S. Abdeldaym, H. M. A. Elkader, R. Hussein, "Modified RSA algorithm using two public key and Chinese remainder theorem," *International Journal of Electronics and Information Engineering*, vol. 10, no. 1, pp. 51-64, 2019.

[2] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *9th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'03)*, LNCS 2894, pp. 452-473, 2003.

[3] C. C. Chang, M. S. Hwang, "Parallel computation of the generating keys for RSA cryptosystems", *Electronics Letters*, vo1. 32, no. 15, pp. 1365–1366, 1996.

[4] S. Y. Chen, P. Zeng, K. K. R. Choo and X. L. Dong, "Efficient ring signature and group signature schemes based on q-ary identification protocols," *The Computer Journal*, vol. 61, no. 4, pp. 545-560, 2018.

[5] L. Z. Deng, S. W. Li and Y. F. Yu, " Identity-based threshold ring signcryption from pairing," *International Journal of Electronic Security and Digital Forensics*, vol. 6, no. 2, pp. 333-342, 2014.

[6] L. Z. Deng, C. Liu and X. Wang, " An improved identity-based ring signcryption scheme," *Information Security Journal*, vol. 22, no. 1, pp. 46-54, 2013.

[7] L. Z. Deng, "Certificateless ring signature scheme based on RSA problem and DL problem," *RAIRO-Theoretical Informatics and Applications*, vol. 49, no. 4, pp. 307-318, 2015.

[8] M. Dissanayake, "A new modular multiplication method and its application in RSA cryptosystem," *International Journal of Electronics and Information Engineering*, vol. 10, no. 1, pp. 24-33, 2019.

[9] T. Feng, N. N. Liu, "A sensitive information protection scheme in named data networking using attribute-based ring signcryption," in *IEEE Second International Conference on Data Science in Cyberspace (DSC'17)*, pp. 187-194, 2017.

[10] C. Gritti, W. Susilo and T. Plantard, "Logarithmic size ring signatures without random oracles," *IET Information Security*, vol. 10, no. 1, pp. 1-7, 2016.

[11] Z. Guo, "Cryptanalysis of a certificateless conditional privacy-preserving authentication scheme for wireless body area networks," *International Journal of Electronics and Information Engineering*, vol. 11, no. 1, pp. 1-8, 2019.

[12] D. B. He, H. Wang, L. Wang, J. Shen and X. Yang, "Efficient certificateless anonymous multi-receiver encryption scheme for mobile devices," *Soft Computing*, vol. 21, no. 22, pp. 6801-6810, 2017.

[13] X. Y. Huang, W. Susilo, Y. Mu and F. T. Zhang, "Identity-based ring signcryption schemes: cryptographic primitives for preserving privacy and authenticity in the ubiquitous world," in *19th International Conference on Advanced Information Networking and Applications (AINA'05)*, IEEE, vol. 2 pp. 649-654, 2005.

[14] M. S. Hwang, J. W. Lo, S. C. Lin, "An efficient user identification scheme based on ID-based cryptosystem", *Computer Standards & Interfaces*, vol. 26, no. 6, pp. 565–569, 2004.

[15] A. A. Jothi and D. B. Srinivasan, "Security analysis in boby area networks using attribuute-based ring signcryption scheme,"*Research Journal of Applied Sciences, Engineering and Technology*, vol. 13, no. 1, pp. 48-56, 2016.

[16] A. V. N. Krishna, A. H. Nareyana, K. M. Vani, "Window method based cubic spline curve public key cryptography," *International Journal of Electronics and Information Engineering*, vol. 4, no. 2, pp. 94-102,

2016.

[17] T. C. Lin, T. Y. Yeh, M. S. Hwang, "Cryptanalysis of an ID-based deniable threshold ring authentication", *International Journal of Network Security*, vol. 21, no. 2, pp. 298–302, 2019.

[18] L. H. Liu, Z. Z. Guo, Z. J. Cao and Z. Chen, "An improvement of one anonymous identity-based encryption scheme," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 11-21, 2018.

[19] L. H. Liu, Z. Z. Guo, Z. J. Cao and Z. Chen, "Anonymity and certificateless property could not be acquired concurrently," *International Journal of Electronics and Information Engineering*, vol. 7, no. 2, pp. 61-67, 2017.

[20] L. H. Liu, W. P. Kong, Z. J. Cao and J. B. Wang, "Analysis of one certificateless encryption for secure data sharing in public clouds," *International Journal of Electronics and Information Engineering*, vol. 6, no. 2, pp. 110-115, 2017.

[21] M. J. Qin, Y. L. Zhao and Z. J. Ma, "Pratical constant-size ring signature," *Journal of Computer Science and Technology*, vol. 33, no. 3, pp. 533-541, 2018.

[22] Z. H. Qi, G. Yang and X. Y. Ren, "Provably secure certificateless ring signcryption scheme," *China Communications*, vol. 8, no. 3, pp. 99-106, 2011.

[23] R. L. Rivest, A. Shamir and Y. Tauman, "How to leak a secret," in *International Conference on the Theory and Application of Cryptology and Information Security*, LNCS 2248, pp. 552-565, 2001.

[24] J. L. Salazar, J. L. Tornos and J. J. Piles, "Efficient ways of prime number generation for ring signatures," in *IET Information Security*, vol. 10, no. 1, pp. 33-36, 2016.

[25] S. Shan, "An efficient certificateless signcryption scheme without random oracles," *International Journal of Electronics and Information Engineering*, vol. 11, no. 1, pp. 9-15, 2019.

[26] S. S. D. Selvi, S.S. Vivek and C.P. Rangan, "On the security of identity based ring signcryption schemes," in *proceedings of ISC*, pp. 310-325, 2009.

[27] A. Shamir, "Identity-based cryptosystem and signature scheme," in *Advances in Cryptology (Crypto'84)*, LNCS 196, pp. 47-53, 1984.

[28] G. Sharma, S. Bala and A. K. Verma, "Pairing-free certificateless ring signcryption (PF-CLRSC) scheme for wireless sensor networks," *Wireless Personal Communications*, vol. 84, no. 2, pp. 1469-1485, 2015.

[29] H. Shen, J. Chen, D. He and J. Shen, "Insecurity of a pairing-free certificateless ring signcryption scheme," *Wireless Personal Communications*, vol. 96, no. 4, pp. 5635-5641, 2017.

[30] L. L. Wang, G. Y. Zhang and C. G. Ma, "A secure ring signcryption scheme for private and anonymous communication," in *IFIP International Conference on Network and Parallel Computing Workshops*, pp. 107-111, 2007.

[31] H. Xiong, J. Geng, Z. G. Qin and G. B. Zhu, "Cryptanalysis of attribute-based ring signcryption scheme," *International Journal of Network Security*, vol. 17, no. 2, pp. 224-228, 2015.

[32] C. X. Zhou, Z. M. Cui and G. Y. Gao, " Efficient identity-based generalized ring signcryption scheme," *Ksii Transactions on Internet and Information Systems*, vol. 10, no. 12, pp. 6116-6134, 2016.

[33] Z. C. Zhu, Y. Zhang and F. J. Wang, " An efficient and provable secure identity-based ring signcryption scheme," *Computer Standard and Interfaces*, vol. 31, no. 6, pp. 1092-1097, 2009.

[34] L. J. Zhu, F. T. Zhang and S. Q. Miao, " A provably secure parallel certificatelesss ring signcryption scheme," in *International Conference on Multimedia Information Networking and Security (MINES'10)*, pp. 423-427, 2010.

Biography

Hui Guo received her B.S. from Guizhou Normal University, Guiyang, PR China, in 2016; She is now a master student in the School of Mathematical Sciences, Guizhou Normal University, Guiyang, PR China. Her recent interest include cryptography and information safety.

Lunzhi Deng received his B.S. from Guizhou Normal University, Guiyang, PR China, in 2002; M.S. from Guizhou Normal University, Guiyang, PR China, in 2008; and Ph.D. from Xiamen, PR China, in 2012. He is now a professor in the School of Mathematical Sciences, Guizhou Normal University, Guiyang, PR China. His recent interest include algebra and information safety.