Hindawi
Mathematical Problems in Engineering
Volume 2017, Article ID 7696858, 9 pages
https://doi.org/10.1155/2017/7696858

Research Article

A Certificateless Ring Signature Scheme with High Efficiency in the Random Oracle Model

Yingying Zhang,[1] Jiwen Zeng,[1,2] Wei Li,[1] and Huilin Zhu[1]

[1] *School of Mathematical Sciences, Xiamen University, Fujian 361005, China*
[2] *School of Mathematical Science, Xinjiang Normal University, Xinjiang, China*

Correspondence should be addressed to Jiwen Zeng; jwzeng@xmu.edu.cn

Received 14 December 2016; Accepted 27 April 2017; Published 21 June 2017

Academic Editor: Haipeng Peng

Ring signature is a kind of digital signature which can protect the identity of the signer. Certificateless public key cryptography not only overcomes key escrow problem but also does not lose some advantages of identity-based cryptography. Certificateless ring signature integrates ring signature with certificateless public key cryptography. In this paper, we propose an efficient certificateless ring signature; it has only three bilinear pairing operations in the verify algorithm. The scheme is proved to be unforgeable in the random oracle model.

1. Introduction

In the traditional cryptography, the communicating parties distribute a private key by sending the key in advance over some secure channels. But there is a major barrier that the key distribution will cost and delay large teleprocessing networks. In 1976, Diffie and Hellman [1] first introduced the concept of public key cryptography (PKC) and proposed some techniques to solve this longstanding problem in traditional cryptography. But the traditional public key infrastructure confronted with the problem of certificate management. In order to solve this problem, Shamir [2] proposed an identity-based cryptography scheme based on public key cryptography (ID-PKC) in 1995. In his scheme, every user chooses his fundamental information as his public key and the user's private key is generated directly by a private key generation (PKG) referred as master key. But there is a problem that the third party PKG has the private keys of all users and must be fully trusted; we call it the key escrow problem.

In 2003, Al-Riyami and Paterson [3] introduced the concept of certificateless public key cryptography (CL-PKC). CL-PKC not only overcomes key escrow problem but also does not lose some advantages of ID-PKC. Key generation cryptography (KGC) in CL-PKC only issues the partial private key to a user. Then, the user combines the private key from KGC with a self-generated secret key to generate his actual private key, so that the KGC does not access user's private key fully like in ID-PKC. Moreover, the public key of a user is generated by user himself by computing the KGC's public parameters and the secret values of the user. Over last years, the certificateless signature (CLS) has been investigated successfully and attracted great attention [4–8].

In 2001, Rivest et al. [9] first proposed the concept of ring signature (RS). Ring signature is designed for the situation that a member in a group wants to sign messages on behalf of the group while keeping his identity anonymous. Therefore, ring signature can protect the identity of the signer. In a ring signature, the signer forms a group (called a ring) only by collecting the public keys of all the group members including himself to keep the singer's identity anonymous. In addition, ring signature is characterized with spontaneity; it means that the singer can generate a valid signature without help of any other members of the ring. Due to above two characteristics of ring signature, it is now widely used in electronic voting.

A ring signature should meet the following three properties:

(i) *Verifiability*. The verifier can be convinced of the signer's agreement on the signed message.

(ii) *Unforgeability*. No one, even any member of the ring, can forge other ring members to generate a valid ring signature.

(iii) *Unconditional Anonymity*. No one can determine the identity of the signer through the final ring signature.

After ring signature given by Rivest et al. [9], many researchers have been proposing ring signature schemes and their variants such as threshold ring signatures [10–12] and constant-size ring signatures [13–16]. Ring signature schemes based on standard assumptions without random oracles were proposed in [17–20].

As we know ring signature has been studied greatly in traditional PKC [18, 21, 22] and ID-PKC [17, 23–27]. But the applications of ring signature in traditional PKC and ID-PKC are restricted since there are some flaws in them. In fact, in a ring signature based on PKC, the verifier must check the validity of certificates of some group members, which will make the signature scheme inefficient since the computational cost will increase with the group size. Moreover, the ring signature based on ID-PKC has the key escrow problem. As described before, certificateless cryptography can make up the drawbacks in traditional PKC and ID-PKC. Therefore, several certificateless ring signatures (CLRS) integrating ring signature with certificateless cryptography have been proposed [28–30].

Over the last few decades, certificateless signature and ring signature have been studied extensively; however there is little work on certificateless ring signatures [28, 31–33]. Chow and Yap [32] presented a CLRS scheme based on a security model they proposed, but their scheme requires $n+1$ pairing operations and 2 exponentiation operations. Later, a CLRS scheme only requiring 5 pairing operations and $4n+1$ exponentiation operations was proposed by Zhang et al. (see [33]). Two years later Chang et al. [31] constructed a concrete CLRS scheme, which reduces the pairing operations to 4 while it needs $4n+2$ exponentiation operations.

We know that it is always interesting to design a cryptographic scheme with less pairing operations to speed up the computation of pairing function in recent years. To the best of our knowledge, the most efficient certificateless ring signature scheme based on bilinear pairings requires at least four bilinear maps. In this paper, we will propose a certificateless ring signature. Our scheme only needs 3 bilinear maps in the verification phase. By the analysis in Section 6, we know that our scheme is more efficient compared with other certificateless ring signature schemes [31–33].

The rest of the paper is organized as follows. Section 2 presents the basic concepts of bilinear pairings and some related mathematical problems. Section 3 presents a formal definition and security model of a certificateless ring signature scheme. Section 4 presents our certificateless ring signature scheme. We prove its security in Section 5. Schemes comparison will be given in Section 6. Finally, we give some conclusions in Section 7.

2. Preliminaries

2.1. Bilinear Pairing. Let $\mathbb{G}_1$ be a cyclic additive group of prime order and $\mathbb{G}_2$ be a cyclic multiplicative group of the same order.

We call e a bilinear pairing if $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ is a map with the following three properties:

(1) Bilinearity: $e(aP, bQ) = e(P,Q)^{ab}$, and $P,Q \in \mathbb{G}_1$.

(2) Nondegeneracy: there exist $P,Q \in \mathbb{G}_1$ such that $e(P,Q) \neq 1$.

(3) Computability: there is an efficient algorithm to compute $e(P,Q)$ for any two random elements $P,Q \in \mathbb{G}_1$.

Security of the proposed scheme relies on the following questions and assumptions.

Definition 1 (computational Diffie-Hellman (CDH) problem). Let $\mathcal{G} = (E,+)$, where E is an elliptic curve over a finite field $\mathbb{F}_q$ and $P \in E$ is a point having prime order $d = |E|/2$. Let $\mathbb{G}_1 = \langle P\rangle \leq \mathcal{G}$, the computational Diffie-Hellman (CDH) Problem is that given two random elements $aP, bP \in \mathbb{G}_1$ for unknown $a, b \in \mathbb{Z}_d^*$, to compute abP.

Definition 2 (computational Diffie-Hellman (CDH) assumption). Let $\mathcal{G}$ be a CDH parameter generator. We say that an algorithm $\mathcal{A}$ has advantage $\varepsilon(k)$ in solving the CDH problem for $\mathcal{G}$ if, for a sufficiently large k,

$$\begin{aligned} \mathrm{Adv}_{\mathcal{G},\mathcal{A}}(k) = \Pr\big[\mathcal{A}(q, \mathbb{G}_1, aP, bP) = abP \mid (q, \mathbb{G}_1) \\ \longleftarrow \mathcal{G}(1^k), P \longleftarrow \mathbb{G}_1, a, b \longleftarrow \mathbb{Z}_d^*\big] \geq \varepsilon(k). \end{aligned} \tag{1}$$

Given an upper limitation time T, we say that $\mathcal{G}$ satisfies the CDH assumption if for any randomized polynomial-time algorithm $\mathcal{A}$, we have that $\mathrm{Adv}_{\mathcal{G},\mathcal{A}}(k)$ is a negligible function. When $\mathcal{G}$ satisfies the CDH assumption, we say that the CDH problem is hard in $\mathbb{G}_1$ generated by $\mathcal{G}$.

Definition 3 (computational co-Diffie-Hellman (co-CDH) problem). Let $\mathcal{G} = (E,+)$, where E is an elliptic curve over a finite field $\mathbb{F}_q$ and $P \in E$ is a point having prime order $d = |E|/2$. Let $\mathbb{G}_1 = \langle P\rangle \leq \mathcal{G}$; the Computational co-Diffie-Hellman (co-CDH) Problem is that given two random elements $aP, X \in \mathbb{G}_1$ for unknown $a \in \mathbb{Z}_d^*$, to compute aX.

Definition 4 (computational co-Diffie-Hellman (co-CDH) assumption). Let $\mathcal{G}$ be a co-CDH parameter generator. We say that an algorithm $\mathcal{A}$ has advantage $\varepsilon(k)$ in solving the co-CDH problem for $\mathcal{G}$ if, for a sufficiently large k,

$$\begin{aligned} \mathrm{Adv}_{\mathcal{G},\mathcal{A}}(k) = \Pr\big[\mathcal{A}(q, \mathbb{G}_1, aP, X) = aX \mid (q, \mathbb{G}_1) \\ \longleftarrow \mathcal{G}(1^k), P, X \longleftarrow \mathbb{G}_1, a \longleftarrow \mathbb{Z}_d^*\big] \geq \varepsilon(k). \end{aligned} \tag{2}$$

Given an upper limitation time T, we say that $\mathcal{G}$ satisfies the (co-CDH) assumption if for any randomized polynomial-time algorithm $\mathcal{A}$, we have that $\mathrm{Adv}_{\mathcal{G},\mathcal{A}}(k)$ is a negligible function. When $\mathcal{G}$ satisfies the (co-CDH) assumption, we say that the (co-CDH) problem is hard in $\mathbb{G}_1$ generated by $\mathcal{G}$.

3. Formal Definition and Security Model

3.1. Formal Definition of a Certificateless Ring Signature Scheme. A certificateless ring signature scheme (CLRS) can be specified by seven algorithms: Setup, Partial Private Key

Extract, Set Secret Value, Set Private Key, Set Public Key, CLRS Generation, and CLRS Verification. Every algorithm is depicted as follows.

(i) *Setup*. Given a security parameter, it outputs a list of system parameters.

(ii) *Partial Private Key Extract*. On input a master key, a user's identity ID_i, and system parameters, it generates the user's partial private key S_{ID_i}.

(iii) *Set Secret Value*. Given a user's identity ID_i, it outputs the user's secret value $x_i \in \mathbb{Z}_d^*$ and computes $Y_i = x_i P$.

(iv) *Set Private Key*. The user takes the pair (S_{ID_i}, x_i) as its private key.

(v) *Set Public Key*. The user with identity ID_i constructs his public key pair (Y_i, Q_{ID_i}) responding to x_i and S_{ID_i}, respectively.

(vi) *CLRS Generation*. Given a message m, signer chooses $n-1$ other users to form a ring $\mathcal{U}$; then it outputs a ring signature σ on behalf of the ring $\mathcal{U}$.

(vii) *CLRS Verification*. Given a message m, a ring signature σ, and the public keys $Y_1, \ldots, Y_n$ of the n signers, it outputs "accept" if σ is a valid ring signature and "reject" otherwise.

3.2. Security Model of Certificateless Ring Signature Scheme. In our certificateless ring signature scheme, we consider the following two attackers.

Type I Adversary. Adversary $\mathcal{A}_{\mathrm{I}}$ does not have access to the master key, but $\mathcal{A}_{\mathrm{I}}$ can replace the public keys of any entity with a value of his choice, because there is no certificate involved in CLRS.

Type II Adversary. This type of adversary $\mathcal{A}_{\mathrm{II}}$ is a malicious KGC. Adversary $\mathcal{A}_{\mathrm{II}}$ is allowed to have access to the master key but does not replace any user's public key. A type II adversary should also be allowed to change a user's partial private key.

Game 1 for Type I Adversary. Type I adversary advantage $\mathrm{Adv}_{\mathrm{CLRS},\mathcal{A}_{\mathrm{I}}}$ is defined as its probability of success in the following game between a challenger $\mathcal{C}$ and a type I adversary $\mathcal{A}_{\mathrm{I}}$.

(i) *Setup*. Given a security parameter, challenger $\mathcal{C}$ runs the setup algorithm to obtain a list of system parameters. And challenger $\mathcal{C}$ sends system parameters to type I adversary $\mathcal{A}_{\mathrm{I}}$.

(ii) *Hash Queries*. $\mathcal{A}_{\mathrm{I}}$ submits any value he chooses, and challenger $\mathcal{C}$ returns the corresponding hash value to him.

(iii) *User Public Key Queries*. $\mathcal{A}_{\mathrm{I}}$ requests any public key of a user ID_i whom he chooses, and challenger $\mathcal{C}$ returns the corresponding public key Y_i to him.

(iv) *Partial Private Key Queries*. $\mathcal{A}_{\mathrm{I}}$ requests any partial private key of a user ID_i whom he chooses, and challenger $\mathcal{C}$ returns the corresponding partial private key S_{ID_i} to him.

(v) *User Public Key Replacements*. $\mathcal{A}_{\mathrm{I}}$ submits a new public key value Y_i' with respect to a user ID_i. Challenger $\mathcal{C}$ replaces the current public key with the value Y_i'.

(vi) *Secret Value Queries*. $\mathcal{A}_{\mathrm{I}}$ requests any secret value of a user ID_i whose public key was not replaced, and challenger $\mathcal{C}$ returns the corresponding secret value x_i to $\mathcal{A}_{\mathrm{I}}$. If a user's public key was replaced, $\mathcal{A}_{\mathrm{I}}$ cannot query the corresponding secret value.

(vii) *Ring Signature Queries*. $\mathcal{A}_{\mathrm{I}}$ submits any message he chooses, and challenger $\mathcal{C}$ returns a ring signature σ to him.

(viii) *Forge*. Eventually, $\mathcal{A}_{\mathrm{I}}$ outputs a certificateless ring signature σ^* on a message m^* such that

(1) σ^* is a valid certificateless ring signature;

(2) $\mathcal{A}_{\mathrm{I}}$ can not query the partial private key of anyone in $\mathcal{U}$;

(3) m^* has never been submitted to the ring signature queries.

Definition 5. A forger $\mathcal{A}_{\mathrm{I}}(t, q_{H_1}, q_{H_2}, q_D, q_U, q_{\mathrm{RS}}, \varepsilon)$ breaks a certificateless ring signature scheme (CLRS) meaning that if $\mathcal{A}_{\mathrm{I}}$ runs in time at most t, $\mathcal{A}_{\mathrm{I}}$ makes at most q_{H_1} H_1 Hash queries, at most q_{H_2} H_2 Hash queries, at most q_D partial private key queries, at most q_U user public key queries, and q_{RS} ring signature queries; then $\mathrm{Adv}_{\mathrm{CLRS},\mathcal{A}_{\mathrm{I}}}$ is at least ε. A certificateless ring signature scheme is $(t, q_{H_1}, q_{H_2}, q_D, q_U, q_{\mathrm{RS}}, \varepsilon)$-existentially unforgeable under an adaptively chosen-message attack if no forger $(t, q_{H_1}, q_{H_2}, q_D, q_U, q_{\mathrm{RS}}, \varepsilon)$ breaks it.

Game 2 for Type II Adversary. Type II adversary advantage $\mathrm{Adv}_{\mathrm{CLRS},\mathcal{A}_{\mathrm{II}}}$ is defined as its probability of success in the following game between a challenger $\mathcal{C}$ and a type II adversary $\mathcal{A}_{\mathrm{II}}$.

(i) *Setup*. Given a security parameter, challenger $\mathcal{C}$ runs the setup algorithm to obtain a list of system parameters. And challenger $\mathcal{C}$ sends system parameters and the master key θ to type II adversary $\mathcal{A}_{\mathrm{II}}$.

(ii) *Hash Queries*. $\mathcal{A}_{\mathrm{II}}$ submits any value he chooses, and challenger $\mathcal{C}$ returns the corresponding hash value to him.

(iii) *User Public Key Queries*. $\mathcal{A}_{\mathrm{II}}$ requests any public key of a user ID_i whom he chooses, and challenger $\mathcal{C}$ returns the corresponding public key Y_i to him.

(iv) *Partial Private Key Queries*. Because $\mathcal{A}_{\mathrm{II}}$ has the system master key θ, so $\mathcal{A}_{\mathrm{II}}$ can compute the partial private key of any user by himself.

(v) *User Public Key Replacements*. $\mathcal{A}_{\mathrm{II}}$ submits a new public key value Y_i' with respect to a user ID_i. Challenger $\mathcal{C}$ replaces the current public key with the value Y_i'.

(vi) *Secret Value Queries*. $\mathcal{A}_{\mathrm{II}}$ requests any secret value of a user ID_i whose public key was not replaced, and challenger $\mathscr{C}$ returns the corresponding secret value x_i to $\mathscr{A}_{\text{II}}$. If a user's public key was replaced, $\mathscr{A}_{\text{II}}$ cannot query the corresponding secret value.

(vii) *Ring Signature Queries*. $\mathscr{A}_{\text{II}}$ submits any message he chooses, and challenger $\mathscr{C}$ returns a ring signature σ to $\mathscr{A}_{\text{II}}$.

(viii) *Forge*. Eventually, $\mathscr{A}_{\text{II}}$ outputs a certificateless ring signature σ^* on a message m^* such that

(1) σ^* is a valid certificateless ring signature;

(2) $\mathscr{A}_{\text{II}}$ can not query the secret value of anyone in $\mathscr{U}$;

(3) $\mathscr{A}_{\text{II}}$ can not replace the user public key of anyone in $\mathscr{U}$;

(4) m^* has never been submitted to the ring signature queries.

Definition 6. A forger $\mathscr{A}_{\text{II}}(t, q_{H_1}, q_{H_2}, q_E, q_R, q_U, q_{\text{RS}}, \varepsilon)$ breaks a certificateless ring signature scheme (CLRS) means that if $\mathscr{A}_{\text{II}}$ runs in time at most t, $\mathscr{A}_{\text{II}}$ makes at most q_{H_1} H_1 Hash queries, at most q_{H_2} H_2 Hash queries, at most q_E secret value queries, at most q_R user public key replacement queries, at most q_U user public key queries, and q_{RS} ring signature queries; then $\text{Adv}_{\text{CLRS},\mathscr{A}_{\text{II}}}$ is at least ε. A certificateless ring signature scheme is $(t, q_{H_1}, q_{\text{H}_2}, q_E, q_R, q_U, q_{\text{RS}}, \varepsilon)$-existentially unforgeable under an adaptively chosen-message attack if no forger $(t, q_{H_1}, q_{H_2}, q_E, q_R, q_U, q_{\text{RS}}, \varepsilon)$ breaks it.

Game 3 Anonymity of a Certificateless Ring Signature Scheme. Let $\mathscr{U} = (u_1, u_2, \ldots, u_n)$ be n signers and W be the n signers' identities. $\mathscr{A}$ be an adversary and $\mathscr{C}$ be a challenger whom are all involved in the game 3.

(i) The challenger $\mathscr{C}$ runs the setup algorithm to obtain a list of system parameters. And challenger $\mathscr{C}$ sends system parameters to adversary $\mathscr{A}$.

(ii) The adversary $\mathscr{A}$ adaptively make a polynomially bounded number of queries.

(iii) In the challenge phase, the adversary outputs a message m, a group of n users' identities W, and two different members $\text{ID}_0, \text{ID}_1 \in W$ to the challenger $\mathscr{C}$. The challenger $\mathscr{C}$ randomly chooses a bit $\mu \in \{0, 1\}$ and sends $\mathscr{A}$ to a ring signature $\sigma = \text{RS}(m, W, x_\mu)$.

(iv) The adversary $\mathscr{A}$ can make a polynomially bounded number of queries.

(v) Finally, adversary $\mathscr{A}$ outputs a bit $\mu' \in \{0, 1\}$.

The adversary $\mathscr{A}$ wins the above game if and only if $\mu = \mu'$.

Definition 7. Define the probability of success in the game 3 of adversary $\mathscr{A}$ as $\text{succ}(\mathscr{A}) = \Pr[\mu = \mu'] = 1/2 + \varepsilon$. A certificateless ring signature scheme is said to have unconditional anonymity if no adversary has no nonnegligible advantage in winning the above game. That is to say, A certificateless ring signature scheme is said to have unconditional anonymity if $\varepsilon = 0$.

4. Our Scheme

In this section, we propose a certificateless ring signature scheme. Participants in the program include n signers $\mathscr{U} = (u_1, u_2, \ldots, u_n)$ and a verifier V. Our scheme is described as follows:

(i) *Setup*. Given a security parameter z, KGC outputs a large prime d. Let $\mathbb{G}_1$ be a cyclic additive group of prime order d. Let $\mathbb{G}_2$ be a cyclic multiplicative group of the same order. Let P, Q be two generators of $\mathbb{G}_1$. KGC chooses the master private key $\theta \in \mathbb{Z}_d^*$ randomly and computes the master public key $P_{\text{Pub}} = \theta P$. Let $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a bilinear map. Let $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$, $H_2 : \{0,1\}^* \times \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{Z}_d^*$, and $H_3 : \{0,1\}^* \times \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{Z}_d^*$ be three secure cryptographic hash functions. KGC publishes system parameters $(\mathbb{G}_1, \mathbb{G}_2, d, P, Q, P_{\text{Pub}}, e, H_1, H_2, H_3)$ and secretly keeps the master key θ.

(ii) *Partial Private Key Extract*. Given a user's identity ID_i, KGC computes $Q_{\text{ID}_i} = H_1(\text{ID}_i)$ and $S_{\text{ID}_i} = \theta Q_{\text{ID}_i}$. Then KGC sends the user's partial private key S_{ID_i} to him. The user can check its correctness by checking whether $e(S_{\text{ID}_i}, P) = e(Q_{\text{ID}_i}, P_{\text{Pub}})$.

(iii) *Set Secret Value*. User ID_i selects $x_i \in \mathbb{Z}_d^*$ randomly as her secret value. Then User computes the corresponding value $Y_i = x_i P$.

(iv) *Set Private Key*. User ID_i takes the pair $\text{PRK} = (x_i, S_{\text{ID}_i})$ as its private key.

(v) *Set Public Key*. User ID_i takes the pair $\text{PUK} = (Y_i, Q_{\text{ID}_i})$ as its public key.

(vi) *CLRS Generation*. Given a message m, $W = \{\text{ID}_1, \text{ID}_2, \ldots, \text{ID}_n\}$ is a set of n users' identities. An actual signer $u_s \in \mathscr{U}$ can propose a certificateless ring signature σ. The signer u_s operates as follows:

(1) Choose $R_i, K_i \in \mathbb{G}_1$ $(i = 1, 2, \ldots, n \backslash s)$ randomly and compute

$$r_i = H_2\left(m \parallel W, R_i, Y_i\right), \quad i = 1, 2, \ldots, s-1, s+1, \ldots, n,$$
$$k_i = H_3\left(m \parallel r_i \parallel W, K_i, Y_i\right), \quad i = 1, 2, \ldots, s-1, s+1, \ldots, n. \tag{3}$$

(2) Select $r, k \in \mathbb{Z}_q^*$ and compute

$$R_s = rP - \sum_{i \neq s}\left(r_i Q_{\text{ID}_i} + R_i\right),$$
$$K_s = kP - \sum_{i \neq s}\left(k_i Y_i + K_i\right); \tag{4}$$

then compute:

$$r_s = H_2\left(m \parallel W, R_s, Y_s\right),$$
$$k_s = H_3\left(m \parallel r_s \parallel W, K_s, Y_s\right). \tag{5}$$

(3) Compute $V = rP_{\text{Pub}} + r_s S_{\text{ID}_s} + k_s x_s Q + kQ$.

(4) Output $\sigma = (R_1, R_2, \ldots, R_n, K_1, K_2, \ldots, K_n, m, V)$.

(vii) *CLRS Verification*. Given public keys of the n signer, a verifier V can verify a certificateless ring signature σ by checking if the following equation holds:

$$e(P, V) = e\left(\sum_{i=1}^{n} r_i Q_{\text{ID}_i} + R_i, P_{\text{Pub}}\right) e\left(\sum_{i=1}^{n} k_i Y_i + K_i, Q\right). \quad (6)$$

If it holds, the verifier "accepts" the signature and "rejects" otherwise.

5. Security Analysis

In this section, we mainly focus on the unforgeability of the proposed certificateless ring signature scheme. Now, we give the following three theorems.

5.1. Unforgeability against Type I Adversary

Theorem 8. *The scheme is unforgeable against a type I adversary $\mathscr{A}_{\text{I}}$ in the random oracle model if the CDH problem is hard.*

Proof. Suppose challenger $\mathscr{C}$ receives a random instance (P, aP, bP) of the CDH problem and has to compute the value of abP. Challenger $\mathscr{C}$ sets the system public key $P_{\text{Pub}} = aP$. $\mathscr{C}$ will run $\mathscr{A}_{\text{I}}$ as a subroutine and act as $\mathscr{A}_{\text{I}}$'s challenger in game 1. Without loss of generality, we assume that all the queries are distinct. Now, we will show how challenger $\mathscr{C}$ answers a type I adversary $\mathscr{A}_{\text{I}}$'s queries in the following. □

Initialization. At the beginning of the game, challenger $\mathscr{C}$ runs the setup algorithm with the parameter z and then gives adversary $\mathscr{A}_{\text{I}}$ the system parameters: $(\mathbb{G}_1, \mathbb{G}_2, d, P, Q, P_{\text{Pub}}, e, H_1, H_2, H_3)$.

(i) H_1 *Queries*. Challenger $\mathscr{C}$ maintains the list L_1 of tuple $(\text{ID}_i, v_i P)$. The list is initially empty. When adversary $\mathscr{A}_{\text{I}}$ makes a query $H_1(\text{ID}_i)$, challenger $\mathscr{C}$ responds as follows. Challenger $\mathscr{C}$ chooses a random integer f in $[1, q_{H_1}]$ firstly. At the ith H_1 query, if $i \neq f$, challenger $\mathscr{C}$ randomly selects a value $v_i \in \mathbb{Z}_d^*$, and sets $H_1(\text{ID}_i) = v_i P$; otherwise, challenger $\mathscr{C}$ sets $H_1(\text{ID}_*) = bP$.

(ii) H_2 *Queries*. Challenger $\mathscr{C}$ maintains the list L_2 of tuple (β_i, h_2). The list is initially empty. When $\mathscr{A}_{\text{I}}$ makes a query $H_2(\beta_i)$, challenger $\mathscr{C}$ selects a value h_2 randomly, and sets $H_2(\beta_i) = h_2$. Then challenger $\mathscr{C}$ adds (β_i, h_2) to the H_2 list and returns h_2 to $\mathscr{A}_{\text{I}}$.

(iii) H_3 *Queries*. Challenger $\mathscr{C}$ maintains the list L_3 of tuple (γ_i, h_3). The list is initially empty. When $\mathscr{A}_{\text{I}}$ makes a query $H_3(\gamma_i)$, challenger $\mathscr{C}$ selects a value h_3 randomly and sets $H_3(\gamma_i) = h_3$. Then challenger $\mathscr{C}$ adds (γ_i, h_3) to the H_3 list and returns h_3 to $\mathscr{A}_{\text{I}}$.

(iv) *User Public Key Queries*. Challenger $\mathscr{C}$ maintains the list L_U of tuple (ID_i, Y_i, x_i). The list is initially empty. When adversary $\mathscr{A}_{\text{I}}$ makes a user public key query for ID_i, challenger $\mathscr{C}$ selects a value $x_i \in Z_d^*$, and sets $Y_i = x_i P$. Then challenger $\mathscr{C}$ adds (ID_i, Y_i, x_i) to the L_U list and returns Y_i to $\mathscr{A}_{\text{I}}$.

(v) *Partial Private Key Queries*. Challenger $\mathscr{C}$ maintains the list L_S of tuple $(\text{ID}_i, S_{\text{ID}_i})$. The list is initially empty. When adversary $\mathscr{A}_{\text{I}}$ makes a user partial private key query for ID_i, if $\text{ID}_i = \text{ID}_*$, $\mathscr{C}$ fails and stops. Otherwise challenger $\mathscr{C}$ computes $S_{\text{ID}_i} = v_i P_{\text{Pub}}$. Then challenger $\mathscr{C}$ adds $(\text{ID}_i, S_{\text{ID}_i})$ to the L_S list and returns S_{ID_i} to $\mathscr{A}_{\text{I}}$.

(vi) *User Public Key Replacements*. Challenger $\mathscr{C}$ maintains the list L_R of tuple (ID_i, Y_i, Y_i'). The list is initially empty. When $\mathscr{A}_{\text{I}}$ makes a user public key replacement request for u_i with other public value Y_i', $\mathscr{C}$ replaces Y_i with Y_i' and adds (ID_i, Y_i, Y_i') to the L_R list.

(vii) *Secret Value Queries*. Challenger $\mathscr{C}$ maintains the list L_E of tuple (ID_i, x_i). The list is initially empty. When adversary $\mathscr{A}_{\text{I}}$ makes a user secret value query for ID_i, $\mathscr{C}$ checks the lists L_U firstly. If the tuple (ID_i, x_i) is found in the list L_U, $\mathscr{C}$ returns x_i to $\mathscr{A}_{\text{I}}$. Otherwise challenger $\mathscr{C}$ randomly chooses $x_i \in Z_d^*$, returns x_i to $\mathscr{A}_{\text{I}}$, and adds (ID_i, x_i) to the L_E list.

(viii) *Ring Signature Queries*. $\mathscr{A}_{\text{I}}$ submits a message m and a set of n users' identities $W = \{\text{ID}_1, \text{ID}_2, \ldots, \text{ID}_n\}$. $\mathscr{C}$ outputs a ring signature as follows. If there exists a user $\text{ID}_s \in W$ such that $\text{ID}_s \neq \text{ID}_*$ and $\text{ID}_s \notin L_R$, then challenger $\mathscr{C}$ returns the ring signature σ by calling the signing algorithm, where ID_s is the actual signer. Otherwise, challenger $\mathscr{C}$ does as follows:

(1) Selects $R_i, K_i \in G_1$ randomly for all $i \in (1, 2, \ldots, n)$ and $i \neq s$.

(2) For all $i \in (1, 2, \ldots, n)$, selects $r_i, k_i \in \mathbb{Z}_d^*$ randomly.

(3) Chooses two values $r, k \in \mathbb{Z}_d^*$ randomly and computes

$$R_s = rP - \sum_{i=1}^{n} r_i Q_{\text{ID}_i} - \sum_{i \neq s} R_i,$$
$$K_s = kP - \sum_{i=1}^{n} k_i Y_i - \sum_{i \neq s} K_i. \quad (7)$$

(4) Computes $V = rP_{\text{Pub}} + kQ$.

(5) Outputs $\sigma = (R_1, R_2, \ldots, R_n, K_1, K_2, \ldots, K_n, m, V)$.

Forge. Adversary $\mathscr{A}_{\text{I}}$ outputs a ring signature σ^* on a message m^* that fulfills the following conditions:

(1) σ^* is a valid ring signature.

(2) $\mathscr{A}_{\text{I}}$ cannot query the partial private key of anyone in $\mathscr{U}$.

(3) The forged signature σ^* is not from signature query.

Output. It follows from the forking lemma that if $\varepsilon \geq 7C^n_{q_{H_2}}/2^k$, adversary $\mathscr{A}_{\mathrm{I}}$ can give a valid forged signature within time T_A in the above interaction; then we can construct another algorithm $\mathscr{A}'_{\mathrm{I}}$ that outputs two signed messages within time $2T_A$ with probability at least $\varepsilon^2/66C^n_{q_{H_2}}$. For the resemble construction, $\mathscr{C}$ can get two valid ring signature σ and σ' satisfying

$$\begin{aligned}\sigma &= \left(R_1, R_2, \ldots, R_{s-1}, R_s, R_{s+1}, \ldots, R_n, K_1, \ldots, K_n, m, V\right),\\ \sigma' &= \left(R_1, R_2, \ldots, R_{s-1}, R'_s, R_{s+1}, \ldots, R_n, K_1, \ldots, K_n, m, V'\right).\end{aligned} \tag{8}$$

So we have

$$\begin{aligned}V &= rP_{\mathrm{Pub}} + r_s S_{\mathrm{ID}_s} + k_s x_s Q + kQ,\\ V' &= rP_{\mathrm{Pub}} + r'_s S_{\mathrm{ID}_s} + k_s x_s Q + kQ.\end{aligned} \tag{9}$$

Challenger $\mathscr{C}$ outputs

$$abP = S_{\mathrm{ID}_s} = \left(r_s - r'_s\right)^{-1}\left(V - V'\right). \tag{10}$$

Probability. Let q_{H_1}, q_{H_2}, q_{H_3}, q_D, q_U, and q_{RS} be times of H_1 queries, H_2 queries, H_3 queries, partial private key queries, user public key queries, and ring signature queries, respectively. The probability that ID_*'s partial private key was not queried by $\mathscr{A}_{\mathrm{I}}$ during the queries is $(q_{H_1} - q_D)/q_{H_1}$. The probability that ID_* belongs to the groups W is $n/(q_{H_1} - q_D)$. The probability that ID_* is the actual signer is $1/n$. So the combined probability is $(q_{H_1} - q_D)/q_{H_1} \cdot n/(q_{H_1} - q_D) \cdot 1/n = 1/q_{H_1}$.

Therefore, according to the forking lemma, if the attacker $\mathscr{A}_{\mathrm{I}}$ can succeed in making a valid ring signature with a probability ε, the advantage of challenger $\mathscr{C}$ solving an instance of CDH problem in game 1 is at least $\varepsilon^2/66C^n_{q_{H_2}} \cdot 1/q_{H_1}$.

5.2. Unforgeability against Type II Adversary

Theorem 9. *The scheme is unforgeable against a type II adversary $\mathscr{A}_{II}$ in the random oracle model if the co-CDH problem is hard.*

Proof. Suppose challenger $\mathscr{C}$ receives a random instance (aP, X) of the co-CDH and has to compute the value of aX. Challenger $\mathscr{C}$ sets $Q = X$. Challenger $\mathscr{C}$ will run adversary $\mathscr{A}_{\mathrm{II}}$ as a subroutine and act as $\mathscr{A}_{\mathrm{II}}$'s challenger in the game 2. Without loss of generality, we assume that all the queries are distinct. Now, we will show how challenger $\mathscr{C}$ answers type II adversary $\mathscr{A}_{\mathrm{II}}$'s queries in the following.

Initialization. At the beginning of the game, challenger $\mathscr{C}$ runs the setup algorithm with the parameter z and gives adversary $\mathscr{A}_{\mathrm{II}}$ the system parameters: $(\mathbb{G}_1, \mathbb{G}_2, d, P, Q, P_{\mathrm{Pub}}, e, H_1, H_2, H_3)$ and the system master secret key θ.

(i) H_1 *Queries*. Challenger $\mathscr{C}$ maintains the list L_1 of tuple $(\mathrm{ID}_i, v_i P)$. The list is initially empty. When adversary $\mathscr{A}_{\mathrm{II}}$ makes a query $H_1(\mathrm{ID}_i)$, challenger $\mathscr{C}$ selects a value $v_i \in \mathbb{Z}^*_d$ randomly and computes $H_1(\mathrm{ID}_i) = v_i P$. Then challenger $\mathscr{C}$ adds $(\mathrm{ID}_i, v_i P)$ to the H_1 list and returns $v_i P$ to $\mathscr{A}_{\mathrm{II}}$.

(ii) H_2 *Queries*. Same as that in the proof of Theorem 8.

(iii) H_3 *Queries*. Same as that in the proof of Theorem 8.

(iv) *User Public Key Queries*. Challenger $\mathscr{C}$ maintains the list L_U of tuple $(\mathrm{ID}_i, Y_i, x_i)$. The list is initially empty. When adversary $\mathscr{A}_{\mathrm{II}}$ makes a user public key query for ID_i, challenger $\mathscr{C}$ responds as follows. Challenger $\mathscr{C}$ chooses a random integer j in $[1, q_U]$ firstly. At the ith q_U query, if $i \neq j$, challenger $\mathscr{C}$ selects a value $x_i \in \mathbb{Z}^*_d$ randomly and sets $Y_i = x_i P$. Otherwise, challenger $\mathscr{C}$ sets $\mathrm{ID}_j = \mathrm{ID}_*$ and $Y_* = aP$.

(v) *Partial Private Key Queries*. Adversary $\mathscr{A}_{\mathrm{II}}$ can compute the partial private keys of any identities by himself with the master secret key.

(vi) *User Public Key Replacements*. Same as that in the proof of Theorem 8.

(vii) *Secret Value Queries*. Challenger $\mathscr{C}$ maintains the list L_E of tuple (ID_i, x_i). The list is initially empty. When adversary $\mathscr{A}_{\mathrm{II}}$ makes a user partial private key query for ID_i, if $\mathrm{ID}_i = \mathrm{ID}_*$, $\mathscr{C}$ fails and stops. Otherwise challenger $\mathscr{C}$ finds the tuple (ID_i, x_i) in the list L_U. Then challenger $\mathscr{C}$ adds (ID_i, x_i) to the L_E list and returns x_i to $\mathscr{A}_{\mathrm{II}}$.

(viii) *Ring Signature Queries*. Same as that in the proof of Theorem 8.

Forge. Eventually, $\mathscr{A}_{\mathrm{II}}$ outputs a ring signature σ^* fulfilling the following conditions:

(1) σ^* is a valid ring signature.

(2) $\mathscr{A}_{\mathrm{II}}$ cannot query the secret value of anyone in $\mathscr{U}$.

(3) $\mathscr{A}_{\mathrm{II}}$ cannot replace any users' public key in $\mathscr{U}$.

(4) The forged signature σ^* is not from signature query.

Output. It follows from the forking lemma that if $\varepsilon \geq 7C^n_{q_{H_3}}/2^k$, adversary $\mathscr{A}_{\mathrm{II}}$ can give a valid forged signature within time T_A in the above interaction; then we can construct another algorithm $\mathscr{A}'_{\mathrm{II}}$ that outputs two signed messages within time $2T_A$ with probability at least $\varepsilon^2/66C^n_{q_{H_3}}$. For the resemble construction, $\mathscr{C}$ can get two valid ring signature σ and σ' satisfying

$$\begin{aligned}\sigma &= \left(R_1, \ldots, R_n, K_1, K_2, \ldots, K_{s-1}, K_s, K_{s+1}, \ldots, K_n, m, V\right),\\ \sigma' &= \left(R_1, \ldots, R_n, K_1, K_2, \ldots, K_{s-1}, K'_s, K_{s+1}, \ldots, K_n, m, V'\right).\end{aligned} \tag{11}$$

So we have

$$\begin{aligned}V &= rP_{\mathrm{Pub}} + r_s S_{\mathrm{ID}_s} + k_s x_s Q + kQ,\\ V' &= rP_{\mathrm{Pub}} + r_s S_{\mathrm{ID}_s} + k'_s x_s Q + kQ.\end{aligned} \tag{12}$$

Challenger C outputs

$$aX = x_s Q = \left(k_s - k'_s\right)^{-1}\left(V - V'\right). \tag{13}$$

Table 1: Cryptographic operation time (in milliseconds).

PO	T_{PO}	T_E	T_N
20.01	6.38	0.83	5.31

Table 2: Comparison of the efficiency of several certificateless ring signature schemes.

Schemes	Sign phase	Verify phase	Time ($n = 10$)
Scheme [32]	$PO + (3n-2)T_E + T_N$	$nPO + nT_E + T_N$	256.96
Scheme [33]	$2PO + 3nT_E + (2n+1)T_N$	$3PO + 2nT_E$	253.06
Scheme [31]	$2PO + T_{PO} + (2n+1)T_N$	$2PO + T_{PO} + (2n+1)T_N$	315.82
Our scheme	$(2n+4)T_E$	$3PO + T_{PO} + 2nT_E$	102.93

Table 3: Comparison of the security of several certificateless ring signature schemes.

Schemes	Hard problems	Models
Scheme [32]	k-CAA problem and mICDH problem	Random oracle model (ROM)
Scheme [33]	DL problem and CDH problem	ROM
Scheme [31]	DL problem and CDH problem	ROM
Our scheme	CDH problem and co-CDH problem	ROM

Probability. Let $q_{H_1}, q_{H_2}, q_{H_3}, q_E, q_R, q_U$, and q_{RS} be the times of H_1 queries, H_2 queries, H_3 queries, secret value queries, user public key replacement requests, user public key queries, and ring signature queries, respectively.

For simplification, we may assume that $L_E \cap L_R = \phi$. The probability that ID_*'s secret value was not queried and ID_*'s public key was not replaced by $\mathscr{A}_{II}$ during the queries is $(q_U - q_E - q_R)/q_U$. The probability that ID_* belongs to the groups W is $n/(q_U - q_E - q_R)$. The probability that ID_* is the actual signer is $1/n$. So the combined probability is: $(q_U - q_E - q_R)/q_U \cdot n/(q_U - q_E - q_R) \cdot 1/n = 1/q_U$. □

Therefore, according to the forking lemma, if the attacker $\mathscr{A}_{II}$ can succeed in making a valid ring signature with a probability ε, the advantage of challenger $\mathscr{C}$ solving an instance of co-CDH problem in the game 2 is at least $\varepsilon^2/66C^n_{q_{H_3}} \cdot 1/q_U$.

5.3. Unconditional Anonymity

Theorem 10. *Our certificateless ring signature scheme has the property of unconditional anonymity. For any algorithm $\mathscr{A}$, any set of signers $\mathscr{U} = (u_1, u_2, \ldots, u_n)$ and a random $u_s \in \mathscr{U}$, the probability $Pr[\mu = \mu'] = 1/2$, where $\sigma = (R_1, R_2, \ldots, R_n, K_1, K_2, \ldots, K_n, m, V)$ is a ring signature on $\mathscr{U}$ generated by u_s.*

Proof. (i) The challenger $\mathscr{C}$ runs the setup algorithm to obtain a list of system parameters. And challenger $\mathscr{C}$ sends system parameters to adversary $\mathscr{A}$.

(ii) The adversary $\mathscr{A}$ adaptively makes a polynomially bounded number of queries.

(iii) The adversary $\mathscr{A}$ outputs a message m, two different members $ID_1, ID_2 \in W$ to the challenger $\mathscr{C}$. The challenger $\mathscr{C}$ randomly chooses a bit $\mu \in \{0, 1\}$ and sends $\mathscr{A}$ to a ring signature $\sigma = RS(m, W, x_\mu)$.

(iv) The adversary $\mathscr{A}$ can make a polynomially bounded number of queries.

(v) Finally, adversary $\mathscr{A}$ outputs a bit $\mu' \in \{0, 1\}$. □

In our scheme, since R_i, K_i are chosen randomly from $\mathbb{G}_1$, r_i, k_i are also random elements from $\mathbb{Z}_d^*$. Moreover, r_s, k_s are chosen randomly from $\mathbb{Z}_d^*$, so V is also a random element from $\mathbb{G}_1$. For anyone of a set of signers $\mathscr{U}$, message m, the distribution of $\sigma = (R_1, R_2, \ldots, R_n, K_1, K_2, \ldots, K_n, m, V)$ is independently and uniformly distributed no matter who the actual signer is. The fact illustrates that anyone has no advantage to know who signs the certificateless ring signature. Hence, $Pr[\mu = \mu'] = 1/2$; the anonymity holds.

6. Comparison

6.1. Comparison of the Efficiency. We will compare the performance of our scheme with several certificateless ring signature schemes; see Table 2. The running times are listed in Table 1. We define some notations as follows:

(i) PO: a pairing operation.

(ii) T_{PO}: a pairing-based scalar multiplication operation.

(iii) T_E: an ECC-based scalar multiplication operation.

(iv) T_N: a modular exponent operation in $\mathbb{G}_2$.

6.2. Comparison of the Security. We will give the comparison of the security of our scheme and several previous certificateless ring signature schemes [31–33] from the hard problems that these schemes rely on and the models these schemes depend on; see Table 3.

7. Conclusion

There are some certificateless ring signature schemes based on bilinear pairings, which have been proposed over last years. But the computation cost of the pairings is very high. Therefore it is always interesting to design a cryptographic scheme with less pairing operations to speed up the computation of pairing function. In this paper, we propose an efficient certificateless ring signature scheme with only three bilinear parings. We also prove the unforgeability of our signature scheme against type I and type II adversaries in the random oracle based on the hardness of Computational Diffe-Hellman problem and co-Computational Diffe-Hellman problem. From Table 2, we can see that our scheme is more efficient than the previous related schemes. Due to the good properties of our scheme, it is very useful for practical applications.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors' research is supported by the National Science Foundation of China (no. 11261060) and the Scientific Research Fund of Sichuan (no. 2015GZ0333).

References

[1] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[2] A. Shamir, "Identity-based cryptosystems and signature schemes," *Lecture Notes in Computer Science*, vol. 21, no. 2, pp. 47–53, 1995.

[3] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," *Lecture Notes in Computer Science*, vol. 2894, no. 2, pp. 452–473, 2003.

[4] S. Björck, M. Rundgren, and K. Ljung et al., "On the security of a certificateless signature scheme," *International Journal of Distributed Sensor Networks*, vol. 519–520, no. 4, pp. 963–966, 2014.

[5] F. Li and P. Liu, "An efficient certificateless signature scheme from bilinear parings," in *Proceedings of the International Conference on Network Computing and Information Security (NCIS '11)*, pp. 35–37, May 2011.

[6] D. He, J. Chen, and R. Zhang, "An efficient and provably-secure certificateless signature scheme without bilinear pairings," *International Journal of Communication Systems*, vol. 25, no. 11, pp. 1432–1442, 2012.

[7] R. Tso, "Certificateless signatures with message recovery," *Information*, vol. 17, no. 4, pp. 1559–1573, 2014.

[8] W. S. Yap, S. H. Heng, and B. M. Goi, "An efficient certificateless signature scheme," *Lecture Notes in Computer Science*, vol. 4097, pp. 322—331, 2006.

[9] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *Advances in Cryptology-ASIACRYPT 2001*, pp. 552–565, 2000.

[10] C. A. Melchor, P.-L. Cayrel, and P. Gaborit, "A new efficient threshold ring signature scheme based on coding theory," *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, vol. 57, no. 7, pp. 4833–4842, 2011.

[11] H. Sun and Y. Ge, "On the security of certificateless threshold ring signature without random oracles," *Journal of Computational Information Systems*, vol. 10, no. 9, pp. 3585–3592, 2014.

[12] T. H. Yuen, J. K. Liu, and M. H. Au et al., "Threshold ring signature without random oracles," *Research Publications*, vol. 56, no. 4, pp. 261–267, 2011.

[13] C. Hu and P. Liu, "A new ID-based ring signature scheme with constant-size signature," in *Proceedings of the 2nd International Conference on Computer Engineering and Technology (ICCET '10)*, pp. V3-579–V3-581, April 2010.

[14] H. A. Man, J. K. Liu, and W. Susilo et al., "Secure ID-based linkable and revocable-iff-linked ring signature with constant-size construction," *Theoretical Computer Science*, vol. 469, no. 1, pp. 1–14, 2013.

[15] H. Sun, L. Guo, X. Zheng, and A. Wang, "Identity-based threshold ring signature scheme with constant signature size," *Journal of Computer Applications*, vol. 32, no. 32, pp. 1385–1387, 2012.

[16] W. Wenqiang and C. Shaozhen, "Attribute-based ring signature scheme with constant-size signature," *IET Information Security*, vol. 4, no. 2, pp. 104–110, 2010.

[17] M. H. Au, J. K. Liu, and T. H. Yuen et al., "ID-based ring signature scheme secure in the standard model," in *Proceedings of the International Conference on Security*, pp. 1–16, Springer-Verlag, 2006.

[18] H. Shacham and B. Waters, "Efficient ring signatures without random oracles," in *Proceedings of the International Conference on Practice and Theory in Public-Key Cryptography (PKC '07)*, pp. 134–148, Beijing, China, 2007.

[19] Y. U. Ting, Z. M. Zhao, and X. F. Ren, "Efficient identity-based ring signature in standard model," *Journal of Computer Applications*, vol. 32, no. 7, pp. 2015–2017, 2012.

[20] Y. Y Zhang, L. I Hui, and Y. M. Wang, "Identity-based ring signature scheme under standard model," *Journal on Communications*, vol. 29, no. 4, pp. 40–44, 2008.

[21] K.-A. Shim, "An efficient ring signature scheme from pairings," *Information Sciences*, vol. 300, pp. 63–69, 2015.

[22] J. Xu, Z. Zhang, and D. Feng, "A ring signature scheme using bilinear pairings," *Lecture Notes in Computer Science*, vol. 3325, pp. 160–169, 2004.

[23] S. S. M. Chow, S. M. Yiu, and L. C. K. Hui, *Efficient Identity Based Ring Signature*, vol. 3545, 2005.

[24] L. Deng and J. Zeng, "Two new identity-based threshold ring signature schemes," *Theoretical Computer Science*, vol. 535, no. 4, pp. 38–45, 2014.

[25] J. Herranz, "Identity-based ring signatures from RSA," *Theoretical Computer Science*, vol. 389, no. 1-2, pp. 100–117, 2007.

[26] W. M. Lang, Z. K. Yang, B. Pene et al., "An improved identity-based ring signature scheme," *Asian Journal of Information Technology*, vol. 3, no. 9, pp. 752–755, 2004.

[27] C. Y. Lin and T. C. Wu, "An identity-based ring signature scheme from bilinear pairings," in *Proceedings of the International Conference on Advanced Information NETWORKING and Applications*, p. 182, 2014.

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

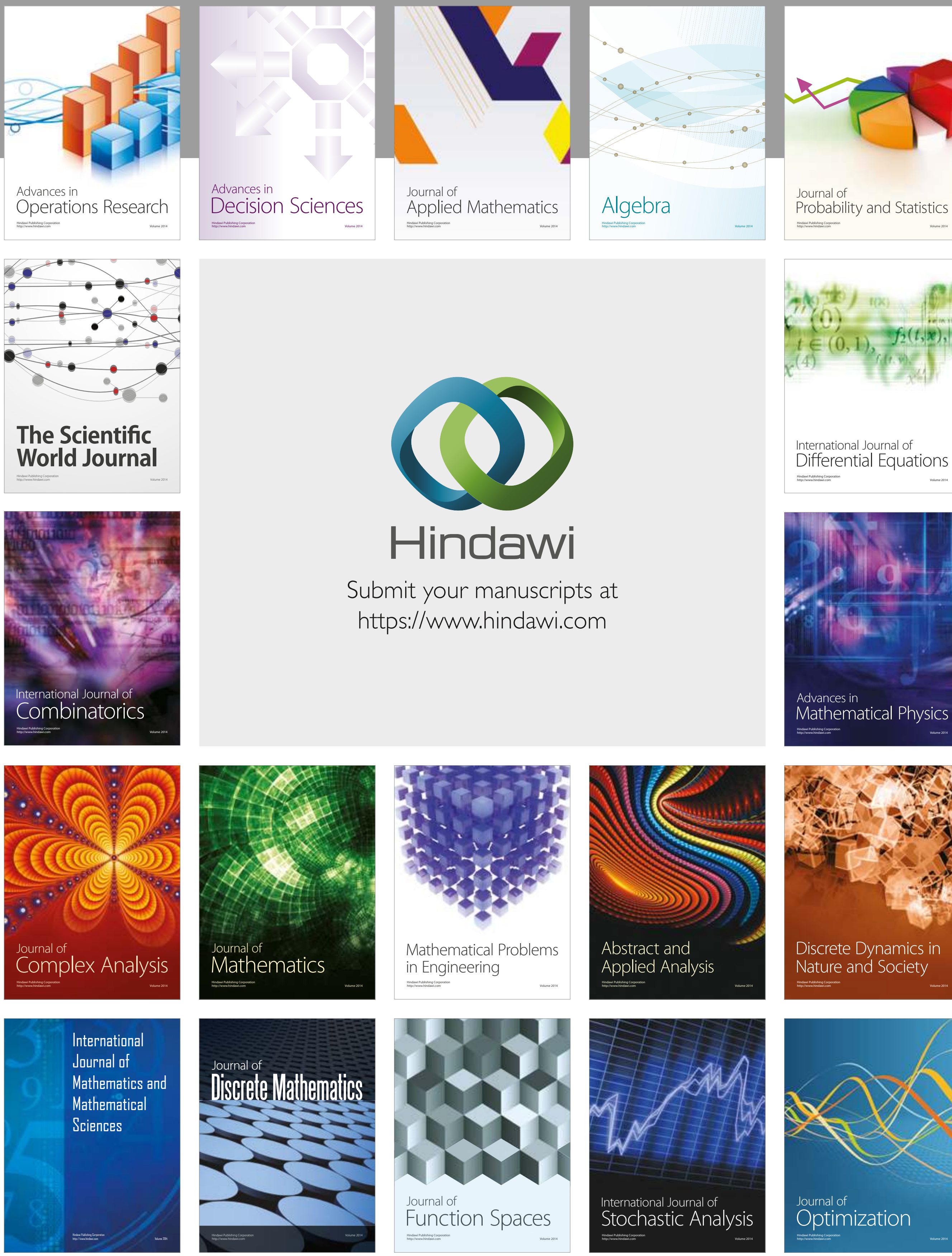
[29] H. Wang and S. Han, "A provably secure threshold ring signature scheme in certificateless cryptography," in *Proceedings of the International Conference of Information Science and Management Engineering (ISME '10)*, pp. 105–108, August 2010.

[30] X. Xia, F. Hong, and G. Cui, "A forward certificateless ring signature scheme," in *Proceedings of the 2nd International Conference on Multimedia Technology (ICMT '11)*, pp. 3315–3318, July 2011.

[31] S. Chang, D. S. Wong, and Y. Mu et al., "Certificateless threshold ring signature," *Information Sciences*, vol. 179, no. 20, pp. 3685–3696, 2009.

[32] S. S. M. Chow and W. S. Yap, "Certificateless ring signatures," *IACR Cryptology ePrint Archive*, vol. 2007, 236 pages, 2007.

[33] L. Zhang, F. F Zhang, and W. Wu, "A provably secure ring signature scheme in certificateless cryptography," in *Proceddings of the 1st International Conference on Provable Security (ProvSec '07)*, vol. 4784, pp. 103–121, Wollongong, Australia, November 2007.


Advances in Operations Research
Advances in Decision Sciences
Journal of Applied Mathematics
Algebra
Journal of Probability and Statistics
The Scientific World Journal
International Journal of Differential Equations
Hindawi
Submit your manuscripts at
https://www.hindawi.com
International Journal of Combinatorics
Advances in Mathematical Physics
Journal of Complex Analysis
Journal of Mathematics
Mathematical Problems in Engineering
Abstract and Applied Analysis
Discrete Dynamics in Nature and Society
International Journal of Mathematics and Mathematical Sciences
Journal of Discrete Mathematics
Journal of Function Spaces
International Journal of Stochastic Analysis
Journal of Optimization