Compact Ring Signature in the Standard Model for Blockchain

Hao Ren[1], Peng Zhang[1(✉)], Qingchun Shentu[1], Joseph K. Liu[2], and Tsz Hon Yuen[3]

[1] College of Information Engineering, Shenzhen University, Shenzhen, China
renhao2016@email.szu.edu.cn, zhangp@szu.edu.cn, shentuqc@bankledger.com
[2] Faculty of Information Technology, Monash University, Melbourne, Australia
joseph.liu@monash.edu
[3] Huawei, Singapore, Singapore
YUEN.TSZ.HON@huawei.com

Abstract. Ring signature is a variant of digital signature, which makes any member in a group generate signatures representing this group with anonymity and unforgeability. In recent years, ring signatures have been employed as a kind of anonymity technology in the blockchain-based cryptocurrency such as Monero. Recently Malavolta et al. introduced a novel ring signature protocol that has anonymity and unforgeability in the standard model [33]. Their construction paradigm is based on non-interactive zero-knowledge (NIZK) arguments of knowledge and re-randomizable keys.

In this work, for the purpose of lower bandwidth cost in blockchain, we improve their ring signature by proposing a compact NIZK argument of knowledge. We show our NIZK holds under a new complexity assumption *Compact Linear Knowledge of Exponent Assumption*. Without the expense of security, our proposed ring signature scheme is anonymous and unforgeable in the standard model. It saves almost half of storage space of signature, and reduces almost half of pairing computations in verification process. When the ring size is large, the effect of our improvements is obvious.

Keywords: Blockchain · Ring signature · NIZK
Argument of knowledge

1 Introduction

In 2008, Satoshi Nakamoto first proposed the blockchain to build cryptocurrency bitcoin as a public transaction ledger [34]. With the decentralization of blockchain, cryptocurrency bitcoin first solves double-spending problem without a central server. The blockchain and bitcoin have also provided inspirations for various applications offering value or trust [41]. In recent years, ring signature was deployed to build transaction protocols for blockchain-based cryptocurrencies. Monero is one of the popular cryptocurrencies that mainly focuses on

C. Su and H. Kikuchi (Eds.): ISPEC 2018, LNCS 11125, pp. 50–65, 2018.
https://doi.org/10.1007/978-3-319-99807-7_4

anonymity, and its underlying CryptoNote protocol deploys ring signature as core cryptographic tools to provide anonymity [36].

The notion of ring signature was first proposed to leak secrets, by Rivest, Shamir and Tauman [35] with many extensions after that such as using different mathematical assumptions [16], based on different cryptosystems [2,4,5], with linkability and/or revocability [1,3,20,22,23,25,27,40], with blinding feature [8], in a threshold setting [24,39,42,44,45], security enhancement [10,18,26,28,30–32] and efficiency improvement [21,29,43]. This cryptographic tool has ability to leak the endorsement of any messages signed by one member in a group, but does not reveal his identification. Compared with the group in group signatures [9], a ring is not managed by a group manager. Actually, ring members can be included in the ring completely unawarely. Since rings are ad-hoc, which means that the signing process cannot be controlled by any centralized authority after original setup.

In the past years, the security of most ring signature constructions holds in ROM (Random Oracle Model) [11] or CRS (Common Reference String) model [19]. In ASIACRYPT 2017, Malavolta et al. presented a generic ring signature construction that has anonymity and unforgeability in the standard model [33]. In their scheme, a ring signature protocol can be divided into two components: the re-randomizable key and the NIZK (Non-Interactive Zero-Knowledge) system. A novel feature of this scheme is that one can modify its NIZK system independently to obtain variants of the original scheme.

Bandwidth usage is one of the main targets for blockchain benchmarks, which influences transaction processing performance of blockchain significantly. To reduce bandwidth in blockchain, Groth et al. proposed a logarithmic-size ring signature for blockchain cryptocurrency [15]. Sun et al. proposed an accumulator-based transaction protocol for Monero to reduce transaction size [38]. These two works are both in the ROM. In this work, to improve the efficiency, we design a new assumption CL-KEA (*Compact Linear Knowledge of Exponent Assumption*), then a compact NIZK argument of knowledge under this assumption is proposed. With the remarkable properties of our compact NIZK, we build a compact ring signature scheme in standard model. Compared with Malavolta et al.'s scheme [33], the signature size of our scheme is smaller, and the verification computation is more efficient.

2 Preliminaries

In this work, we use λ to denote a security parameter, use $\mathsf{negl}(\lambda)$ to denote a negligible function in a security parameter λ, and use $[n]$ to denote a set $\{1, ..., n\}$ for a positive integer $n \in \mathbb{N}$. We define $y \leftarrow S$ for sampling y from a set S randomly.

2.1 Bilinear Maps

Let g_1 and g_2 be generators of two cyclic groups $(\mathbb{G}_1, \mathbb{G}_2)$ of large prime order p, respectively. There exits a homomorphism function $\phi : \mathbb{G}_2 \to \mathbb{G}_1$ and a bilinear map function $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ which holds:

- Non-degeneracy. $e(g_1, g_2) \neq 1$.
- Computability. All group operations in $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$, the homomorphism ϕ and the map e are efficiently computable.
- Bilinearity. For all $(a, b) \in \mathbb{Z}_p^2$ and $(C, D) \in \mathbb{G}_1 \times \mathbb{G}_2$, $e(C^a, D^b) = e(C, D)^{a \cdot b}$.
- Homomorphism. For all $(D, E) \in \mathbb{G}_2^2$, $\phi(D \cdot E) = \phi(D) \cdot \phi(E)$.

2.2 NIZK Arguments of Knowledge

Definition 1 (NIZK Arguments of Knowledge [14]). *Let $\mathcal{R}$ be a relation corresponding to a NP language $\mathcal{L}$.* NIZK *arguments of knowledge have following* PPT *algorithms:*

$(\alpha, \theta) \leftarrow \mathcal{G}(1^\lambda)$*: On input the security parameter λ, this algorithm outputs a trapdoor α and a common reference string θ.*

$\pi \leftarrow \mathcal{P}(\theta, w, s)$*: On input a θ, a witness w and a statement s, where $(w, s) \in \mathcal{R}$, this algorithm outputs a argument π.*

$1/0 \leftarrow \mathcal{V}(\theta, \pi, s)$*: On input a θ, a proof π and a statement s, this algorithm outputs a bit b, which is 1 or 0.*

$\pi \leftarrow \mathcal{S}(\theta, \alpha, s)$*: On input a θ, a trapdoor α and a statement s, this algorithm outputs an argument π.*

$(s, \pi, w) \leftarrow \mathcal{E}(\alpha, \theta)$*: On input a trapdoor α and a θ, this algorithm outputs a statement s, a argument π and a witness w.*

Definition 2 (Perfect Completeness). *For all $\lambda \in \mathbb{N}$, $(\alpha, \theta) \leftarrow \mathcal{G}(1^\lambda)$ and $(w, s) \in \mathcal{R}$ such that*

$$\Pr[(\alpha, \theta) \leftarrow \mathcal{G}(1^\lambda), \pi \leftarrow \mathcal{P}(\theta, w, s) : 1 \leftarrow \mathcal{V}(\theta, \pi, s)] = 1.$$

Definition 3 (Perfect Zero-Knowledge). *For all $\lambda \in \mathbb{N}$, $(\alpha, \theta) \leftarrow \mathcal{G}(1^\lambda)$ and $(w, s) \in \mathcal{R}$, there exists a simulator $\mathcal{S}$ such that*

$$\Pr[\mathcal{P}(\theta, w, s) = \mathcal{S}(\theta, \alpha, s)] = 1.$$

Definition 4 (Computational Knowledge Soundness). *For all $\lambda \in \mathbb{N}$, $(\alpha, \theta) \leftarrow \mathcal{G}(1^\lambda)$, $(w, s) \in \mathcal{R}$ and any* PPT *adversary $\mathcal{A}$, there is an extractor $\mathcal{E}$ that has full access to the adversary it holds that*

$$\Pr\left[\begin{array}{r}(\pi, s) \leftarrow \mathcal{A}(\theta), (s, \pi, w) \leftarrow \mathcal{E}_{\mathcal{A}}(\alpha, \theta) \\ : (w, s) \in \mathcal{R}\end{array}\middle| 1 \leftarrow \mathcal{V}(\theta, \pi, s)\right] \geq 1 - \mathsf{negl}(\lambda).$$

2.3 Ring Signature

Definition 5 (Ring Signature [6]). *A ring signature protocol includes a triple of* PPT *algorithms* $\mathsf{RSig} = (\mathsf{Gen}, \mathsf{Sig}, \mathsf{Ver})$ *as follows:*

$(vk, sk) \leftarrow \mathsf{Gen}(1^\lambda)$*: On input the security parameter* λ*, this algorithm outputs a verification key* vk *and a signing key* sk*. Define the ring* $R = \{vk_i\}_{i\in[n]}$*.*

$\sigma \leftarrow \mathsf{Sig}(R, sk, m)$*: On input a ring* R*, a signing key* sk *and a message* m*, this algorithm outputs a signature* σ*.*

$1/0 \leftarrow \mathsf{Ver}(R, m, \sigma)$*: On input a ring* R*, a message* m *and a signature* σ*, this algorithm outputs a bit* 1 *which means the ring signature passes the verification. Otherwise, output a bit* 0*.*

A ring signature must satisfies ***Anonymity*** *and* ***Unforgeability*** *as defined in* [6]*.*

2.4 Programmable Hash Function

Definition 6 (Programmable Hash Function [17]). *There are two algorithms* $\mathsf{H} = (\mathsf{HGen}, \mathsf{HEval})$ *in the programmable hash function as follows:*

$k \leftarrow \mathsf{HGen}(1^\lambda)$*: On input the security parameter* λ*, this algorithm generates a public key* k*.*

$c \leftarrow \mathsf{HEval}(k, m)$*: On input a public key* k *and a message* $m \in \{0,1\}^*$*, this algorithm outputs a hash value* c*.*

3 Overview of Malavolta et al.'s Scheme

In this section, we show an overview of Malavolta et al.'s scheme [33].

3.1 NIZK

Firstly, we recall the language $\mathcal{L}$ corresponding to disjunction of discrete logarithm defined in [33] as follows:

$$\mathcal{L} = \{\{A_i\}_{i\in[n]} \in \mathbb{G}_1^n : \exists(a, i) : g_1^a = A_i\}.$$

Then we recall the NIZK system of [33] as Fig. 1.

As we can see, this NIZK argument doesn't need random oracles and the security is mainly based on L-KEA (*Linear Knowledge of Exponent Assumption*). We note that although there exists a common reference string in their NIZK, it doesn't mean their ring signatures need the CRS, we talk about it later.

$\mathcal{G}(1^\lambda)$	$\mathcal{P}(\theta, w, s)$	$\mathcal{V}(\theta, \pi, s)$
$\alpha \leftarrow \mathbb{Z}_p$	parse $\theta = T \in \mathbb{G}_2$	parse $\theta = T \in \mathbb{G}_2$
$\theta \leftarrow g_2^\alpha$	$w = (a, j)$	$\pi = \{T_i\}_{i\in[n]} \in \mathbb{G}_2^n$
output (α, θ)	$s = \{A_i\}_{i\in[n]} \in \mathbb{G}_1^n$	$\{Q_i\}_{i\in[n]} \in \mathbb{G}_1^n$
	$\forall i \in [n]\backslash j$:	$s = \{A_i\}_{i\in[n]} \in \mathbb{G}_1^n$
	$t_i \leftarrow \mathbb{Z}_p$	output 1 iff
	$T_i \leftarrow g_2^{t_i}$	$\prod_{i\in[n]} T_i = T \wedge$
	$Q_i \leftarrow (A_i)^{t_i}$	$\forall i \in [n]$:
	$T_j \leftarrow T \cdot (\prod_{i\in[n]\backslash j} g_2^{t_i})^{-1}$	$e(Q_i, g_2) = e(A_i, T_i)$
	$Q_j \leftarrow \phi(T_j^a)$	
	output $\pi = \{(T_i, Q_i)\}_{i\in[n]}$	

Fig. 1. NIZK for disjunctive statements in Malavolta et al.'s scheme [33]

3.2 Ring Signature

Then we show the generic ring signature constructions introduced by Malavolta et al. as Fig. 2. Their novel work is based on re-randomizable keys [12] and the above NIZK arguments of knowledge. To make their ring signature scheme independent with the CRS, they divide the CRS of NIZK into a part of each verification key, achieving that the CRS of NIZK is not the CRS of ring signature. A potential feature of their ring signature is that the NIZK argument of knowledge is a independent component, thus it can be modified with other valid NIZK systems, such as [13,14].

An obvious deficiency of their ring signature scheme is the signature size. In their scheme, a signature includes two proofs of NIZK arguments of knowledge and each proof consists of $2n$ group points for a n-sized ring. Consequently, their signature consists of $(4n + 3)$ group points and an integer.

4 Our NIZK Arguments of Knowledge

We propose a new NIZK argument of knowledge to improve efficiencies of [33]. Our main idea is to compress the size of NIZK argument without changing degrees of the polynomials in the security proof of assumption, thus the security of new NIZK arguments of knowledge holds as before. We note that our NIZK is secure based on CL-KEA, which is a variant of L-KEA.

4.1 Complexity Assumptions

Assumption 1 (Compact Linear Knowledge of Exponent (CL-KEA)). *For all $\lambda \in \mathbb{N}$, $n \in \mathsf{poly}(\lambda)$ and* PPT *adversaries $\mathcal{A}$ there is a* PPT *algorithm $\mathcal{E}_\mathcal{A}$ with full access to $\mathcal{A}$ it holds that*

$\mathsf{Gen}(1^\lambda)$	$\mathsf{Sig}(R, sk, m)$	$\mathsf{Ver}(R, m, \sigma)$
$(\alpha, x) \leftarrow \mathbb{Z}_p^2$	parse $R = \{vk_i\}_{i\in[n]}$	parse $R = \{vk_i\}_{i\in[n]}$
$C \leftarrow g_2^\alpha$	if $\nexists i : vk = vk_i$	$vk_i = (z_i, C_i, k_i)$
$k \leftarrow \mathsf{HGen}(1^\lambda)$	output $\perp$	$\sigma = (\sigma', \pi, z')$
output $(x, (g_2^x, C, k))$	parse $vk = (z, C, k)$	$\sigma' = (s, y, c)$
	$vk_i = (z_i, C_i, k_i)$	$x := R\|\|z'\|\|c\|\|(m, R)$
	$(s, \rho, \delta) \leftarrow \mathbb{Z}_p^3$	$b \leftarrow \mathcal{V}\left(\prod_i C_i, x, \pi\right)$
	$z' \leftarrow z \cdot g_2^\rho$	$b' = 1$ if
	$x' \leftarrow sk + \rho$	$e(y, vk' \cdot g_2^s) = e(c, g_2)$
	$c \leftarrow \mathsf{HEval}(k, m\|\|R)^\delta$	output $(b = b' = 1)$
	$x := R\|\|z'\|\|c\|\|(m, R)$	
	$\pi \leftarrow \mathcal{P}\left(\prod_i C_i, (\rho, \delta, i), x\right)$	
	$y \leftarrow c^{\frac{1}{x'+s}}$	
	$\sigma = (s, y, c)$	
	output (σ, π, z')	

Fig. 2. Ring signature scheme in Malavolta et al.'s scheme [33]

$$\Pr\left[\begin{array}{l|l} (Q, \{T_i, A_i\}_{i\in[n]}) \leftarrow \mathcal{A}(p, e, g_1, g_2, g_2^x), & \sum_{i\in[n]} \mathsf{Dlog}_{g_2}(T_i) \cdot \mathsf{Dlog}_{g_1}(A_i) \\ \quad (a, P, \{T_i, A_i\}_{i\in[n]}) \leftarrow \mathcal{E}_{\mathcal{A}}(p, e, g_1, g_2, g_2^x) & \quad = \mathsf{Dlog}_{g_1}(Q) \\ & \wedge \prod_{i\in[n]} T_i = g_2^x \\ & \wedge \forall i \in [n] : g_1^a \neq A_i \end{array}\right] \leq \mathsf{negl}(\lambda).$$

W.l.o.g., we use $\mathcal{O}$ to represent the set of five oracles with the generic group model from [7] and we randomly pick encoding functions $(\gamma_1, \gamma_2, \gamma_T)$ corresponding to groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ in the following.

Theorem 1. *For all $\lambda \in \mathbb{N}$, $n \in \mathsf{poly}(\lambda)$ and* PPT *adversaries $\mathcal{A}$ with oracle access to $\mathcal{O}$ there is a* PPT *extractor $\mathcal{E}_A$ with full access to $\mathcal{A}$ such that*

$$\Pr\left[\begin{array}{l|l} (\gamma_1(q), \{\gamma_2(t_i), \gamma_1(a_i)\}_{i\in[n]}) \leftarrow \mathcal{A}(p, \gamma_1(1), \gamma_2(1), \gamma_2(x)), & \sum_{i\in[n]} t_i \cdot a_i \\ (a, \gamma_1(q), \{\gamma_2(t_i), \gamma_1(a_i)\}_{i\in[n]}) \leftarrow \mathcal{E}_{\mathcal{A}}(p, \gamma_1(1), \gamma_2(1), \gamma_2(x)) & \quad = q \\ & \wedge \sum_{i\in[n]} t_i \\ & \quad = x \\ & \wedge \forall i \in [n] : \\ & \quad a \neq a_i \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Proof. We construct an extractor $\mathcal{E}$ as follows.

1. $\mathcal{E}$ initializes 3 lists $(\mathcal{W}_1, \mathcal{W}_2, \mathcal{W}_T)$.
2. $\mathcal{E}$ randomly picks $s_1 \leftarrow \{0,1\}^*$, $s_2 \leftarrow \{0,1\}^*$ and $s_x \leftarrow \{0,1\}^*$, then it adds $(1, s_1)$ to $\mathcal{W}_1$, adds $(1, s_2)$ to $\mathcal{W}_2$ and adds (x, s_x) to $\mathcal{W}_1$. We note that the entries of the lists can be denoted by (F, s), where F is a generic polynomial and s is a randomly picked string.
3. $\mathcal{E}$ simulates the queries of $\mathcal{A}$ to the oracle set $\mathcal{O}$:
   - On input 2 strings (s_i, s_j), $\mathcal{E}$ first retrieves F_i and F_j from lists $\mathcal{W}_1$, $\mathcal{W}_2$ or $\mathcal{W}_T$. Next it calculates $F_k = F_i \pm F_j$ and outputs s_k if $(F_k, s_k) \in \mathcal{W}_*$.
   - On input 2 strings (s_i, s_j), $\mathcal{E}$ first retrieves F_i and F_j from lists $\mathcal{W}_1$ or $\mathcal{W}_2$. Next it calculates $F_k = F_i \cdot F_j$ and outputs s_k if $(F_k, s_k) \in \mathcal{W}_T$.
   - On input a string s_k, $\mathcal{E}$ first retrieves F_k from list $\mathcal{W}_2$. Next it outputs s_i if $(F_k, s_i) \in \mathcal{W}_1$.

   Whenever $(F_k, s_*) \notin \mathcal{W}_*$, $\mathcal{E}$ randomly picks $s'_k \leftarrow \{0,1\}^*$, adds (F_k, s'_k) to the corresponding list $\mathcal{W}_*$ and outputs s'_k.
4. At some time, $\mathcal{E}$ receives a tuple $(q, \{a_i, t_i\}_{i\in[n]})$ from $\mathcal{A}$.
5. For all $i \in [n]$, $\mathcal{E}$ retrieves F_{a_i} from list $\mathcal{W}_1$, which corresponds to a_i.
6. If some F_{a_i} is a constant $(\mathsf{deg}_x(F_{a_i}) = 0)$, $\mathcal{E}$ returns F_{a_i}. Otherwise it aborts.

Whenever $\mathcal{E}$ doesn't abort, we denote the element that $\mathcal{E}$ outputs by o, thus $\gamma_1(o) = a_i$. Then we prove this happens with negligible probability.

Our prove includes three lemmas, first we recall the lemma in [37]:

Lemma 1. *Let $F(\{x_i\}_{i\in[m]})$ be a polynomial and $\mathsf{deg}(F) \leq d$, p be the largest prime dividing a integer n' and we randomly generate $\{x_i\}_{i\in[m]} \leftarrow \mathbb{Z}_{n'}^m$ it holds that:*

$$\Pr[F(\{x_i\}_{i\in[m]}) = 0 \mod n'] \leq \frac{d}{p}$$

Lemma 1 provides any polynomials $F = 0$ with deterministic maximum probability. As our extractor described above, we note that $\mathsf{deg}_x(F_i) \leq 1$ and $\mathsf{deg}_x(F_j) \leq 1$, then $\mathsf{deg}_x(F_k) \leq 2$, where $(F_i, s_i) \in \mathcal{W}_1$, $(F_j, s_j) \in \mathcal{W}_2$ and $(F_k, s_k) \in \mathcal{W}_T$.

Lemma 2. *For all $(F_{a_i}, s_{a_i}) \in \mathcal{W}_1$ and $(F_{t_i}, s_{t_i}) \in \mathcal{W}_2$ it holds that:*

$$\Pr[\mathsf{deg}_x(F_{t_i}) = 1 \wedge \mathsf{deg}_x(F_{a_i}) = 1] \leq \mathsf{negl}(\lambda).$$

Proof. Let F_q be a polynomial such that $(F_q, s_q) \in \mathcal{W}_1$, thus $\mathsf{deg}_x(F_q) \leq 1$. If we assume $F_q = \sum_{i\in[n]} F_{t_i} \cdot F_{a_i}$, it is obvious that for all $i \in [n]$ either F_{t_i} or F_{a_i} must be a constant. For some random $x \leftarrow \mathbb{Z}_p$, it is required that $F_q(x) = \sum_{i\in[n]} F_{t_i}(x) \cdot F_{a_i}(x)$.

By Lemma 1 we know that:

$$\Pr[F_q(x) - \sum_{i\in[n]} F_{t_i}(x) \cdot F_{a_i}(x) = 0] \leq \frac{1}{p}$$

where $\frac{1}{p}$ is negligible. It follows that

$$\Pr[F_q - \sum_{i\in[n]} F_{t_i} \cdot F_{a_i} \neq 0] \leq \frac{1}{p}.$$

Then we conclude that

$$\Pr[\mathsf{deg}_x(F_{t_i}) = 0 \vee \mathsf{deg}_x(F_{a_i}) = 0] \geq \epsilon(\lambda)$$

where ϵ is a non-negligible function. □

Here we note that $\mathsf{deg}_x(F_{t_i}) = \mathsf{deg}_x(F_{a_i}) = 0$ doesn't contradict our theorem.

Lemma 3. *For all* $(F_{t_i}, s_{t_i}) \in \mathcal{W}_2$*:*

$$\Pr[\forall i \in [n] : \mathsf{deg}_x(F_{t_i}) = 0] \leq \mathsf{negl}(\lambda).$$

Proof. We assume that for all $i \in [n]$:

$$\Pr[\forall i \in [n] : \mathsf{deg}_x(F_{t_i}) = 0] \geq \epsilon(\lambda).$$

As we argued that $\sum_{i\in[n]} F_{t_i}(x) = x$, it is required that

$$\Pr[\sum_{i\in[n]} F_{t_i}(x) - x = 0] \geq \epsilon(\lambda)$$

where $\sum_{i\in[n]} F_{t_i}(x)$ is some random constant. Obviously this contradicts Lemma 1. Thus we conclude that there exits at least one i such that $\mathsf{deg}_x(F_{t_i}) = 0$.

□

By Lemmas 2 and 3 we show that there exits an i:

$$\Pr[\mathsf{deg}_x(F_{t_i}) = 1 \wedge \mathsf{deg}_x(F_{a_i}) = 0] \leq \mathsf{negl}(\lambda)$$

which follows that the extractor $\mathcal{E}$ returns o with negligible probability. □

4.2 Our Construction

Then we propose a new NIZK argument of knowledge. Our scheme is described in Fig. 3. The biggest improvement we make is to sum all Q_i to obtain one element Q in the process of proving, and then we replace Q_i with Q to reduce the size of argument. At the same time, the smaller argument size yields less pairing computations in the verification process. Thus our construction saves almost half of storage space of signature and reduces almost half of pairing computations. When n is large, the effect of this improvement is obvious.

Theorem 2. *The scheme in Fig. 3 has perfect zero-knowledge.*

Proof. We construct a simulator $\mathcal{S}(\theta, \alpha, s)$ to prove perfect zero-knowledge as follows:

$\mathcal{G}(1^\lambda)$	$\mathcal{P}(\theta, w, s)$	$\mathcal{V}(\theta, \pi, s)$
$\alpha \leftarrow \mathbb{Z}_p$	parse $\theta = T \in \mathbb{G}_2$	parse $\theta = T \in \mathbb{G}_2$
$\theta \leftarrow g_2^\alpha$	$w = (a, j)$	$\pi = \{T_i\}_{i\in[n]} \in \mathbb{G}_2^n$
output (θ, α)	$s = \{A_i\}_{i\in[n]} \in \mathbb{G}_1^n$	$Q \in \mathbb{G}_1$
	$\forall i \in [n]\backslash j$:	$s = \{A_i\}_{i\in[n]} \in \mathbb{G}_1^n$
	$t_i \leftarrow \mathbb{Z}_p$	output 1 iff
	$T_i \leftarrow g_2^{t_i}$	$\prod_{i\in[n]} T_i = T \wedge$
	$Q_i \leftarrow (A_i)^{t_i}$	$\prod_{i\in[n]} e(A_i, T_i) = e(Q, g_2)$
	$T_j \leftarrow T \cdot (\prod_{i\in[n]\backslash j} g_2^{t_i})^{-1}$	
	$Q_j \leftarrow \phi(T_j^a)$	
	$Q = \prod_{i\in[n]} Q_i$	
	output $\pi = (Q, \{T_i\}_{i\in[n]})$	

Fig. 3. NIZK for disjunctive statements.

1. $\mathcal{S}$ parses the common reference string θ as $T \in \mathbb{G}_2$ and parses a statement s as $\{A_i\}_{i\in[n]} \in \mathbb{G}_1^n$.
2. $\mathcal{S}$ randomly picks a $j \leftarrow [n]$ and $\{t_i\}_{i\in[n]\backslash j} \leftarrow \mathbb{Z}_p^{n-1}$, it computes $\{T_i = (g_2)^{t_i}\}_{i\in[n]\backslash j}$ and $\{Q_i = (A_i)^{t_i}\}_{i\in[n]\backslash j}$.
3. $\mathcal{S}$ computes

$$T_j = \frac{T}{\prod_{i\in[n]\backslash j} g_2^{t_i}}$$

$$Q_j = A_j^{\alpha - \sum_{i\in[n]\backslash j} t_i}$$

$$Q = \prod_{i\in[n]} Q_i.$$

4. $\mathcal{S}$ outputs $(Q, \{T_i\}_{i\in[n]})$.

As this simulation is efficient, we note that $\{T_i\}_{i\in[n]}$ is picked identically to $\mathcal{P}$ and $Q = \prod_{i\in[n]} A_i^{\mathsf{Dlog}_{g_1}(T_i)}$. It shows that the scheme has perfect zero-knowledge. □

Theorem 3. *The scheme in Fig. 3 has computational knowledge soundness.*

Proof. We construct an extractor $\mathcal{E}$ to prove computational knowledge soundness as follows:

$\mathcal{E}(\alpha, \theta)$. This extractor runs the adversaries $\mathcal{A}$ on the θ and receives $(s = \{A_i\}_{i\in[n]}, \pi = (Q, \{T_i\})$. As we defined above, $\mathcal{E}$ has full access to $\mathcal{A}$ to obtain (s, π, w). For all $i \in [n]$, it outputs (a, i) when $A_i = g_1^a$.

We note that if $\prod_{i\in[n]} T_i = T = g_2^{\alpha}$ and $\mathsf{Dlog}_{g_1}(Q) = \sum_{i\in[n]} \mathsf{Dlog}_{g_2}(T_i) \cdot \mathsf{Dlog}_{g_1}(A_i)$, the extraction is successful. As CL-KEA we described above, it happens with $\epsilon(\lambda)$. □

5 Compact Ring Signature

In this section, we present a compact ring signature scheme based on our proposed NIZK arguments of knowledge. Before introducing our ring signature scheme, we first recall the corresponding language described in [33].

$$\mathcal{L} = \left\{ \begin{array}{r} (\{k_i\}_{i\in[n]}, c, \{z_i\}_{i\in[n]}, z', m) \in \mathbb{G}_1^{\lambda\cdot n+1} \times \mathbb{G}_2^{n+1} \times \{0,1\}^* : \\ \exists(\rho, \delta, i) : \frac{z'}{z_i} = g_2^{\rho} \wedge c = \mathsf{HEval}(k_i, m)^{\delta} \end{array} \right\}.$$

This language can be separated into two sub-languages as follows:

$$\mathcal{L}_1 = \left\{ \begin{array}{r} (\{z_i\}_{i\in[n]}, z') \in \mathbb{G}_2^{n+1} : \\ \exists(\rho, i) : \frac{z'}{z_i} = g_2^{\rho} \end{array} \right\}.$$

$$\mathcal{L}_2 = \left\{ \begin{array}{r} (\{k_i\}_{i\in[n]}, c, m) \in \mathbb{G}_1^{\lambda\cdot n+1} \times \{0,1\}^* : \\ \exists(\delta, i) : c = \mathsf{HEval}(k_i, m)^{\delta} \end{array} \right\}.$$

We note that $\mathcal{L}$ essentially includes two NIZK arguments of knowledge for disjunctive discrete logarithms $(\frac{z'}{z_i}, \rho)$ and (c, δ) as above. It is easy to see the first language $\mathcal{L}_1$ works well with their NIZK arguments of knowledge. However we have no idea for the second one, in their scheme the set $\{\mathsf{HEval}(k_i, m)^{\delta}\}_{i\in[n]\setminus j}$ is not public to all and not generated. To make it compatible we make some small changes such that:

$$\mathcal{L}_2' = \left\{ \begin{array}{r} (\{k_i\}_{i\in[n]}, c, m) \in \mathbb{G}_1^{\lambda\cdot n} \times \mathbb{G}_2 \times \{0,1\}^* : \\ \exists(\frac{1}{\delta}, i) : \mathsf{HEval}(k_i, m) = c^{\frac{1}{\delta}} \end{array} \right\}.$$

First we change the witness from (δ, i) to $(\frac{1}{\delta}, i)$, thus the corresponding disjunctive discrete logarithm becomes $(\mathsf{HEval}(k_i, m), \frac{1}{\delta})$. Then we change the range of hash function from $\mathbb{G}_1$ to $\mathbb{G}_2$. From these two changes, it is easy to show that both $\mathcal{L}_1$ and $\mathcal{L}_2'$ can work well with their NIZK arguments of knowledge, same to ours. More details about this feature are shown in Figs. 4 and 5.

Formally, we combine $\mathcal{L}_1$ and $\mathcal{L}_2'$ as follows:

$$\mathcal{L}' = \left\{ \begin{array}{r} (\{k_i\}_{i\in[n]}, \{z_i\}_{i\in[n]}, z', c, m) \in \mathbb{G}_1^{\lambda\cdot n} \times \mathbb{G}_2^{n+2} \times \{0,1\}^* : \\ \exists(\rho, \frac{1}{\delta}, i) : \frac{z'}{z_i} = g_2^{\rho} \wedge \mathsf{HEval}(k_i, m) = c^{\frac{1}{\delta}} \end{array} \right\}.$$

5.1 Scheme Description

Based on primitives, our ring signature $\mathsf{RSig} = (\mathsf{Gen}, \mathsf{Sig}, \mathsf{Ver})$ includes three algorithms as follows:

$\mathsf{Gen}(1^\lambda)$: on input a security parameter λ, this algorithm randomly picks $x \leftarrow \mathbb{Z}_p$, $\beta \leftarrow \mathbb{Z}_p$ and generates k by calling $\mathsf{HGen}(1^\lambda)$. It calculates $z = g_1^x$ and $C = g_2^\beta$, outputs (sk, vk), where $vk = (z, k, C)$ is a verification key and $sk = x$ is a signing key.

$\mathsf{Sig}(R, sk_j, m)$: on input $R = \{vk_i\}_{i\in[n]}$, a signing key sk_j and a message m, this algorithm randomly picks $(s, \rho, \delta) \leftarrow \mathbb{Z}_p^3$, generates a re-randomizable signing key $sk'_j = sk_j + \rho$ and corresponding re-randomizable verification key $z'_j = z_j \cdot g_1^\rho$, computes $c_i = \phi(\mathsf{HEval}(k_i, m||R)) \in \mathbb{G}_1$, $c = \mathsf{HEval}(k_j, m||R)^\delta \in \mathbb{G}_2$ and $y = c^{\frac{1}{x'+s}}$. This algorithm proves two statements as follows:

- Prove a statement (R, z') by calling $\mathcal{P}\left(\prod_{i\in[n]} C_i, (R, z'), (\rho, j)\right)$ as Fig. 4 and outputs π_1.
- Call $\mathcal{P}\left(\prod_{i\in[n]} C_i, (R, c_i, c), (\frac{1}{\delta}, j)\right)$ to prove a statement (R, c_i) as Fig. 4 and outputs π_2.

As a result, this algorithm outputs $\sigma = (\pi_1, \pi_2, c, y, s, z')$.

$\mathsf{Verify}(R, m, \sigma)$: on input a ring $R = \{vk_i\}_{i\in[n]}$, a message m and a signature σ, compute $c_i = \phi(\mathsf{HEval}(k_i, m||R)) \in \mathbb{G}_1$. First this algorithm verifies two statements as follows:

- Verify a statement (R, z') by calling $\mathcal{V}\left(\prod_{i\in[n]} C_i, (R, z'), \pi_1\right)$ as Fig. 5 and outputs b_1.

$\mathcal{P}\left(\prod_{i\in[n]} C_i, (R, z'), (\rho, j)\right)$	$\mathcal{P}\left(\prod_{i\in[n]} C_i, (R, c_i, c), (\frac{1}{\delta}, j)\right)$
$T = \prod_{i\in[n]} C_i \in \mathbb{G}_2$	$T = \prod_{i\in[n]} C_i \in \mathbb{G}_2$
$\forall i \in [n]\backslash j$:	$\forall i \in [n]\backslash j$:
$t_i \leftarrow \mathbb{Z}_p$	$t_i \leftarrow \mathbb{Z}_p$
$T_i \leftarrow g_2^{t_i}$	$T_i \leftarrow c^{t_i}$
$Q_i \leftarrow (\frac{z'}{z_i})^{t_i}$	$Q_i \leftarrow (c_i)^{t_i}$
$T_j \leftarrow T \cdot (\prod_{i\in[n]\backslash j} T_i)^{-1}$	$T_j \leftarrow T \cdot (\prod_{i\in[n]\backslash j} T_i)^{-1}$
$Q_j \leftarrow \phi(T_j^\rho)$	$Q_j \leftarrow \phi(T_j^{\frac{1}{\delta}})$
$Q = \prod_{i\in[n]} Q_i$	$Q = \prod_{i\in[n]} Q_i$
output $\pi_1 = (Q, \{T_i\}_{i\in[n]})$	output $\pi_2 = (Q, \{T_i\}_{i\in[n]})$

Fig. 4. Proving of NIZK arguments of knowledge.

$\mathcal{V}\left(\prod_{i\in[n]} C_i, (R, z'), \pi_1\right)$

output 1 iff

$$\prod_{i\in[n]} T_i = \prod_{i\in[n]} C_i \wedge$$
$$\prod_{i\in[n]} e(\frac{z'}{z_i}, T_i) = e(Q, g_2)$$

where

$$\begin{aligned}
&\prod_{i\in[n]} e(\frac{z'}{z_i}, T_i) \\
&= \prod_{i\in[n]} e(g_1, T_i)^{sk'-sk_i} \\
&= \prod_{i\in[n]\setminus j} e(g_1, g_2)^{(sk'-sk_i)\cdot t_i} \cdot e(g_1^{sk'-sk_j}, T_j) \\
&= \prod_{i\in[n]\setminus j} e(g_1, g_2)^{(sk'-sk_i)\cdot t_i} \cdot e(g_1, T_j^{sk'-sk_j}) \\
&= \prod_{i\in[n]\setminus j} e(Q_i, g_2) \cdot e(Q_j, g_2) \\
&= \prod_{i\in[n]} e(Q_i, g_2) \\
&= e(Q, g_2)
\end{aligned}$$

$\mathcal{V}\left(\prod_{i\in[n]} C_i, (R, c_i, c), \pi_2\right)$

output 1 iff

$$\prod_{i\in[n]} T_i = \prod_{i\in[n]} C_i \wedge$$
$$\prod_{i\in[n]} e(c_i, T_i) = e(Q, c)$$

where

$$\begin{aligned}
&\prod_{i\in[n]} e(c_i, T_i) \\
&= \prod_{i\in[n]\setminus j} e(c_i, c)^{t_i} \cdot e(c_j, T_j) \\
&= \prod_{i\in[n]\setminus j} e(c_i, c)^{t_i} \cdot e(c_j^{\delta}, T_j^{\frac{1}{\delta}}) \\
&= \prod_{i\in[n]\setminus j} e(c_i^{t_i}, c) \cdot e(Q_j, c) \\
&= \prod_{i\in[n]\setminus j} e(Q_i, c) \cdot e(Q_j, c) \\
&= e(Q, c)
\end{aligned}$$

Fig. 5. Verification of NIZK arguments of knowledge.

– Verify a statement (R, c_i) by calling $\mathcal{V}\left(\prod_{i\in[n]} C_i, (R, c_i, c), \pi_2\right)$ as Fig. 5 and outputs b_2.

Then if $e(z' \cdot g_1^s, y) = e(g_1, c) \wedge b_1 = 1 \wedge b_2 = 1$ it returns 1. Otherwise it returns 0.

5.2 Scheme Analysis

The **Anonymity** and **Unforgeability** of this kind of ring signature have been proven in [33], we don't show details again. We compare Malavolta et al.'s scheme and ours in Table 1.

As shown in the table, both L-KEA and CL-KEA are secure in the generic group model, thus the improvements are not at the expense of security. On the other hand, we do not change the sizes of signing key and verification key. Our main contribution is that we reduce almost half of the signature size and half of pairing computations in verification, when n is large.

Table 1. Comparisons between Malavolta et al.'s scheme[33] and ours

Ring signature	[33]	Ours
Model	Standard	Standard
Anonymity	✓	✓
Unforgeability	✓	✓
Assumption	q-SDH + L-KEA	q-SDH + CL-KEA
Ring size	$\mathsf{poly}(\lambda)$	$\mathsf{poly}(\lambda)$
Signing key size	$\mathbb{Z}_p$	$\mathbb{Z}_p$
Verification key size	$(\lambda+2)\mathbb{G}$	$(\lambda+2)\mathbb{G}$
Signature size	$(4\cdot n+3)\mathbb{G}+\mathbb{Z}_p$	$(2\cdot n+5)\mathbb{G}+\mathbb{Z}_p$
Signing computations	$(4\cdot n+3)\mathrm{E}+n\mathrm{H}$	$(4\cdot n+3)\mathrm{E}+n\mathrm{H}$
Verification computations	$(4\cdot n+2)\mathrm{P}+\mathrm{E}+n\mathrm{H}$	$(2\cdot n+4)\mathrm{P}+\mathrm{E}+n\mathrm{H}$

Here we denote an exponentiation computation by E, a bilinear pairing computation by P and a hash function computation by H.

6 Conclusion

In this work, first we propose a new NIZK argument of knowledge. With its good properties, a compact ring signature scheme is constructed in the standard model. Compared with the Malavolta et al.'s scheme [33], our construction reduces the signature size and pairing computations in verification process. We believe this improvement will reduce bandwidth cost in blockchain in the future.

Acknowledgement. This work was supported by the National Natural Science Foundation of China (61702342), the Science and Technology Innovation Projects of Shenzhen (GJHZ 20160226202520268, JCYJ 20170302151321095, JCYJ 20170302145623566) and Tencent "Rhinoceros Birds" -Scientific Research Foundation for Young Teachers of Shenzhen University.

References

1. Au, M.H., Liu, J.K., Susilo, W., Yuen, T.H.: Constant-size ID-based linkable and revocable-iff-linked ring signature. In: Barua, R., Lange, T. (eds.) INDOCRYPT 2006. LNCS, vol. 4329, pp. 364–378. Springer, Heidelberg (2006). https://doi.org/10.1007/11941378_26
2. Au, M.H., Liu, J.K., Susilo, W., Yuen, T.H.: Certificate based (linkable) ring signature. In: Dawson, E., Wong, D.S. (eds.) ISPEC 2007. LNCS, vol. 4464, pp. 79–92. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72163-5_8
3. Au, M.H., Liu, J.K., Susilo, W., Yuen, T.H.: Secure ID-based linkable and revocable-iff-linked ring signature with constant-size construction. Theor. Comput. Sci. **469**, 1–14 (2013)
4. Au, M.H., Liu, J.K., Susilo, W., Zhou, J.: Realizing fully secure unrestricted ID-based ring signature in the standard model based on HIBE. IEEE Trans. Inf. Forensics Secur. **8**(12), 1909–1922 (2013)

5. Au, M.H., Liu, J.K., Yuen, T.H., Wong, D.S.: ID-based ring signature scheme secure in the standard model. In: Yoshiura, H., Sakurai, K., Rannenberg, K., Murayama, Y., Kawamura, S. (eds.) IWSEC 2006. LNCS, vol. 4266, pp. 1–16. Springer, Heidelberg (2006). https://doi.org/10.1007/11908739_1
6. Bender, A., Katz, J., Morselli, R.: Ring signatures: stronger definitions, and constructions without random oracles. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 60–79. Springer, Heidelberg (2006). https://doi.org/10.1007/11681878_4
7. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-28628-8_3
8. Chan, T.K., Fung, K., Liu, J.K., Wei, V.K.: Blind spontaneous anonymous group signatures for Ad Hoc groups. In: Castelluccia, C., Hartenstein, H., Paar, C., Westhoff, D. (eds.) ESAS 2004. LNCS, vol. 3313, pp. 82–94. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-30496-8_8
9. Chaum, D., van Heyst, E.: Group signatures. In: Davies, D.W. (ed.) EUROCRYPT 1991. LNCS, vol. 547, pp. 257–265. Springer, Heidelberg (1991). https://doi.org/10.1007/3-540-46416-6_22
10. Chow, S.S.M., Wei, V.K., Liu, J.K., Yuen, T.H.: Ring signatures without random oracles. In: ASIACCS 2006, pp. 297–302. ACM (2006)
11. Dodis, Y., Kiayias, A., Nicolosi, A., Shoup, V.: Anonymous identification in *Ad Hoc* groups. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 609–626. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24676-3_36
12. Fleischhacker, N., Krupp, J., Malavolta, G., Schneider, J., Schröder, D., Simkin, M.: Efficient unlinkable sanitizable signatures from signatures with re-randomizable keys. In: Cheng, C.-M., Chung, K.-M., Persiano, G., Yang, B.-Y. (eds.) PKC 2016. LNCS, vol. 9614, pp. 301–330. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49384-7_12
13. Groth, J.: Simulation-sound NIZK proofs for a practical language and constant size group signatures. In: Lai, X., Chen, K. (eds.) ASIACRYPT 2006. LNCS, vol. 4284, pp. 444–459. Springer, Heidelberg (2006). https://doi.org/10.1007/11935230_29
14. Groth, J.: On the size of pairing-based non-interactive arguments. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016. LNCS, vol. 9666, pp. 305–326. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49896-5_11
15. Groth, J., Kohlweiss, M.: One-out-of-many proofs: or how to leak a secret and spend a coin. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015. LNCS, vol. 9057, pp. 253–280. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46803-6_9
16. Herranz, J., Sáez, G.: Forking lemmas for ring signature schemes. In: Johansson, T., Maitra, S. (eds.) INDOCRYPT 2003. LNCS, vol. 2904, pp. 266–279. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-24582-7_20
17. Hofheinz, D., Kiltz, E.: Programmable hash functions and their applications. J. Cryptol. **25**(3), 484–527 (2012)
18. Huang, X., et al.: Cost-effective authentic and anonymous data sharing with forward security. IEEE Trans. Comput. **64**(4), 971–983 (2015)
19. Lai, R.W.F., Zhang, T., Chow, S.S.M., Schröder, D.: Efficient sanitizable signatures without random oracles. In: Askoxylakis, I., Ioannidis, S., Katsikas, S., Meadows, C. (eds.) ESORICS 2016. LNCS, vol. 9878, pp. 363–380. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45744-4_18

20. Liu, D.Y.W., Liu, J.K., Mu, Y., Susilo, W., Wong, D.S.: Revocable ring signature. J. Comput. Sci. Technol. **22**(6), 785–794 (2007)
21. Liu, J.K., Au, M.H., Susilo, W., Zhou, J.: Online/offline ring signature scheme. In: Qing, S., Mitchell, C.J., Wang, G. (eds.) ICICS 2009. LNCS, vol. 5927, pp. 80–90. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-11145-7_8
22. Liu, J.K., Au, M.H., Susilo, W., Zhou, J.: Linkable ring signature with unconditional anonymity. IEEE Trans. Knowl. Data Eng. **26**(1), 157–165 (2014)
23. Liu, J.K., Susilo, W., Wong, D.S.: Ring signature with designated linkability. In: Yoshiura, H., Sakurai, K., Rannenberg, K., Murayama, Y., Kawamura, S. (eds.) IWSEC 2006. LNCS, vol. 4266, pp. 104–119. Springer, Heidelberg (2006). https://doi.org/10.1007/11908739_8
24. Liu, J.K., Wei, V.K., Wong, D.S.: A separable threshold ring signature scheme. In: Lim, J.-I., Lee, D.-H. (eds.) ICISC 2003. LNCS, vol. 2971, pp. 12–26. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24691-6_2
25. Liu, J.K., Wei, V.K., Wong, D.S.: Linkable spontaneous anonymous group signature for Ad Hoc groups. In: Wang, H., Pieprzyk, J., Varadharajan, V. (eds.) ACISP 2004. LNCS, vol. 3108, pp. 325–335. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-27800-9_28
26. Liu, J.K., Wong, D.S.: On the security models of (threshold) ring signature schemes. In: Park, C., Chee, S. (eds.) ICISC 2004. LNCS, vol. 3506, pp. 204–217. Springer, Heidelberg (2005). https://doi.org/10.1007/11496618_16
27. Liu, J.K., Wong, D.S.: Linkable ring signatures: security models and new schemes. In: Gervasi, O., et al. (eds.) ICCSA 2005. LNCS, vol. 3481, pp. 614–623. Springer, Heidelberg (2005). https://doi.org/10.1007/11424826_65
28. Liu, J.K., Wong, D.S.: Enhanced security models and a generic construction approach for linkable ring signature. Int. J. Found. Comput. Sci. **17**(6), 1403–1422 (2006). https://doi.org/10.1142/S0129054106004480
29. Liu, J.K., Wong, D.S.: A more efficient instantiation of witness-indistinguishable signature. I. J. Netw. Secur. **5**(2), 199–204 (2007)
30. Liu, J.K., Wong, D.S.: Solutions to key exposure problem in ring signature. I. J. Netw. Secur. **6**(2), 170–180 (2008)
31. Liu, J.K., Yeo, S.L., Yap, W., Chow, S.S.M., Wong, D.S., Susilo, W.: Faulty instantiations of threshold ring signature from threshold proof-of-knowledge protocol. Comput. J. **59**(7), 945–954 (2016)
32. Liu, J.K., Yuen, T.H., Zhou, J.: Forward secure ring signature without random oracles. In: Qing, S., Susilo, W., Wang, G., Liu, D. (eds.) ICICS 2011. LNCS, vol. 7043, pp. 1–14. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25243-3_1
33. Malavolta, G., Schröder, D.: Efficient ring signatures in the standard model. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017. LNCS, vol. 10625, pp. 128–157. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70697-9_5
34. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008). https://bitcoin.org/bitcoin.pdf
35. Rivest, R.L., Shamir, A., Tauman, Y.: How to leak a secret. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 552–565. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45682-1_32
36. van Saberhagen, N.: Cryptonote v 2.0 (2013). https://cryptonote.org/whitepaper.pdf
37. Schwartz, J.T.: Fast probabilistic algorithms for verification of polynomial identities. J. ACM **27**(4), 701–717 (1980)

38. Sun, S.-F., Au, M.H., Liu, J.K., Yuen, T.H.: RingCT 2.0: a compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency monero. In: Foley, S.N., Gollmann, D., Snekkenes, E. (eds.) ESORICS 2017. LNCS, vol. 10493, pp. 456–474. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66399-9_25
39. Tsang, P.P., Au, M.H., Liu, J.K., Susilo, W., Wong, D.S.: A suite of non-pairing ID-based threshold ring signature schemes with different levels of anonymity (extended abstract). In: Heng, S.-H., Kurosawa, K. (eds.) ProvSec 2010. LNCS, vol. 6402, pp. 166–183. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16280-0_11
40. Tsang, P.P., Wei, V.K., Chan, T.K., Au, M.H., Liu, J.K., Wong, D.S.: Separable linkable threshold ring signatures. In: Canteaut, A., Viswanathan, K. (eds.) INDOCRYPT 2004. LNCS, vol. 3348, pp. 384–398. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30556-9_30
41. Wijaya, D.A., Liu, J.K., Suwarsono, D.A., Zhang, P.: A new blockchain-based value-added tax system. In: Okamoto, T., Yu, Y., Au, M.H., Li, Y. (eds.) ProvSec 2017. LNCS, vol. 10592, pp. 471–486. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68637-0_28
42. Wong, D.S., Fung, K., Liu, J.K., Wei, V.K.: On the RS-code construction of ring signature schemes and a threshold setting of RST. In: Qing, S., Gollmann, D., Zhou, J. (eds.) ICICS 2003. LNCS, vol. 2836, pp. 34–46. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-39927-8_4
43. Yang, X., Wu, W., Liu, J.K., Chen, X.: Lightweight anonymous authentication for Ad Hoc group: a ring signature approach. In: Au, M.-H., Miyaji, A. (eds.) ProvSec 2015. LNCS, vol. 9451, pp. 215–226. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26059-4_12
44. Yuen, T.H., Liu, J.K., Au, M.H., Susilo, W., Zhou, J.: Threshold ring signature without random oracles. In: ASIACCS 2011, pp. 261–267. ACM (2011)
45. Yuen, T.H., Liu, J.K., Au, M.H., Susilo, W., Zhou, J.: Efficient linkable and/or threshold ring signature without random oracles. Comput. J. **56**(4), 407–421 (2013)