Anonymous Post-Quantum Cryptocash

Huang Zhang[1,2], Fangguo Zhang[1,2] ⋆, Haibo Tian[1,2], and Man Ho Au[3]

[1] School of Data and Computer Science, Sun Yat-sen University,
Guangzhou 510006, China
[2] Guangdong Key Laboratory of Information Security,
Guangzhou 510006, China
[3] Department of Computing, The Hong Kong Polytechnic University,
Hong Kong, China

Abstract. In this paper, we construct an anonymous and decentralized cryptocash system which is potentially secure against quantum computers. In order to achieve that, a linkable ring signature based on ideal lattices is proposed. The size of a signature in our scheme is $O(\log N)$, where N is the cardinality of the ring. The framework of our cryptocash system follows that of CryptoNote with some modifications. By adopting the short quantum-resistant linkable ring signature scheme, our system is anonymous and efficient. We also introduce how to generate the verifying and signing key pairs of the linkable ring signature temporarily. With these techniques, the privacy of users is protected, even though their transactions are recorded in the public ledger.

1 Introduction

Electronic currencies or cryptocash systems have been proposed for many years. But none of them are prevalent before the Bitcoin system appears. Bitcoin was first described by Satoshi Nakamoto in 2008 [25]. Its success is partially due to its properties of decentralization and anonymity. To prevent "double spending", the system maintains the history of transactions among most nodes in a peer-to-peer network. A consensus mechanism called proof-of-work is used to maintain the history.

Later, researchers find that the public history of Bitcoin causes weaknesses which violate its original designing goals. The latest result states that Bitcoin only addresses the anonymity and unlinkability issues partially [3]. For example, multiple public keys of the same user can potentially be linked when a user gets changes back, in which case two or more of a single user's public keys will appear in the same transaction [28]. Recently, there are more discussions about the weak anonymity of Bitcoin [27, 30]. Although this weakness can be overcome by adopting mixing and distributed methods, the solutions have to include a trusted third party which is a violation to the decentralization property.

There are some creative works to design a strong anonymous cryptocash system. Miers *et al.* [23] proposed "Zerocoin" that allows users to spend their coins

⋆ Corresponding author, email: `isszhfg@mail.sysu.edu.cn` using anonymous proof of ownership instead of explicit public-key based digital signatures. Saberhagen presented two properties, namely, "untraceability" and "unlinkability", which a fully anonymous cryptocash model must satisfy. They designed the "CryptoNote" system with these properties [31]. Monero is a system based on CryptoNote. In CryptoNote, to provide anonymity, there are two ways for all transactions on the network: (1) hiding the sender's address using ring signatures, (2) hiding the receiver's address using stealth addresses. Both sending and receiving addresses are verifying keys of a ring signature scheme. The ring signature can also be used in the Zerocoin system [12].

The ring signature, introduced by Rivest *et al.* [29], permits a user to sign a message on behalf of a group. A verifier is convinced that the real signer is a member of the group, but cannot explicitly identify the real signer. Considering the anonymity of a cryptocash system, a ring signature is obviously more suitable than a standard signature. But there is a cost: the size of the signature and the computational complexity are inherently larger than those of a standard signature. A traditional ring signature scheme usually features a signature size of $O(N)$, where the ring has N participants. To construct a ring signature of $O(\log N)$ or $O(1)$ size was an open problem in this field. Recently, Groth and Kohlweiss proposed a commitment-based scheme with logarithmic signature size [12].

However, a cryptocash system which naively replace a standard signature with a ring signature suffers from the double spending attack. To address this problem, it is necessary for the public to determine ring signatures generated by the same key pair. The traceable ring signature [9], which is modified as a "one-time signature" and adopted in CryptoNote and Monero *etc.*, provides the ability to trace the verifying and signing key pair which have been used for signing different messages. In general, a linkable ring signature [17], which is a variant of the linkable spontaneous anonymous group signature [16], is sufficient enough for cryptocash systems to determine double spending. Even though signatures of these schemes are of size $O(N)$, CryptoNote and Monero do provide better privacy than Bitcoin.

Most cryptocash systems are based on classic cryptographic schemes. The security of these schemes is based on hard mathematical problems, such as the factorization and discrete logarithm problem (DLP). However, researchers have proved that a quantum computer is able to solve these problems efficiently so that schemes based on these problems are not secure under a quantum computing model. One solution is to build schemes on mathematical problems that remain even hard for quantum computers. Lattice problems are widely believed as suitable choices to build quantum resistant cryptographic schemes since Ajtai brought it into the region of cryptology [2]. Some post-quantum signature schemes have been proposed recently [18, 7, 10]. Relying on these schemes, it is easy to obtain a post-quantum cryptocash system by replacing the ECDSA signature scheme in Bitcoin. However, the resulting cryptocash system is simply like Bitcoin in which the transactions are still linkable. Even though there are several lattice-based ring signatures [6, 36, 37] including the one with logarithmic size [15], none of them has the linkable or traceable property which is vital to prevent double spending.

In this paper, we aim at designing an anonymous post-quantum cryptocash (APQC) system. In order to achieve this goal, we propose a linkable ring signature (LRS) based on ideal lattices. The size of a signature in this scheme is $O(\log N)$, where N is the cardinality of the ring. The framework of our cryptocash system follows that of CryptoNote [31], and the lattice-based signature scheme is inspired by the work of Groth and Kohlweiss [12] with some modifications.

The paper is organized as follows: in Sect. 2, we introduce notations and concepts applied in our work. The model of the ring signature based cryptosystem is described in Sect. 3. Section 4 involves the concrete construction of the ideal-lattice-based linkable ring signature. We design the standard transaction of our cryptocash system in Sect. 5; Section 6 is the conclusion and our future works.

2 Preliminaries

2.1 Notations

We use $\mathbb{Z}$, $\mathbb{R}$ to denote the set of all integers and the set of all reals, respectively. For any $x \in \mathbb{R}$, $\lceil x \rceil$ denotes the smallest integer that is not smaller than x. Column vectors are named by lower-case bold letters (e.g., $\mathbf{x}$) and matrices by upper-case bold letters (e.g., $\mathbf{X}$). The i^{th} entry of a vector $\mathbf{x}$ is denoted by x_i. For a vector $\mathbf{x}$, $\|\mathbf{x}\|_p$ represents its ℓ_p norm, and p is omitted if $p = \infty$. The norm of a polynomial is defined similarly by regarding it as a vector. If $\mathbf{x}$ is a vector of polynomial, then $\|\mathbf{x}\| = \max_{x_i \in \mathbf{x}}\{x_i\}$. For a matrix $\mathbf{X}$, define $\|\mathbf{X}\|_p = \max_{\mathbf{y} \in \mathbf{X}}(\|\mathbf{y}\|_p)$. If $a \in R$ and $\mathbf{X}$ is a matrix with entries in ring R, $a\mathbf{X}$ denotes the scalar multiplication. Let x be any symbol, $\{x_i\}_{i=1}^n$ denotes the set $\{x_1, \ldots, x_n\}$. $\mathbf{I}$ is the identity matrix whose dimension is known from the context. For an integer i, i_j symbolizes the j^{th} bit of i. $\delta_{i\ell}$ is Kronecker's delta, i.e., $\delta_{\ell\ell} = 1$ and $\delta_{i\ell} = 0$ for $i \neq \ell$. For two strings x_1 and x_2, $x_1\|x_2$ denotes the concatenation of them.

2.2 Lattices and Hard problems

A lattice $\Lambda = \mathcal{L}(\mathbf{B})$ with dimension m and rank n is a subgroup of the linear space $\mathbb{R}^m$. Every element in Λ can be represented as an integral combination of its basis $\mathbf{B} \in \mathbb{R}^{m\times n}$. In our work, we will focus on a specific class of lattices, called ideal lattices, which can be described as ideals of certain polynomial rings.

Definition 1 ([19]). *An ideal lattice is an integer lattice $\mathcal{L}(\mathbf{B}) \subseteq \mathbb{Z}^n$ such that $\mathcal{L}(\mathbf{B}) = \{g \bmod f : g \in \mathcal{I}\}$ for some monic polynomial f of degree n and ideal $\mathcal{I} \in \mathbb{Z}[x]/\langle f\rangle$.*

The quotient ring $\mathbb{Z}[x]/\langle f\rangle$ is additively isomorphic to the integer lattice $\mathbb{Z}^n$.

To extend the hash function family in previous works [2, 5, 22], Micciancio defined the generalized knapsack function family [20, 21].

Definition 2 ([21]). *For any ring R, subset $D \subset R$ and integer $m \geq 1$, the generalized knapsack function family $\mathcal{K}(R, D, m) = \{f_{\mathbf{a}} : D^m \to R\}_{\mathbf{a}\in R^m}$ is defined* *by*

$$f_{\mathbf{a}}(\mathbf{x}) = \sum_{i=1}^{m} x_i \cdot a_i,$$

for all $\mathbf{a} \in R^m$ *and* $\mathbf{x} \in D^m$*, where* $\sum_i x_i \cdot a_i$ *is computed using the ring addition and multiplication operations.*

In the process of proving the one-way property of their function family, Micciancio showed that for a special case of the generalized knapsack function family, the distribution of $f_{\mathbf{a}}(\mathbf{x})$ is uniform and independent from $\mathbf{a}$.

Theorem 1 ([21]). *For any finite field* $\mathbb{F}$*, subset* $S \subset \mathbb{F}$*, and integers* n, m*, the hash function family* $\mathcal{K}(\mathbb{F}^n, S^n, m)$ *is* ϵ*-regular for*

$$\epsilon = \frac{1}{2}\sqrt{(1 + |\mathbb{F}|/|S|^m)^n - 1}.$$

In particular, for any $q = n^{O(1)}$*,* $|S| = n^{\Omega(1)}$ *and* $m = \omega(1)$*, the function ensemble* $\mathcal{K}(\mathbb{F}_q^n, S^n, m)$ *is almost regular.*

Here, ϵ-regular means that the statistical distance between uniform distribution $U((\mathbb{F}^n)^m, \mathbb{F}^n)$ and $\{(\mathbf{a}, f_{\mathbf{a}}(\mathbf{x})) : \mathbf{a} \leftarrow U((\mathbb{F}^n)^m), \mathbf{x} \leftarrow U((S^n)^m)\}$ is at most ϵ. Note that $\mathbb{F}^n$ can be instantiated with the quotient ring $R = \mathbb{F}[x]/\langle f \rangle$, where $f \in \mathbb{F}[x]$ is a monic polynomial of degree n. S^n can be regarded as the subset of R.

Sometimes, the one-way property of a function is not sufficient enough to design a cryptographic protocol. Lyubashevsky and Micciancio proved that finding a collision in some instance of the generalized knapsacks function family is as hard as solving the worst-case problem in a certain lattice [19].

Definition 3 (Collision Problem). *For any function family* $\mathcal{K}(R, D, m)$*, define the collision problem* $Col_{\mathcal{K}}(h_{\mathbf{a}})$ *as follows: given a function* $h_{\mathbf{a}} \in \mathcal{K}$*, find* $\mathbf{b}, \mathbf{c} \in D^m$ *such that* $\mathbf{b} \neq \mathbf{c}$ *and* $h_{\mathbf{a}}(\mathbf{b}) = h_{\mathbf{a}}(\mathbf{c})$*.*

If there is no polynomial time algorithm that can solve $\mathrm{Col}_{\mathcal{K}}$ with non-negligible probability when given a function $h_{\mathbf{a}}$ which is distributed uniformly at random in $\mathcal{K}$, then $\mathcal{K}$ is collision resistant.

The expansion factor is a parameter proposed to quantify the quality of modulus f in the ideal lattice [19]. The expansion factor of f is defined as

$$\mathrm{EF}(f, k) = \max_{g \in \mathbb{Z}[x], \deg(g) \leq k(\deg(f)-1)} \|g\|_f / \|g\|_\infty$$

where $\|g\|_f$ is short for $\|g \bmod f\|_\infty$. Moreover, $\mathrm{EF}(x^n + 1, k) \leq k$.

The generalized knapsacks function family $\mathcal{K}(R, D, m)$ considered in their paper are instantiated as follows. Let $R = \mathbb{Z}[x]_q/\langle f \rangle$ be a ring for some integer q, where $f \in \mathbb{Z}[x]$ is a monic, irreducible polynomial of degree n with expansion factor $\mathrm{EF}(f, 3) \leq \varepsilon$. Let $D = \{g \in R : \|g\| \leq \beta\}$ for some positive integer β.

Theorem 2 ([21]). *Let* $\mathcal{K}(R, D, m)$ *be a Generalized Compact Knapsacks function family as above. If* $m \geq \frac{\log q}{\log 2\beta}$ *and* $q > 2\varepsilon\beta m n^{1.5} \log n$*. Then, for* $\gamma = 8\varepsilon^2 \beta m n \log^2 n$*, there is a polynomial time reduction from* f*-*$SPP_\gamma(\mathcal{I})$ *for any ideal* $\mathcal{I} \in R$ *to* $Col_{\mathcal{K}}(h)$ *where* h *is chosen uniformly at random from* $\mathcal{K}$*.*

If we denote by $\mathcal{I}(f)$ the set of lattices that are isomorphic (as additive groups) to ideals of $\mathbb{Z}[x]/\langle f\rangle$ where f is monic, then there is a straightforward reduction from $\mathcal{I}(f)$-SVP$_\gamma$ to f-SPP$_\gamma$, and the vise versa. The $\mathcal{I}(f)$-SVP$_\gamma$ with a polynomial approximating factor γ is widely believed to be intractable against even a quantum computer.

2.3 The Public-key Encryption on Ideal Lattices

The cryptosystem we described here was proposed by Stehlé *et al.* [34]. The ideal-lattice-based encryption scheme is formalized as a tuple of efficient procedures $\mathcal{ES}$=(**Setup**, **KGen**, **Enc**, **Dec**).

Setup(1^n): n is the security parameter. Fix $f(X) = X^n + 1 \in \mathbb{Z}[X]$ and $q = \mathrm{poly}(n)$ a prime satisfying $q \equiv 3 \bmod 8$. Set $\sigma = 1$, $r = 1 + \log_3 q$, and $m = (\lceil \log q\rceil + 1)\sigma + r$. Let $R = \mathbb{Z}_q[x]/\langle f\rangle$. All the parameters generated in this procedure are published as the global parameter *pp*.

KGen(pp): On input global parameter *pp*, it runs the trapdoor generation algorithm **Id-Trap** to get a trapdoor function $h_{\mathbf{g}} : \mathbb{Z}_q^n \times \mathbb{Z}_q^{mn} \to \mathbb{Z}_q^{mn}$ and a trapdoor S, where $\mathbf{g}$ is the function index. The first component of the domain of $h_{\mathbf{g}}$ can be viewed as a subset of $\mathbb{Z}_2^{\ell_I}$ for $\ell_I = O(n \log q)$. Generate $\mathbf{r} \in \mathbb{Z}_2^{\ell_I + \ell_\mu}$ uniformly and define the Toeplitz matrix $M_{\mathrm{GL}} \in \mathbb{Z}_2^{\ell_\mu \times \ell_i}$ whose i^{th} row is $(r_i, \ldots, r_{\ell_I + i - 1})$. It outputs the public key $epk = (\mathbf{g}, \mathbf{r})$ and the secret key $esk = S$.

Enc(pp, epk, μ): Given ℓ_μ bit message μ with $\ell_\mu = n/\log n$ and public key $epk = (\mathbf{g}, \mathbf{r})$, sample $(\mathbf{s}, \mathbf{e})$ with $\mathbf{s} \in \mathbb{Z}_q^n$ uniform and $\mathbf{e}$ sampled from $\bar{\Psi}_{\alpha q}$, where $\bar{\Psi}_{\alpha q}$ is the reduction modulo q of the standard Gaussian distribution with parameter αq. It then evaluates $C_1 = h_{\mathbf{g}}(\mathbf{s}, \mathbf{e})$ and computes $C_2 = \mu \oplus (M_{\mathrm{GL}} \cdot \mathbf{s})$, where the product $M_{\mathrm{GL}} \cdot \mathbf{s}$ is computed over $\mathbb{Z}_2$, and $\mathbf{s}$ is viewed as a string over $\mathbb{Z}_2^{\ell_I}$. Return the ciphertext $C = (C_1, C_2)$.

Dec(pp, esk, C): Given cyphertext $C = (C_1, C_2)$ and secret key $esk = (S, \mathbf{r})$, invert C_1 to compute $(\mathbf{s}, \mathbf{e})$ such that $h_{\mathbf{g}}(\mathbf{s}, \mathbf{e}) = C_1$, and return message $\mu = C_2 \oplus (M_{\mathrm{GL}} \cdot \mathbf{s})$.

To see the details of the trapdoor generation algorithm **Id-Trap** and the one-way trapdoor function family $\{h_{\mathbf{g}} : \mathbb{Z}_q^n \times \mathbb{Z}_q^{mn} \to \mathbb{Z}_q^{mn}\}_{g \in (\mathbb{Z}_q[x]/\langle f\rangle)^m}$, we refer to the literature [34] in which Stehlé *et al.* also proved that the above encryption scheme is IND-CPA secure if the Ideal-LWE$^f_{m,q;\Psi_{\alpha q}}$ problem is hard.

The notion of key privacy is formally defined by Bellare *et al.* [4]. It requires that the receiver of a ciphertext is anonymous from the point of view of the adversary. Fortunately, we can deduce from the observation 1 of [13] that the aforementioned encryption scheme $\mathcal{ES}$ is of key privacy

3 Anonymous Cryptographic Currency Model Based on Linkable Ring Signatures

Cryptocash system based on linkable ring signatures emerged after researchers found that Bitcoin was not fully anonymous and untraceable. CrytoNote and

Monero are two typical instances. We describe here the properties of an anonymous cryptocash system and state the techniques [31] to construct such a system.

In a cryptocash system, there are three parties: a sender, who owns a coin and decides to spend it, a receiver, who is the destination that a coin is delivered to, and a public ledger where all transactions are recorded. An anonymous cryptocash system should satisfy the following properties:

- **Untraceability**: If Tx is a transaction from sender A to receiver B, and Tx has been recorded in the public ledger, no one else can determine the sender with probability significantly larger than $1/N$ by accessing the transcript of Tx, where N is the number of possible senders in a related input of the Tx. Moreover, even receiver B cannot prove that A is the true sender of Tx.
- **Unlinkability**: If Tx_1 is a transaction from sender A to receiver C, Tx_2 is another transaction from sender B to receiver C, and Tx_1, Tx_2 have been recorded in the public ledger, then for any subsequent transactions in the public ledger, no one else can use them to link the outputs of the two transactions to a single user, even for senders A and B.
- **Detecting Double Spending**: If Tx_1 is a transaction which describes that coin c has been sent from sender A to receiver B, and Tx_1 has been recorded in the public ledger, every user of the system could detect another transaction Tx_2 that describes the same coin c. Furthermore, Tx_2 will never be accepted and recorded in the public ledger.

To design a cryptocash protocol which provides all the above properties, the CryptoNote and Monero suggested to adopt the modification of the traceable ring signature [9], which generates a one-time signature on behalf of a temporal group. Since it is a one-time signature with an explicit identification tag about the signing key, it could prevent a coin being double-spent. Besides, since it is a ring signature where the identity of the real signer is hidden within a set of possible signers, it guarantees untraceability. In addition, ring signature supports unlinkability since the inputs in a transaction may be brought from outputs of transactions belonging to other users.

To employ a linkable ring signature in a cryptocash system, the receiver should produce a one-time key pair for each transaction. A sender could obtain the public key of the receiver for the transaction and build a transaction with an output script containing that key's information. The drawback of this trivial method is that a receiver has to maintain a lot of one-time keys. Furthermore, a sender has to contact each receiver for their fresh one-time public key when the sender builds a transaction. Alternatively, CryptoNote suggests another method which enables a receiver to store only a single key pair. A sender could produce a random value to generate a one-time public key for the receiver based on this single public key. The one-time public key is referred to as the destination address. This is a convenient design at the cost of a slightly weakened unlinkability. Specifically, if a user has a single key, a sender could always identify a receiver from the sender's transaction by its random value of the transaction. If two senders collude, and they have sent coins to the same receiver, they could identify the same receiver while the trivial method avoids this. And if a later transaction includes the two senders' outputs at the same time, with a higher probability, the later transaction is made by the receiver. Note that a receiver could still produce another key pair at will as in the Bitcoin system to avoid the small problem.

Finally, let us observe a standard transaction in a linkable ring signature based cryptocash system. In such a system, the value of a coin is bound with a destination address. Suppose A and B are two users in the system. B has a single key pair (pk_B, sk_B). A has the private key sk_1 of a destination address vk_1, which represents a coin, say c, which has been sent to A previously. If A decides to send c to B, he generates a destination address vk_2 and an auxiliary input aux for B; he then chooses a number of transactions from the public ledger such that the delivered value of coin is equivalent to c; he extracts the destination addresses of those transactions and assembles them with vk_1 to form a ring L; he runs a ring signature algorithm to sign transaction Tx, which involves information about (c, aux, vk_2, L), with signing key sk_1 and broadcasts the transaction; If the signature generated by sk_1 is not linkable to any transaction on the ledger, the public ledger will accept this transaction and record it; B uses its private key sk_B to check every passing transaction to determine if transaction Tx is for B and recovers the signing key sk_2 corresponding to vk_2. With sk_2, user B can spend c by signing another transaction. However, even A does not know when and where B spends it due to the functionality of the linkable ring signature.

It is obvious that linkable ring signature is vital for an anonymous cryptocash system. We next detail the lattice-based version of a linkable ring signature.

4 Linkable Ring Signature Based on Ideal-Lattices

The strong similarity in the construction between a lattice-based signature and DLP-based one (see Lyubashevsky's signature [18] and the Schnorr's signature [32, 33]) implies that the latter can help us to design the lattice-based counterparts of DLP-based schemes, e.g., using the work in [17] or [18], it can easily obtain a linkable ring signature based on lattices with signature size of $O(N)$, where N is the number of participants of the ring. However, such a construction is not efficient enough for a practical cryptocash system. In this section, we aim at presenting a linkable ring signature of size $O(\log N)$ using the idea in [12]. We start this section with a brief recall on their work.

4.1 A Brief Recall

In [12], Groth and Kohlweiss proposed an efficient Sigma-protocol, which can be used as an ad-hoc group identification scheme. Their ring signature scheme is a direct transformation of the identification scheme with the Fiat-Shamir heuristic [8]. As the transmission of the identification scheme involves only logarithmic number of commitments, the resulting ring signature scheme is of size $O(\log N)$.

Their work started from homomorphic commitments scheme such as Pedersen commitment $(\mathrm{com}(m; r) = h^m g^r)$. The first step is to design a Sigma-protocol Σ_1 to prove in zero-knowledge that such a commitment is opened to 0 or 1. Once the subroutine Σ_1 is established, to design a ad-hoc group identification scheme is to construct a Sigma-protocol Σ_2 to show in zero-knowledge that one of N commitments is opened to 0. Here, a commitment to 0 is the public key of a user and the randomness used is the corresponding secret key. If the ℓ^{th} user of the ad-hoc group $\{\text{user}_0, \cdots \text{user}_{N-1}\}$ wants to identify himself secretly, Σ_2 first commits the integer ℓ bit by bit and runs Σ_1 to prove in zero-knowledge that those $\log N$ commitments are opened to 0 or 1. Then Σ_2 proves in zero-knowledge that the ℓ^{th} user can open the ℓ^{th} public key (a commitment to 0) to 0, with the help of the intermediate parameters used in the foregoing Σ_1's. By replacing the challenge message with the hash value of all initial messages in Σ_2, we obtain a non-interactive zero-knowledge proof system which can be regarded as a ring signature. For the details of the generic construction of Σ_2, we refer the reader to the literature [12].

It is worth mentioning that the underlying homomorphic commitment is the corner stone of both the construction and security proof of the foregoing ring signature. As a counterpart of their work, our scheme also contains a lattice-based commitment (e.g., $\text{com}(\mathbf{S};\mathbf{X}) = \mathbf{HS}+\mathbf{GX}$). The details of the commitment scheme is left to Sect. 4.3.

4.2 Our Construction

To construct an $O(\log N)$ ring signature, Groth and Kohlweiss proposed a technique to compute the coefficients of a polynomial in the indeterminate x over the finite filed $\mathbb{Z}_q$ in advance, where x is a hash value computed later [12]. We extend their method to handle the polynomial with coefficients belonging to a ring of square matrices. The major difference is that the multiplication of matrices is not commutative. This is the reason why we restrict x in our scheme to be a 1×1 matrix. Since the scalar multiplication is commutative, we have the following result.

Given matrices $\mathbf{B}_j$, set $\mathbf{W}_j = \ell_j x\mathbf{I} + \mathbf{B}_j$, for $\ell_j \in \{0,1\}$. Let $\mathbf{W}_{j,1} = \mathbf{W}_j = \ell_j x\mathbf{I} + \mathbf{B}_j = \delta_{1\ell_j} x\mathbf{I} + \mathbf{B}_j$ and $\mathbf{W}_{j,0} = x\mathbf{I} - \mathbf{W}_j = (1-\ell_j)x\mathbf{I} - \mathbf{B}_j = \delta_{0\ell_j} x\mathbf{I} - \mathbf{B}_j$. Then for each i, the product $\prod_{j=1}^{n} \mathbf{W}_{j,i_j}$ is a polynomial of the form

$$P_i(x) = \prod_{j=1}^{n}(\delta_{i_j\ell_j} x\mathbf{I}) + \sum_{k=0}^{M-1} \mathbf{P}_{i,k} x^k = \delta_{i\ell} x^n \mathbf{I} + \sum_{k=0}^{M-1} \mathbf{P}_{i,k} x^k.$$

Hence, $\mathbf{P}_{i,k}$ is the coefficient of the k^{th} degree term in the polynomial, and can be efficiently computed when $\{\mathbf{B}_j\}_{j=1}^{M}$ and ℓ are given.

The linkable ring signature scheme consists of a tuple of efficient procedures $\mathcal{LRS}$ = (**Setup**, **KGen**, **Sign**, **Vfy**, **Link**). Let N be the maximum size of the ring, $M = \lceil \log N \rceil$, and n be the security parameter. The details of those procedures are shown as follows:

Setup$(1^n, N)$: On input n and N, the procedure initiates a hash function introduced in [18] as a random oracle $\mathcal{H} : \{0,1\}^* \to \{\mathbf{v} : \mathbf{v} \in \{-1,0,1\}^n, \|\mathbf{v}\|_1 \leq p\}$, such that $2^p \cdot \binom{n}{p} \geq 2^{100}$. It sets $\varepsilon = 3$, $t = \Theta(n)$, $\beta \geq \frac{t(p^{M+1}-1)}{(p-1)}$, $m = \Theta(\log n)$.

Pick a prime q such that $(2\beta)^m > q > 2\varepsilon\beta m n^{1.5} \log n$. All operations in this system are done in $R = \mathbb{Z}_q[X]/\langle f \rangle$, for $f = X^n + 1 \in \mathbb{Z}[X]$. Let $D = \{g \in R : \|g\| \leq t\}$. Relying on those parameters, this procedure samples matrices $\mathbf{G}, \mathbf{H} \in R^{1\times m}$ uniformly at random. Finally it outputs $pp = (n, m, \mathbf{G}, \mathbf{H}, \mathcal{H}, q, t, N)$ as the global parameters.

KGen(pp): For the i^{th} user, this procedure randomly chooses $\mathbf{X}_i \leftarrow D^{m\times m}$ and computes $\mathbf{Y}_i = \mathbf{G}\mathbf{X}_i$. The i^{th} user's verifying key is $vk_i = \mathbf{Y}_i$ and the singing key is $sk_i = \mathbf{X}_i$.

Sign(pp, sk_ℓ, μ, L): Without loss of generality, let $L = (\mathbf{Y}_0, \mathbf{Y}_1, \ldots, \mathbf{Y}_{N-1})$ be the ensemble of a ring with the largest size. On input a message μ, the ℓ^{th} user's signature on behalf of L is generated as follows

- Compute $\mathbf{R}_\ell = \mathbf{H}\mathbf{X}_\ell$.
- For j from 1 to M,
  - sample $\mathbf{K}_j, \mathbf{C}_j, \mathbf{D}_j, \mathbf{E}_k \leftarrow D^{m\times m}$,
  - if $\ell_j = 0$, randomly pick $\mathbf{B}_j \leftarrow D^{m\times m}$,
    else if $\ell_j = 1$, draw $\mathbf{B}_j \in \{g \in R : \|g\| \leq t-1\}$ randomly,
  - compute $\mathbf{V}_{\ell_j} = \mathbf{H}(\ell_j \mathbf{I}) + \mathbf{G}\mathbf{K}_j$, and $\mathbf{V}_{a_j} = \mathbf{H}\mathbf{B}_j + \mathbf{G}\mathbf{C}_j$,
  - compute $\mathbf{V}_{b_j} = \mathbf{H}(\ell_j \mathbf{B}_j) + \mathbf{G}\mathbf{D}_j$,
  - compute $\mathbf{V}_{d_k} = (\sum_{i=0}^{N-1} \mathbf{Y}_i \mathbf{P}_{i,k}) + \mathbf{G}\mathbf{E}_k$, where $k = j-1$,
  - compute $\mathbf{V}'_{d_k} = \mathbf{H}\mathbf{E}_k$, where $k = j-1$.
- Let set $S_1 = \{\mathbf{V}_{\ell_j}, \mathbf{V}_{a_j}, \mathbf{V}_{b_j}, \mathbf{V}_{d_{j-1}}, \mathbf{V}'_{d_{j-1}}\}_{j=1}^{M}$ and then compute hash value $x = \mathcal{H}(pp, \mu, L, S_1, \mathbf{R}_\ell)$.
- For j from 1 to M, compute
  1. $\mathbf{W}_j = \ell_j x \mathbf{I} + \mathbf{B}_j$,
  2. $\mathbf{Z}_{a_j} = \mathbf{K}_j(x\mathbf{I}) + \mathbf{C}_j$,
  3. $\mathbf{Z}_{b_j} = \mathbf{K}_j(x\mathbf{I} - \mathbf{W}_j) + \mathbf{D}_j$,
  4. $\mathbf{Z}_d = \mathbf{X}_\ell(x^M \mathbf{I}) - \sum_{k=0}^{M-1} \mathbf{E}_k x^k$.
- Let $S_2 = \{\mathbf{W}_j, \mathbf{Z}_{a_j}, \mathbf{Z}_{b_j}\}_{j=1}^{M}$. Publish $\sigma = \{S_1, S_2, \mathbf{Z}_d, \mathbf{R}_\ell, L\}$ as the signature of the ring L on the message μ.

Vfy(pp, μ, σ):

1. Compute hash value $x = \mathcal{H}(pp, \mu, L, S_1, \mathbf{R}_\ell)$.
2. For j from 1 to M, consider the following inequalities
   - $\|\mathbf{W}_j\| \leq t$,
   - $\|\mathbf{Z}_{a_j}\| \leq (p+1)t$,
   - $\|\mathbf{Z}_{b_j}\| \leq tp + t^2 nm + t$,
   - $\|\mathbf{Z}_d\| \leq \frac{t(p^{M+1}-1)}{p-1}$. If any of them does not hold, output 0 and abort.
3. For j from 1 to M, consider following equations
   - $\mathbf{V}_{\ell_j}(x\mathbf{I}) + \mathbf{V}_{a_j} = \mathbf{H}\mathbf{W}_j + \mathbf{G}\mathbf{Z}_{a_j}$,
   - $\mathbf{V}_{\ell_j}(x\mathbf{I} - \mathbf{W}_j) + \mathbf{V}_{b_j} = \mathbf{G}\mathbf{Z}_{b_j}$.

   If any of the aforementioned equations does not hold, output 0 and abort.

4. If the equation $\mathbf{R}_\ell(x^M\mathbf{I}) + \sum_{k=0}^{M-1} \mathbf{V}'_{d_k}(-x^k) = \mathbf{H}\mathbf{Z}_d$ does not hold, output 0 and abort.
5. Inspect whether

$$\sum_{i=0}^{N-1}(\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}) + \sum_{k=0}^{M-1} \mathbf{V}_{d_k}(-x^k) = \mathbf{G}\mathbf{Z}_d$$

holds. If not, output 0, otherwise output 1 (accept).

Link(pp, σ_1, σ_2): For two signatures $\sigma_1 = (\dots, \mathbf{R}_1, L_1)$ and $\sigma_2 = (\dots, \mathbf{R}_2, L_2)$, if $\mathbf{R}_1 = \mathbf{R}_2$, return 1 (linked) for concluding that they are generated by the same signer; otherwise, return 0 (unlinked).

Correctness: To see that the signature generated by **Sign** procedure always passes the **Vfy** procedure, we first observe the four equations in the **Vfy** procedure. The equations in step 3 are to prove in zero-knowledge that the signer is the ℓ^{th} user. The correctness of those equations is shown directly through a simple deduction. The equation in step 4 is to prove that the parameter for linking is correct. For a valid signature, it holds since

$$\begin{aligned} &\mathbf{R}_\ell(x^M\mathbf{I}) + \textstyle\sum_{k=0}^{M-1} \mathbf{V}'_{d_k}(-x^k) \\ &= \mathbf{H}\mathbf{X}_\ell(x^M\mathbf{I}) + \textstyle\sum_{k=0}^{M-1} \mathbf{H}\mathbf{E}_k(-x^k) \\ &= \mathbf{H}(\mathbf{X}_\ell(x^M\mathbf{I}) - \textstyle\sum_{k=0}^{M-1} \mathbf{E}_k x^k) = \mathbf{H}\mathbf{Z}_d \end{aligned} .$$

The equation in step 5 is to prove in zero-knowledge that the anonymous signer holds the secret key of the ℓ^{th} user. To see the correctness of the last equation, note that $\prod_{j=1}^{M} \mathbf{W}_{j,i_j}$ is a polynomial in the indeterminate x. Subsequently, $\mathbf{Y}_\ell \prod_{j=1}^{M} \mathbf{W}_{j,i_j}$ yields a polynomial of degree n, while the other $\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}$, $i \neq \ell$ leads to polynomials of degree $n-1$. Formally, we have

$$\begin{aligned} &\textstyle\sum_{i=0}^{N-1}(\mathbf{Y}_i \prod_{j=1}^{M} \mathbf{W}_{j,i_j}) + \sum_{k=0}^{M-1} \mathbf{V}_{d_k}(-x^k) \\ &= \textstyle\sum_{i=0}^{N-1} \mathbf{Y}_i(\delta_{i\ell}x^M\mathbf{I} + \sum_{k=0}^{M-1} \mathbf{P}_{i,k}x^k) + \sum_{k=0}^{M-1}((\sum_{i=0}^{N-1} \mathbf{Y}_i\mathbf{P}_{i,k}) + \mathbf{G}\mathbf{E}_k)(-x^k) \\ &= \textstyle\sum_{i=0}^{N-1} \sum_{k=0}^{M-1}(\mathbf{Y}_i\mathbf{P}_{i,k}x^k - \mathbf{Y}_i\mathbf{P}_{i,k}x^k) + \mathbf{Y}_\ell(\delta_{\ell\ell}x^M\mathbf{I}) + \sum_{k=0}^{M-1} \mathbf{G}\mathbf{E}_k(-x^k) \\ &= \mathbf{G}(\mathbf{X}_\ell(x^M\mathbf{I}) - \textstyle\sum_{k=0}^{M-1} \mathbf{E}_k x^k) \\ &= \mathbf{G}\mathbf{Z}_d \end{aligned} .$$

It remains to show that $\{\mathbf{W}_j, \mathbf{Z}_{a_j}, \mathbf{Z}_{b_j}\}_{j=1}^M$, and $\mathbf{Z}_d$ are short enough to pass step 2 of the **Vfy** procedure.

We note that for polynomials $a, b \in R$, the norm of their product is bounded by $\|a\| \cdot \|b\| \cdot n$. For $a \in R$ and $b \in \{v : v \in \{-1, 0, 1\}^n, \|v\|_1 \leq p\}$ the norm of $a \cdot b$ is not larger than $\|a\| \cdot \|b\| \cdot p$. With the help of above two facts and the triangle inequality, the correctness of the norm about those matrices can be validated easily. For example, $\|\mathbf{Z}_{b_j}\| = \|\mathbf{K}_j(x\mathbf{I} - \mathbf{W}_j) + \mathbf{D}_j\| \leq \|\mathbf{K}_j x\mathbf{I}\| + \|\mathbf{K}_j(-\mathbf{W}_j)\| + \|\mathbf{D}_j\| \leq tp + t^2nm + t$ and $\|\mathbf{Z}_d\| \leq \|\mathbf{X}_\ell(x^M\mathbf{I})\| + \|\sum_{k=0}^{M-1}\mathbf{E}_k x^p\| \leq tp^M + \|\mathbf{E}_0 x^0\| + \|\mathbf{E}_1 x^1\| + \cdots + \|\mathbf{E}_{M-1}x^{M-1}\| \leq tp^M + t + tp + \cdots + tp^{M-1} = \frac{t(p^{M+1}-1)}{p-1}$.

Even though the foregoing linkable ring signature is designed over ideal lattices, a classic edition of this signature can be built by instead using any cyclic group as long as its underlying DLP is hard. We propose a linkable ring signature based on the ECDLP, and discuss how to implement this signature with ECC in the full version of this paper [38]. It is easy to see that the ring confidential transactions [26] (later strengthened by Sun *et al.* [35]), which are adopted in Monero to hide the amount of a transaction, are trivially achievable with the ECDLP-based signature.

4.3 Security Proof

Groth and Kohlweiss have proved that the generic construction of their ring signature is secure in the random oracle model, if its underlying commitment scheme is perfectly hiding and computationally binding [12]. Since our linkable ring signature is designed over the framework of their generic construction, to prove the security of our scheme is sufficient to prove the binding and hiding properties of the commitment scheme applied in our signature scheme.

Theorem 3 ([12]). *The generic construction of the ring signature scheme in [12] is perfect anonymity if the underlying commitment scheme is perfectly hiding. It is unforgeable in the random oracle model if the commitment scheme is perfectly hiding and computationally binding.*

A non-interactive commitment scheme allows a sender to construct a commitment to a value. The sender may later open the commitment and reveal the value so that the receiver can verify the opening and check if it was the value that was committed at the beginning. A commitment scheme is said to be hiding, only if it reveals nothing about the committed value. The binding property ensures that a sender cannot open the commitment to two different values.

We now proceed to introduce the details of the underlying commitment in our $\mathcal{LRS}$. The non-interactive commitment scheme adopted in our $\mathcal{LRS}$ consists of a pair of PPT algorithms $\mathcal{CMT}$=(**Gen, Com**).

Gen(1^n)**:** The setup algorithm is tightly associated with that of $\mathcal{LRS}$. It runs $\mathcal{LRS}.\mathbf{Setup}(1^n)$ to get the global parameters $\mathcal{LRS}.pp$ of the signature scheme and picks $pp = (n, m, \mathbf{G}, \mathbf{H}, q, t)$ out of $\mathcal{LRS}.pp$ to be the global parameters of the commitment scheme. The value to be committed and the randomness to be chosen are elements in $D^{m\times m}$.

Com$(pp, \mathbf{S})$**:** If a sender wants to construct a commitment $\mathbf{C}$ to the matrix $\mathbf{S} \in D^{m\times m}$, it uniformly samples $\mathbf{X} \leftarrow D^{m\times m}$ and computes $\mathbf{C} = \mathbf{HS} + \mathbf{GX}$. The commitment $\mathbf{C}$ can later be opened by unveiling the short $\mathbf{S}$ and $\mathbf{X}$, where $\|\mathbf{S}\|, \|\mathbf{X}\| \leq t$.

The correctness of the foregoing commitment scheme $\mathcal{CMT}$ is obvious. It remains to prove that $\mathcal{CMT}$ is hiding and biding.

Theorem 4 (Binding and Hiding). *For any committed matrix* $\mathbf{S} \leftarrow D^{m\times m}$ *and any random matrix* $\mathbf{X} \leftarrow D^{m\times m}$*, the commitment* $\mathbf{C} = \mathbf{HS} + \mathbf{GX}$ *reveals nothing about* $\mathbf{S}$*. Moreover, the sender can't open the commitment* $\mathbf{C}$ *to* $\mathbf{S}' \neq \mathbf{S}$*, if the collision problem* $Col_{\mathcal{K}}$ *defined in Definition 3 is hard.*

Proof. Let $\mathcal{K}(R, D, m)$ be the generalized knapsacks functions family for $R = \mathbb{Z}_q[X]/\langle f\rangle$, $D = \{g \in R : \|g\| \leq t\}$, $f = X^n + 1$. Given a matrix $\mathbf{G} \in R^{1\times m}$ sampled uniformly at random, we obtain a uniformly random instance of the function family $h_{\mathbf{G}} : D^m \to R$. Such R, D, m are used as the system parameters in our linkable ring signature and commitment schemes. Let $\mathbf{X}_i$ symbolize the i^{th} column of the matrix $\mathbf{X} \in D^{m\times m}$ that is sampled uniformly at random. Note that R can be regarded as $\mathbb{Z}_q^n$ and $\mathbb{Z}_q$ is a finite field. According to Theorem 1, the distribution of $f_{\mathbf{G}}(\mathbf{X}_i) = \mathbf{GX}_i$ is almost uniform over $\mathbb{Z}_q^n$ (namely R), since $q \geq 2\varepsilon\beta mn^{1.5}\log n$, $t = \Theta(n)$, $m = \Theta(\log n)$ in our setting. Consequently, $f_{\mathbf{G}}(\mathbf{X}) = \mathbf{GX}$ is almost uniform over $R^{1\times m}$. Note that, this fact is also suitable for the product $\mathbf{HS}$ when they are selected as in our signature and commitment schemes. As a result, $\mathbf{C} = \mathbf{HS} + \mathbf{GX}$ is uniform over $R^{1\times m}$ and hence $\mathbf{C}$ reveals nothing about the committed value $\mathbf{S}$.

We proceed to prove the binding property. Because $(2\beta)^m \geq q$, we have $m > \frac{\log q}{\log 2\beta}$. Depending on Theorem 2, to find a collision in the generalized knapsack function $f_{\mathbf{G}}(\mathbf{X})$ is as hard as to solve the $\mathcal{I}(f)$-SVP$_\gamma$ problem, so is the function $f_{\mathbf{H}}(\mathbf{S})$. Here, $\gamma = 8\varepsilon^2\beta mn\log^2 n$ is a polynomial in security parameter n. It is conjectured that approximating $\mathcal{I}(f)$-SVP$_\gamma$ to within a polynomial factor is a hard problem, although it is not NP-hard [1, 11].

Suppose for the sake of contradiction that the sender can open the commitment $\mathbf{C}$ to $\mathbf{S}, \mathbf{S}' \in D^{m\times m}$ such that $\mathbf{S} \neq \mathbf{S}$, $\|\mathbf{S}\| \leq t$, $\|\mathbf{S}'\| \leq t$ and $\mathbf{HS} + \mathbf{GX} = \mathbf{HS}' + \mathbf{GX}'$. We consider the two possible cases.

Case 1: If $\mathbf{HS} = \mathbf{HS}'$, then $\mathbf{H}(\mathbf{S} - \mathbf{S}') = \mathbf{0}$. Since $\|\mathbf{S} - \mathbf{S}'\| \leq \|\mathbf{S}\| + \|\mathbf{S}'\| \leq 2t < \beta$. $\mathbf{S}$ and $\mathbf{S}'$ are a pair of collisions of the function $f_{\mathbf{H}}(\mathbf{S})$ which yield a contradiction to the hardness assumption introduced in Theorem 2.

Case 2: If $\mathbf{HS} \neq \mathbf{HS}'$, then $\mathbf{X} \neq \mathbf{X}'$ and we have $\mathbf{H}(\mathbf{S} - \mathbf{S}') = \mathbf{G}(\mathbf{X}' - \mathbf{X})$. Similar to the foregoing discussions, $\mathbf{S}$ and $\mathbf{S}'$ yield a contradiction to the fact that $f_{\mathbf{H}}(\mathbf{S})$ is collision resistant.

Depending on the discussions made in **Case 1** and **Case 2**, we have shown that the sender cannot open the commitment $\mathbf{C}$ to different values. □

Since our underlying commitment scheme $\mathcal{CMT}$ is binding and hiding, the anonymity and unforgeability of the linkable ring signature $\mathcal{LRS}$ can be shown according to Theorem 3. For a complete discussion of the security proof, we refer readers to the full version of this paper [38]. Actually, most of the techniques are follows that of [12] and x has a unique multiplicative inverse in R.

The next is to prove that our linkable ring signature is linkable.

Theorem 5 (Linkability). *Our linkable ring signature* $\mathcal{LRS}$ *is linkable. Formally, given a set of signing keys* $\mathbf{SK} = \{\mathbf{X}_0, \ldots, \mathbf{X}_{N-1}\}$*, it is impossible to produce* $N + 1$ *signatures* $\sigma_0, \ldots, \sigma_N$*, such that any two of them can pass the* **Link** *procedure.*

Proof. Suppose toward the contradiction that an adversary can produce $N+1$ valid signatures $\sigma_i = \{S_1^{(i)}, S_2^{(i)}, \mathbf{Z}_d^{(i)}, \mathbf{R}_i, L_i\}$ such that $\mathbf{R}_i$'s are pairwise distinct, for $i \in \{0, 1, \ldots, N\}$. Since $|\mathbf{SK}| = N$, there is at least one $\mathbf{R}_i$ which does not belong to the set $\{\mathbf{HX}_j : \mathbf{X}_j \in \mathbf{SK}\}$. Without loss of generality, consider this event happened in σ_π. As σ_π is a valid signature, from the verification equation we have

$$\mathbf{R}_\pi((x^{(\pi)})^M\mathbf{I}) + \sum_{k=0}^{M-1}(\mathbf{V}_{d_k}^{(\pi)})'(-(x^{(\pi)})^k) = \mathbf{HZ}_d^{(i)} \ , i \in [0, N-1] \quad , \tag{1}$$

where $\mathbf{Z}_d^{(i)}$ is generated by using the knowledge of one of the signing keys in $\mathbf{SK}$. From Equation (1), we can deduce $\mathbf{R}_\pi((x^{(\pi)})^M\mathbf{I}) = \mathbf{HX}_i((x^{(\pi)})^M\mathbf{I})$, $\mathbf{X}_i \in \mathbf{SK}$, because that $\mathbf{Z}_d^{(i)}$ can pass the verification algorithm. Since the component $(x^{(\pi)})^M\mathbf{I}$ is an invertible matrix, we know that $\mathbf{R}_\pi = \mathbf{HX}_i$, $\mathbf{X}_i \in \mathbf{SK}$. This yields a contradiction to the hypothesis that an adversary can produce $N+1$ valid signatures with N signing keys. Consequently, our signature scheme $\mathcal{LRS}$ is linkable. □

5 APQC Based on LRS

In this section, we will introduce how a sender generates the one-time address and how a receiver recovers the corresponding signing key of a transaction. By combining tools introduced here and the linkable ring signature presented in the previous section, we describe the standard transaction of APQC in detail at last.

5.1 Stealth Addresses

In CryptoNote, the author suggests using stealth addresses to protect the privacy of the receiver in all transactions. In this system, each user is associated with fixed public and private keys. When a sender wants to pay coins to a receiver, a one-time address (a verifying key which is called the destination address) is generated for the receiver and broadcasted with the transaction by the sender. The receiver then checks every passing transaction with his private key to identify which transaction belongs to him. Finally he recovers the correlative signing key from the transaction.

To protect the privacy of receivers, we also design a key-generation protocol to produce stealth addresses. The stealth addresses and its corresponding signing key are later used as the verifying and signing keys in the linkable ring signature.

5.2 Key-Generation Protocol

The key-generation protocol is responsible for three purposes. Firstly, it generates the fixed public and private keys for a user that initially joins the cryptocash system. Secondly, if Alice wants to pay coins to Bob, this protocol produces a new one-time destination key for Bob by using random values of Alice and public keys of Bob. Note that the destination key is essentially a verifying key of the linkable ring signature scheme. Thirdly, since Alice broadcasts the transaction labeled with the destination key, the receiver Bob has to efficiently recognize this transaction and recover the corresponding signing key by using the key-generation protocol.

This protocol is formalized as four efficient procedures $\mathcal{KG}$=(**Setup**, **UKeyGen**, **DKeyGen**, **DKeyRec**) which are short forms for setup, user keys generation, destination keys generation, and destination keys recovery respectively.

Setup$(1^n, 1^\lambda)$: On input security parameter, this procedure generates the global parameters pp for the whole cryptocash system which means this procedure also runs $\mathcal{LRS}.\mathbf{Setup}(1^n)$ and $\mathcal{ES}.\mathbf{Setup}(1^n)$ as subroutines so that the signature scheme and encryption scheme are accurately initiated (see Sect. 2.3 and Sect. 4.2 for details). Let $(n, m, \mathbf{G}, \mathbf{H}, \mathcal{H}, q, t, N)$ be the system parameters of the linkable ring signature, and $R = \mathbb{Z}[x]_q/\langle x^n + 1\rangle$. Besides that, it chooses a cryptographic hash function $hash : \{0,1\}^* \to \{0,1\}^\lambda$. Let $\bar{D} = \{g \in R : \|g\| \leq t/2\}$.

UKeyGen(pp)**:** When a user wants to join the cryptocash system, he executes this procedure. This procedure first generates the keys for public key encryption scheme $(epk, esk) \leftarrow \mathcal{ES}.\mathbf{KGen}(pp)$. It then generates a partial key pair of the linkable ring signature scheme $\mathbf{X} \leftarrow \bar{D}^{m\times m}$, $\mathbf{Y} = \mathbf{GX}$. Note that the norm of the partial signing key $\mathbf{X}$ is a little smaller than the one of the original linkable ring signature. (epk, esk) and $(\mathbf{Y}, \mathbf{X})$ are two pairs of public and private keys which are held by the user.

DKeyGen$(pp, epk, \mathbf{Y})$**:** If Alice wants to send coins to Bob who holds keys (epk, esk), $(\mathbf{Y}, \mathbf{X})$, she runs the procedure with epk and $\mathbf{Y}$. This procedure samples $\mathbf{X}_p \leftarrow \bar{D}^{m\times m}$ and generates the destination key $\mathbf{Y}_d = \mathbf{GX}_p + \mathbf{Y}$ for Bob. $\mathbf{X}_p$ is a part of the signing key with respect to the destination key $\mathbf{Y}_d$, but no one except Bob can recover the integral signing key. This procedure proceeds to pick an AES secret key k uniformly at random. It then computes $c_1 = \mathcal{ES}.\mathbf{Enc}_{epk}(k)$ with the public key encryption and computes $c_2 = \mathbf{AES}_k(hash(epk)\|\mathbf{X}_p)$ with the AES algorithm. Finally, it outputs the destination key $\mathbf{Y}_d$, and the auxiliary information c_1, c_2. The process of **DkeyGen** procedure is depicted in Fig.1.

DKeyRec$(pp, epk, esk, \mathbf{Y}, \mathbf{X}, (\mathbf{Y}_d, c_1, c_2))$**:** Bob runs this procedure to check every passing transaction. If Alice's transaction with Bob as recipient was among them, it will be that (1) $k = \mathcal{ES}.\mathbf{Dec}_{esk}(c_1)$; (2) $(hash(epk)\|\mathbf{X}_p) = \mathbf{AES}_k(c_2)$. If this procedure finds that the first part of the plaintext of c_2 is not the hash value of Bob's public encryption key epk, then this procedure aborts and outputs 0. Otherwise, Bob computes $\mathbf{X}_d = \mathbf{X}_p + \mathbf{X}$ and $\mathbf{Y}'_d = \mathbf{GX}_d$. If $\mathbf{Y}'_d = \mathbf{Y}_d$, this procedure outputs 1 and admits the validity of the destination key $\mathbf{Y}_d$ and its signing key $\mathbf{X}_d$. Since $\|\mathbf{X}_d\| \leq \|\mathbf{X}_p\| + \|\mathbf{X}\| \leq t$, $\mathbf{X}_d$ is a valid signing key correlative to the destination key $\mathbf{Y}_d$. The process of this procedure is briefly shown in Fig.2.

5.3 Transactions

We proceed to introduce the transactions in APQC. Let Bob and Alice be two users of our APQC. Bob will runs $\mathcal{KG}.\mathbf{UKeyGen}$ to generates his fixed user key

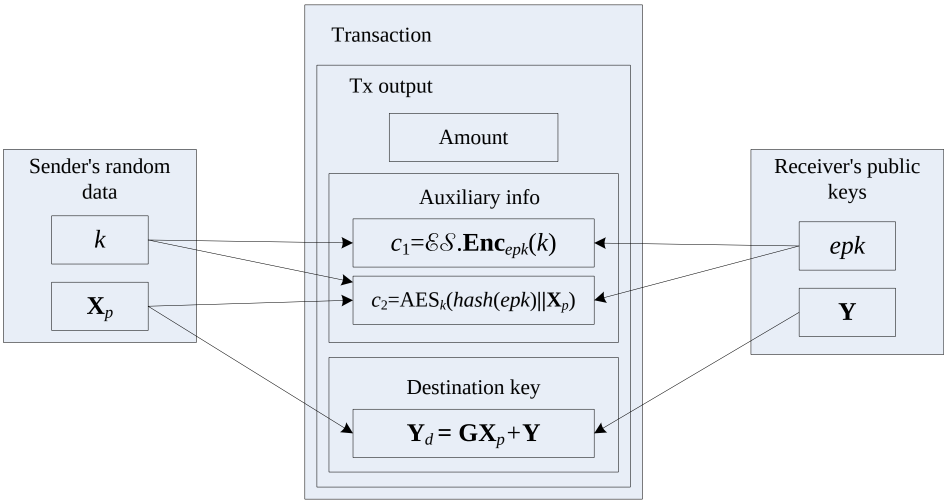
Transaction
Tx output
Amount
Auxiliary info
c_1=$\mathcal{ES}$.$\mathbf{Enc}_{epk}(k)$
c_2=AES$_k$($hash$(epk)||$\mathbf{X}_p$)
Destination key
$\mathbf{Y}_d = \mathbf{G}\mathbf{X}_p+\mathbf{Y}$
Sender's random data
k
$\mathbf{X}_p$
Receiver's public keys
epk
$\mathbf{Y}$

Fig. 1. DKeyGen procedure

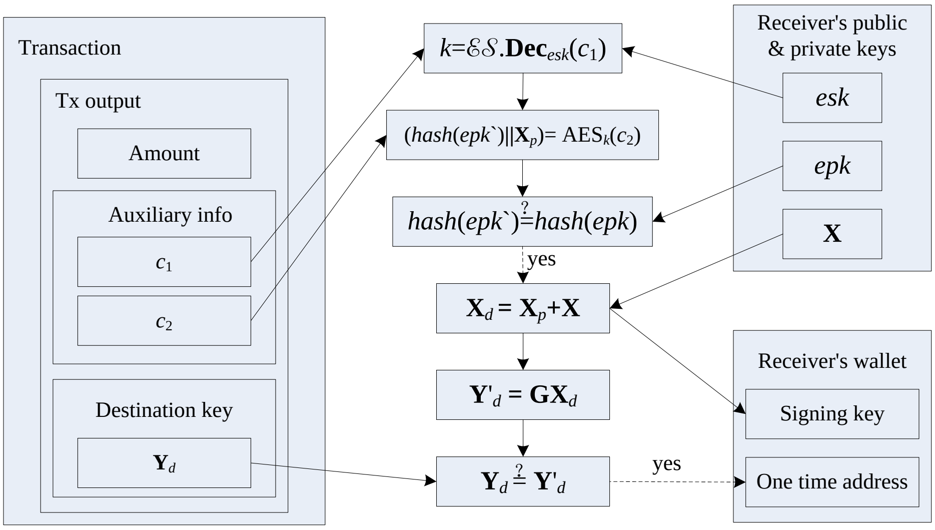
Transaction
Tx output
Amount
Auxiliary info
c_1
c_2
Destination key
$\mathbf{Y}_d$
k=$\mathcal{ES}$.$\mathbf{Dec}_{esk}(c_1)$
($hash$(epk`)||$\mathbf{X}_p$)= AES$_k$(c_2)
$hash(epk`)\overset{?}{=}hash(epk)$
yes
$\mathbf{X}_d = \mathbf{X}_p+\mathbf{X}$
$\mathbf{Y'}_d = \mathbf{G}\mathbf{X}_d$
$\mathbf{Y}_d \overset{?}{=} \mathbf{Y'}_d$
yes
Receiver's public & private keys
esk
epk
$\mathbf{X}$
Receiver's wallet
Signing key
One time address

Fig. 2. DKeyRec procedure

pairs $(epk_{\text{Bob}}, esk_{\text{Bob}})$, $(\mathbf{Y}_{\text{Bob}}, \mathbf{X}_{\text{Bob}})$ when he initially joins the system. Similarly, $(epk_{\text{Alice}}, esk_{\text{Alice}})$, $(\mathbf{Y}_{\text{Alice}}, \mathbf{X}_{\text{Alice}})$ are the key pairs hold by Alice. Besides the user keys, Alice and Bob maintain their own wallet addresses respectively.

Now, assume that the destination address $\mathbf{Y}_{Bj}$ and its signing key $\mathbf{X}_{Bj}$ are in Alice's wallet, and she wants to sent coins of this address to Bob. Alice will specify $N-1$ foreign outputs $(\text{Output}_{B1}, \ldots, \text{Output}_{B(j-1)}, \text{Output}_{B(j+1)}, \ldots, \text{Output}_{BN})$ in which the amount is equivalent to that of Output_{Bj}. She proceeds to find Bob's public keys epk_{Bob} and $\mathbf{Y}_{\text{Bob}}$ and runs $\mathcal{KG}.\mathbf{DkeyGen}(pp, epk_{\text{Bob}}, \mathbf{Y}_{Bob})$ to generate the destination key $\mathbf{Y}_{Cj}$ and its auxiliary information c_1, c_2 for Bob (see Fig.1). She then pushes (1) Tx input including $\{\text{Output}_{Bi}\}_{i=1}^{N}$ and the amount she sends to Bob, (2) the destination key $\mathbf{Y}_{Cj}$ and auxiliary information c_1, c_2 she generated for Bob, (3) all previous transactions with output $\{\text{Output}_{Bi}\}_{i=1}^{N}$, into the hash function and signs the hash value h by running $\sigma \leftarrow \mathcal{LRS}.\mathbf{Sign}(pp, \mathbf{X}_{Bj}, h, \mathbf{Y}_{B1}, \ldots, \mathbf{Y}_{BN})$. Finally she broadcasts the transaction which transfers coins from address $\mathbf{Y}_{Bj}$ to $\mathbf{Y}_{Cj}$.

Bob checks all passing transactions. For each transaction, he extracts the destination key and auxiliary information $(\mathbf{Y}_d, c_1, c_2)$ and runs the procedure $\mathcal{KG}.\mathbf{DKeyRec}(pp, epk_{Bob}, esk_{Bob}, \mathbf{Y}_{Bob}, \mathbf{X}_{Bob}, (\mathbf{Y}_d, c_1, c_2))$. If this transaction is the one that Alice sent to Bob, the foregoing procedure will return the signing key $\mathbf{X}_{Cj}$ for the destination key $\mathbf{Y}_d = \mathbf{Y}_{Cj}$. If this happens, Bob accepts this transaction and records $\mathbf{X}_{Cj}$, $\mathbf{Y}_d$ into his wallet. Bob can later spend the coin stored in the destination address $\mathbf{Y}_d$ because he has the signing key $\mathbf{X}_{Cj}$.

The standard transaction is also briefly depicted in the full version [38].

6 Conclusions and Future Works

While a lot of lattice-based ring signature and standard signature have recently been designed, linkable ring signature over lattices has not been to the best of our knowledge. The strong similarity in the construction between a lattice-based signature and DLP-based one, e.g., the signature in [18] and the Schnorr signature [32, 33], can help us to design the lattice-based counterparts of DLP-based linkable ring signatures. In this paper, using the techniques in [12], we construct a linkable ring signature from lattices in which the size of a signature, on behalf of a ring with N participants, is $O(\log N)$. Based on the proposed signature scheme, we present an anonymous post-quantum cryptocash system by following the major ideas in CryptoNote and Monero. In order to generate stealth addresses (verifying keys) and recover corresponding signing keys for the linkable ring signature, we provide a key-generation protocol as a subroutine of the cryptocash system. By combining all those techniques together, our cryptocash protocol obtains a new level anonymity comparing to the original Bitcoin system. Furthermore, the new designed cryptocash system has the potential to resist quantum attacks.

Recently, the unlinkability and untraceability of Monero were analyzed by [24] and [14]. Some of them were blamed on the abuses of users, e.g. signing a transaction on behalf of a ring with only 1 participant; Besides, there are still a few inherent weakness in Monero, e.g. for a overwhelming proportion of input addresses, a user can't find enough addresses with the same value to hide, especially in the early time of the system. Next, we shall trace these problems and discuss what should be done to make our cryptocash system secure under these analyses. A full cryptocash system will be implement to test the communication and computation costs. And if possible, we would like to contribute our system to the cryptocash community for public usage.

Acknowledgements

This work was supported by the National Key R&D Program of China (2017YB0802500), the National Natural Science Foundation of China (No. 61672550) and the Fundamental Research Funds for the Central Universities (No.17lgjc45).

The authors are grateful to the anonymous reviewers for their valuable suggestions and comments on this paper.

References

1. Aharonov, D., Regev, O.: Lattice problems in NP $\cap$ coNP. J. ACM 52(5), 749–765 (Sep 2005)
2. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). STOC 1996. pp. 99–108. ACM (1996)
3. Barber, S., Boyen, X., Shi, E., Uzun, E.: Bitter to better — how to make Bitcoin a better currency. FC 2012. pp. 399–414. Springer Berlin Heidelberg (2012)
4. Bellare, M., Boldyreva, A., Desai, A., Pointcheval, D.: Key-privacy in public-key encryption. ASIACRYPT 2001. pp. 566–582. Springer Berlin Heidelberg (2001)
5. Cai, J.Y., Nerurkar, A.P.: An improved worst-case to average-case connection for lattice problems. FOCS 1997. pp. 468–477. IEEE (Oct 1997)
6. Cayrel, P.L., Lindner, R., Rückert, M., Silva, R.: A lattice-based threshold ring signature scheme. LATINCRYPT 2010. pp. 255–272. Springer Berlin Heidelberg (2010)
7. Ducas, L., Durmus, A., Lepoint, T., Lyubashevsky, V.: Lattice signatures and bimodal gaussians. CRYPTO 2013. pp. 40–56. Springer Berlin Heidelberg (2013)
8. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. CRYPTO' 86. pp. 186–194. Springer Berlin Heidelberg (1987)
9. Fujisaki, E., Suzuki, K.: Traceable ring signature. PKC 2007. pp. 181–200. Springer Berlin Heidelberg (2007)
10. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. STOC 2008. pp. 197–206. ACM (2008)
11. Goldreich, O., Goldwasser, S.: On the limits of non-approximability of lattice problems. STOC 1998. pp. 1–9. ACM (1998)
12. Groth, J., Kohlweiss, M.: One-out-of-many proofs: Or how to leak a secret and spend a coin. EUROCRYPT 2015. pp. 253–280. Springer Berlin Heidelberg (2015)
13. Halevi, S.: A sufficient condition for key-privacy. Cryptology ePrint Archive, Report 2005/005 (2005)
14. Kumar, A., Fischer, C., Tople, S., Saxena, P.: A traceability analysis of Monero's blockchain. Cryptology ePrint Archive, Report 2017/338 (2017)

15. Libert, B., Ling, S., Nguyen, K., Wang, H.: Zero-knowledge arguments for lattice-based accumulators: Logarithmic-size ring signatures and group signatures without trapdoors. EUROCRYPT 2016. pp. 1–31. Springer Berlin Heidelberg (2016)
16. Liu, J.K., Wei, V.K., Wong, D.S.: Linkable spontaneous anonymous group signature for ad hoc groups. ACISP 2004. pp. 325–335. Springer Berlin Heidelberg (2004)
17. Liu, J.K., Wong, D.S.: Linkable ring signatures: Security models and new schemes. ICCSA 2005. pp. 614–623. Springer Berlin Heidelberg (2005)
18. Lyubashevsky, V.: Lattice signatures without trapdoors. EUROCRYPT 2012. pp. 738–755. Springer Berlin Heidelberg (2012)
19. Lyubashevsky, V., Micciancio, D.: Generalized compact knapsacks are collision resistant. ICALP 2006. pp. 144–155. Springer Berlin Heidelberg (2006)
20. Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. FOCS 2002. pp. 356–365 (2002)
21. Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions. computational complexity 16(4), 365–411 (Dec 2007)
22. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. SIAM Journal on Computing 37(1), 267–302 (2007)
23. Miers, I., Garman, C., Green, M., Rubin, A.D.: Zerocoin: Anonymous distributed e-cash from Bitcoin. SP 2013. pp. 397–411 (May 2013)
24. Miller, A., Möser, M., Lee, K., Narayanan, A.: An empirical analysis of linkability in the Monero blockchain. eprint arXiv:1704.04299 (2017),
25. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system. URL: http://www. bitcoin. org/bitcoin. pdf (2012)
26. Noether, S.: Ring signature confidential transactions for Monero. Cryptology ePrint Archive, Report 2015/1098 (2015)
27. Ober, M., Katzenbeisser, S., Hamacher, K.: Structure and anonymity of the Bitcoin transaction graph. Future Internet 5(2), 237–250 (2013)
28. Reid, F., Harrigan, M.: An Analysis of Anonymity in the Bitcoin System, pp. 197–223. Springer New York (2013)
29. Rivest, R.L., Shamir, A., Tauman, Y.: How to leak a secret. ASIACRYPT 2001. pp. 552–565. Springer Berlin Heidelberg (2001)
30. Ron, D., Shamir, A.: Quantitative analysis of the full Bitcoin transaction graph. FC 2013. pp. 6–24. Springer Berlin Heidelberg (2013)
31. Saberhagen, N.v.: Cryptonote v 2. 0. HYPERLINK https://cryptonote. org/whitepaper. pdf (2013)
32. Schnorr, C.P.: Efficient identification and signatures for smart cards. EUROCRYPT '89. pp. 688–689. Springer Berlin Heidelberg (1990)
33. Schnorr, C.P.: Efficient signature generation by smart cards. Journal of Cryptology 4(3), 161–174 (Jan 1991)
34. Stehlé, D., Steinfeld, R., Tanaka, K., Xagawa, K.: Efficient public key encryption based on ideal lattices. ASIACRYPT 2009. pp. 617–635. Springer Berlin Heidelberg (2009)
35. 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. ESORICS 2017. pp. 456–474. Springer International Publishing (2017)
36. Wang, C., Wang, H.: A new ring signature scheme from NTRU lattice. ICCIS 2012. pp. 353–356. IEEE (Aug 2012)
37. Wang, J., Sun, B.: Ring signature schemes from lattice basis delegation. ICICS 2011. pp. 15–28. Springer Berlin Heidelberg (2011)
38. Zhang, H., Zhang, F., Tian, H., Au, M.H.: Anonymous post-quantum cryptocash. Cryptology ePrint Archive, Report 2017/716 (2017)