A Secure and Efficient Conditional Anonymous Scheme for Permissionless Blockchains

Ruiyang Li[1], Jun Shen[1], Shichong Tan[1], Kuan-Ching Li[2]

[1] State Key Laboratory of Integrated Service Networks (ISN), Xidian University, China
[2] Department of Computer Science and Information Engineering, Providence University, Taiwan
ry_Li@stu.xidian.edu.cn, demon_sj@126.com, sctan@mail.xidian.edu.cn, kuancli@pu.edu.tw

Abstract

The conditional anonymous scheme for permissionless blockchains satisfies anonymity and traceability simultaneously. Specifically, though transactions are published anonymously, the supervision authority can still trace the identity of the transaction producer when there is a suspicious transaction. However, the uncontrolled randomness employed in existing schemes leads to security risks and unsatisfying storage efficiency. In order to deal with these problems, we propose a secure and efficient conditional anonymous scheme in permissionless blockchains in this paper. In particular, we employ the bilinear ring signature to avoid security risks such as secret key leakage, covert communication and hidden persistent storage. Subsequently, the double-chain structure is introduced to significantly improve the storage efficiency, in which a supervision chain storing users' hidden identity information is built on top of permissionless blockchains. We demonstrate the security and efficiency of our conditional anonymous scheme by numerous analyses and experiments.

Keywords: Permissionless blockchain, Conditional anonymity, Double-chain structure, Bilinear ring signature

1 Introduction

Blockchain has been considered as one of the most disruptive and revolutionary technological innovations in recent years, benefitting from its decentralization, immutability, spontaneity, etc. These features provide a good advantage to solve the problems of single-point failure and coin issuance in the centralized electronic cash system [1], making cryptocurrency be the widest application of blockchain.

Transparent transactions in cryptocurrencies pose a threat to users' privacy and security. Though some cryptocurrencies such as Bitcoin [2] and Ethereum [3] use the pseudonym technology to hide users' identities to a certain extent, it is easy to match users' identities with transaction data by repeatedly using transaction addresses, and there are still hidden dangers in terms of privacy [4]. Consequently, a large number of cryptocurrencies with privacy preserving have emerged. From Dash [5] to Monero [6-7] and Zcash [8-9], cryptographic technologies such as linkable ring signature [10] and Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zk-SNARKs) [11] are used to realize privacy preserving.

Unconditional anonymity not only protects users' privacy, but also provides a natural barrier for criminals. In recent years, illegal behaviors under the guise of anonymous cryptocurrency such as money laundering, smuggling, drug trafficking and extortion, have brought challenges to the regulator. Take WannaCry for a simple instance. In May 2017, hundreds of thousands of hosts and systems in more than 150 regions were infected by this global ransomware worm in a short period of time, blackmailing victims' Bitcoins by encrypting files [12]. In the next few months, criminals moved their assets to Monero, which brought great challenges to the authority in tracing them. Criminals have laundered billions of dollars every year through cryptocurrencies such as Zcash and Monero in recent years, according to the research by cybersecurity firm CipherTrace.

Compared with the permissioned blockchain with a registration authority, criminals tend to conduct transactions in the permissionless blockchain with privacy preserving, so balancing anonymity and traceability in permissionless blockchain is necessary. As far as we know, most permissionless blockchains with privacy preserving use the linkable ring signature technology to achieve unconditional anonymity, which leads to large-scale transaction signatures and unsatisfying storage efficiency. Furthermore, the research results of Alsalami et al. [13] show that cryptocurrency using linkable ring signature technology has a lot of randomness, which leads to covert communication and secret key leakage. Therefore, it is significantly crucial to propose a secure and efficient conditional anonymous scheme for permissionless blockchains.

[*] Corresponding Author: Jun Shen; E-mail: demon_sj@126.com

DOI: 10.53106/160792642021112206002

1.1 Our Contribution

In order to improve the security and storage efficiency, in this work, we design a secure and efficient conditional anonymous scheme for permissionless blockchains. Our contributions are as follows:

We improve the storage efficiency in traceable anonymous blockchains by reducing the size of transactions. Specially, we put the identity information associated with user behavior in the supervision blockchain which is maintained by a supervision authority, relieving storage burden of the underlying permissionless blockchains.

We enhance the security of existing schemes which have security risks brought by uncontrolled randomness. In particular, we employ the bilinear ring signature rather than the linkable ring signature based on the zero-knowledge proof, which achieves the purpose of avoiding security risks like secret key leakage and covert communication.

We analyze the security of the proposed scheme under the co-CDH assumption in the random oracle model, including unforgeability, anonymity, and traceability. In addition, the experimental results show the efficiency of our scheme.

1.2 Related Work

Anonymity and traceability are two opposite aspects of the blockchain system, and how to balance the relationship between them has been studied by many researchers.

(1) Anonymity in blockchains. Initially, though Bitcoin and Ethereum employ pseudonyms to achieve anonymity, this weakened privacy preserving mechanism poses great security risks to user's privacy. It is easy to utilize transaction analysis or social engineering methods to match the identity of the user. Dash [5], the first cryptocurrency with real privacy preserving, was published in 2014 using the coin shuffle strategy. This strategy separates the relationship between the input and output addresses by introducing a large number of unrelated accounts in the transaction. However, the use of a large number of unrelated accounts not only reduces the efficiency of transactions, but also wastes the precious storage space of the blockchain system.

Some blockchain systems exploit cryptographic tools to reach a higher level of anonymity, except for the coin shuffle strategy. Zerocash [9], an improved version of Zerocoin, was proposed in 2014 utilizing zk-SNARKs to hide the identity of the user. Transaction amounts are hidden and verified through the homomorphic Pedersen commitment simultaneously. This scheme provides an anonymous cryptocurrency with a strict security level from the perspective of cryptography, but due to the large number of bilinear pairing operations used in zk-SNARKs, the overall efficiency of the scheme is intolerable. In the same year, another anonymous cryptocurrency with a completely different structure, Monero [7], was proposed, which was the first blockchain system to realize unconditional anonymity by the linkable ring signature. The payer has generated a one-time address for payee and has signed the transaction by the linkable ring signature, before the transaction was published. The efficiency of Monero is significantly higher than Zerocash, but it does not consider the hiding of the amount, so that the adversary can infer the identity of the adversary through the amount. In 2017, Sun et al. [6] proposed an efficient RingCT protocol (called RingCT 2.0) based on Monero, which realized the hiding of transaction amounts. This scheme significantly improves the privacy and security of users in Monero.

In 2019, research by Alsalami et al. [13] showed that many anonymous currencies, such as Monero, have a lot of uncontrol randomness which provides the possibility of security risks such as hidden communication, persistent storage and secret key leakage, when users generate ring signature or zero-knowledge proof.

(2) Traceability in blockchain. Anonymity while providing privacy preserving to the user, but also provides a natural barrier for the malicious acts of criminals, bringing challenges to regulatory authority to trace the identity of criminals. In order to reduce the abuse of anonymity, numerous researches have been carried out around traceability though transaction flow analysis and cryptographic tools.

Originally, some researchers [14-17] realized the traceability of users' identities from the transaction flow analysis. Abundant researches trace the identity of anonymous users in Bitcoin though different transaction flow analysis techniques such as cluster analysis, network topology reproduction and transaction injection. In the Monero with high anonymity, the transaction flow analysis technology can also achieve the purpose of associating the address with users. In 2017, Kumar et al. [18] utilized the correlation between users, the aggregation characteristics of the transaction and an analysis attack based on transaction output time respectively to remove the anonymity of Monero. However, such methods often require too much data to trace efficiently when users only make a few transactions.

In order to theoretically achieve traceability, some researches have gradually begun to seek ways of adding cryptographic tools to the scheme. Using cryptographic tools to solve the traceability of blockchain system is mainly divided into two types: deployment in permissioned blockchains and permissionless blockchains. In 2016, Garman et al. [19] introduced privacy responsibility in DAP protocol, allowing selective tracking of users and coins in transactions through hybrid encryption. Though this scheme achieves traceability for the specific scenario of Zerocash, it still cannot avoid the problem of low efficiency in Zerocash. In 2019, Zhang et al. [20] proposed an anonymous blockchain tracking scheme based on linkable group signature [21]. The group administrator is able to trace the real signer when necessary. Wu et al. [22] employed the ciphertext-policy attribute-based encryption (CP-ABE) to implement the audit and key track the ciphertext data stored in the blockchain, so as to achieve conditional anonymity. However, these two schemes only solve the problem of anonymous traceability in permissioned blockchains, and they are not suitable for permissionless blockchains with higher crime rates. In the same year, Li et al. [12] implemented Traceable Monero using linkable ring signature and one-way domain accumulators, allowing the regulator to track users' long-term addresses as well as one-time addresses. However, the construction of Tag may expose the privacy of users. Adversary can use bilinear mapping to solve the co-DDH problem, and it is easy to match the signer. In 2020, Huang et al. [23] utilized the connectable group signature to realize anonymous tracking in the blockchain system. Ma et al. [24] proposed the SkyEye scheme implementing chameleon hash algorithm and zk-SNARKs to realize traceability. Similarly, the above two schemes still only apply to permissioned blockchains. In contrast, we explore to ensure the security and efficient conditional anonymous scheme for permissionless blockchains.

1.3 Organization

The rest of the paper is organized as follows. Section 2 introduces some preliminaries. Section 3 describes the system model and definitions. We present details of the proposed secure and efficient conditional anonymous scheme for permissionless blockchains, security analysis and efficiency analysis respectively in section 4. Section 5 shows the performance evaluation. Finally, the conclusion is drawn in Section 6.

2 Preliminaries

We list corresponding notations in Table 1 and recall preliminaries used in this paper.

2.1 Bilinear Maps

Let $\mathbb{G}_1, \mathbb{G}_2$ and $\mathbb{G}_T$ be three multiplicative cyclic groups of the same prime order *p*, such that $|\mathbb{G}_1| = |\mathbb{G}_2| = |\mathbb{G}_T|$. g_1 and g_2 are generators of groups $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. There exists a computable isomorphism ψ from $\mathbb{G}_1$ to $\mathbb{G}_2$, with $\psi(g_1) = g_2$. $e: \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is a computable bilinear map with the following propertirs:

Table 1. Notations

Notation	Description
λ	The security parameter;
$\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$	Cycle groups of prime p in bilinear map;
g_1, g_2	Generators of $\mathbb{G}_1, \mathbb{G}_2$;
e, ψ	The bilinear map e and the corresponding isomorphism ψ;
$((a, b)), ((A, B))$	The user's long-term private and public key pair;
(x_i, P_i)	The user's one-time private key and public key address;
(r, R)	The private and public key pair of supervision authority;
M	The message of signature;
U_{pk}	The set of public keys in the ring signature $\{P_1, \ldots, P_n\}$;
I	The message related link and trace in each ring signature;
σ	The ring signature;
H_1	The secure collision resistant hash function $H_1 : \{0,1\}^* \to \mathbb{G}_2$;
H_2	The secure collision resistant hash function $H_2 : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{Z}_p$;
Aux	The auxiliary information of each one-time public key address;

- Bilinear: $e(u^a, v^b) = e(u, v)^{ab}$ for any $u \in \mathbb{G}_1$, $v \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_p$.
- Non-degenerate: $e(g_1, g_2) \neq 1$.
- Computational: The map e and isomorphism can be computed efficientiy for all $u \in \mathbb{G}_1$, $v \in \mathbb{G}_2$.

Then we define the following intractable problems in $\mathbb{G}_1$ and $\mathbb{G}_2$.

Definition 1 (Discrete Logarithm Problem). DL problem states that given a tuple $(g, g^a) \in \mathbb{G}$ in which $\mathbb{G}$ is a cycle group, *g* is the generator of it, $a \in_R \mathbb{Z}_p$ is a random number output *a*. DL assumption holds that the following advantage $Adv_{\mathcal{A}}^{DL}$ is negligible in group $\mathbb{G}$ and λ for any polynomial-time algorithm $\mathcal{A}$.

$$Adv_{\mathcal{A}}^{DL}(\lambda) = \Pr[\mathcal{A}(g, g^a) = a : a \xleftarrow{R} \mathbb{Z}_p] \leq \text{negl}(\lambda).$$

Definition 2 (Computational Co-Diffie-Hellman Problem). Co-CDH problem is that, given $g_1, g_1^a \in \mathbb{G}_1$, $a \xleftarrow{R} \mathbb{Z}_p$ and $h \in \mathbb{G}_2$ output $h^a \in \mathbb{G}_2$. Co-CDH assumption holds that for any polynomial-time algorithm $\mathcal{A}$, the following advanrage $Adv_{\mathcal{A}}^{co-CDH}$ is negligible in groups $\mathbb{G}_1, \mathbb{G}_2$ and λ.

$$Adv_{\mathcal{A}}^{co-CDH}(\lambda) = \Pr[\mathcal{A}(g_1, g_1^a, h) = h^a : a \xleftarrow{R} \mathbb{Z}_p, \\ h \xleftarrow{R} \mathbb{G}_2] \leq \text{negl}(\lambda).$$

It can be proved that co-CDH problem is hard under bilinear map groups.

2.2 Bilinear Ring Signatures

Definition 3 (Bilinear Ring Signatures). A bilinear ring signature scheme is consisted of four algorithms [25], $\Sigma = (\text{Setup, KeyGen, RingSign, RingVerity})$, which are defined as follows.

- $\text{Setup}(1^{\lambda}) \rightarrow (param)$: A probabilistic polynomial - time algorithm which, on input the security parameter $\lambda \in \mathbb{N}$, outputs a set of security parameters $param = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e, \psi)$.
- $\text{KeyGen}(n, param) \rightarrow (\{x_i, P_i\})$: A probabilistic polynomial-time algorithm which, on input a size of the ring group *n* and the set of security parameters *param*, outputs *n* pairs of public/secret-key pair (P_i, x_i) for all $i \in [1, n]$.
- $\text{RingSign}(sk_s, m, U_{pk}) \rightarrow (\sigma)$: A probabilistic polynomial-time algorithm which, on input the signer's secret key x_s for $s \in [1, n]$, a message *m* and a set of users' public keys $U_{pk} = \{P_1, ..., P_n\}$, outputs a ring signature $\sigma = \langle \sigma_1, ..., \sigma_n \rangle \in \mathbb{G}_2$, in which $\sigma_s \leftarrow (h / \psi(\prod_{i \neq s} P_i^{a_i}))^{1/x_s}$, $\sigma_i \leftarrow g_2^{\sigma_i} \in \mathbb{G}_2$ for all $i \neq s$, $h \leftarrow H_1(m) \in \mathbb{G}_2$ and $a_i \leftarrow_R \mathbb{Z}_p$.
- $\text{RingVerify} = (\sigma, m, U_{pk}) \rightarrow (1/0)$: A probabilistic polynomial-time algorithm which, on input a ring signature σ, a message m and a set of users' public keys U_{pk} , outputs 1/0 by checking the equation $e(g_1, h) = \prod_{i=1}^{n} e(P_i, \sigma_i)$.

2.3 CryptoNote Technology

CryptoNote [7] is a scheme applied in Monero to hide the actual addresses of payers and payees using one-time address technology and linkable ring signature. The former protects the privacy of the payee's identity, while the latter hides the payer's identity. The formal definition is given as follows:

Definition 4 (One-time address technology). An one-time address scheme is consisted of three algorithms, $\Sigma = (\text{Setup, Long - term KeyGen, One - time KeyGen, Key Recovery})$, which are defined as follows.

- $\text{Setup}(1^{\lambda}) \rightarrow (param)$: A probabilistic polynomial - time algorithm which, on input the security parameter $\lambda \in \mathbb{N}$, outputs a set of security parameters *param*.
- $\text{Long - term KeyGen}(param) \rightarrow (a, b)(A, B)$: A probabilistic polynomial-time algorithm which, on input security parameters *param*, outputs long-term address public/private key $((a, b)(A, B))$.
- $\text{One - time KeyGen}((A, B), r) \rightarrow (P, R)$: A probabilistic polynomial-time algorithm which, on input the long-term public key and a random number $r \xleftarrow{R} [1, l-1]$, outputs an one-time address public key *P* and a public imformation *R*.
- $\text{KeyRecovery}(P, R, (a, b)) \rightarrow (x)$: A probabilistic polynomial-time algorithm which, on input an one-time public key, public imformation *R*, and the corresponding long-term private key, outputs the one-time private key *x*.

Definition 5 (Linkable ring signatures). A linkable ring signature [26-27] is consisted of five algorithms, $\Sigma = (\text{Setup, KeyGen, RingSign, RingVerity, Link})$, which are defined as follows.

- $\text{Setup}(1^{\lambda}) \rightarrow (param)$: A probabilistic polynomial - time algorithm which, on input the security parameter $\lambda \in \mathbb{N}$, outputs a set of security parameters *param*.
- $\text{KeyGen}(n, param) \rightarrow (\{sk_i, pk_i\})$: A probabilistic polynomial-time algorithm which, on input a size of the ring group *n* and the set of security parameters *param*, outputs *n* pairs of public/secret-key pair (pk_i, sk_i) where $i[1, n]$.
- $\text{RingSign}(sk_s, m, U_{pk}) \rightarrow (I, \sigma)$: A probabilistic polynomial-time algorithm which, on input the signer's secret key sk_s , a message *m* and a set of users' public keys $U_{pk} = \{pk_1, pk_2, ...pk_n\}$, in which $pk_s \in U_{pk}$, outputs a ring signature σ and a tag *I*.
- $\text{RingVerify} = (\sigma, I, m, U_{pk}) \rightarrow (1/0)$: A probabilistic polynomial-time algorithm which, on input a ring signature σ, a tag I, a message *m* and a set of users' public keys U_{pk} , outputs 1/0.
- $\text{Link} = ((\sigma_1, I_1), (\sigma_2, I_2)) \rightarrow (Link / \perp)$: A probabilistic polynomial-time algorithm which, on input two ring signatures and corresponding tags, output $Link / \perp$.

3 The System Model and Definitions

In this section, we describe the system model, threat model and security model of our scheme.

3.1 The System Model

As shown in Figure 1, in order to achieve traceability of anonymous users in the permissionless blockchain, our model introduces an additional supervision blockchain. Specifically, such a model involves five entities: the payer of a transaction (*Payer*), the miner in the blockchain system (*Miner*), the supervision authority (*SA*), the anonymous permissionless blockchain (*AP-blockchain*) and the supervision blockchain (*S-blockchain*).

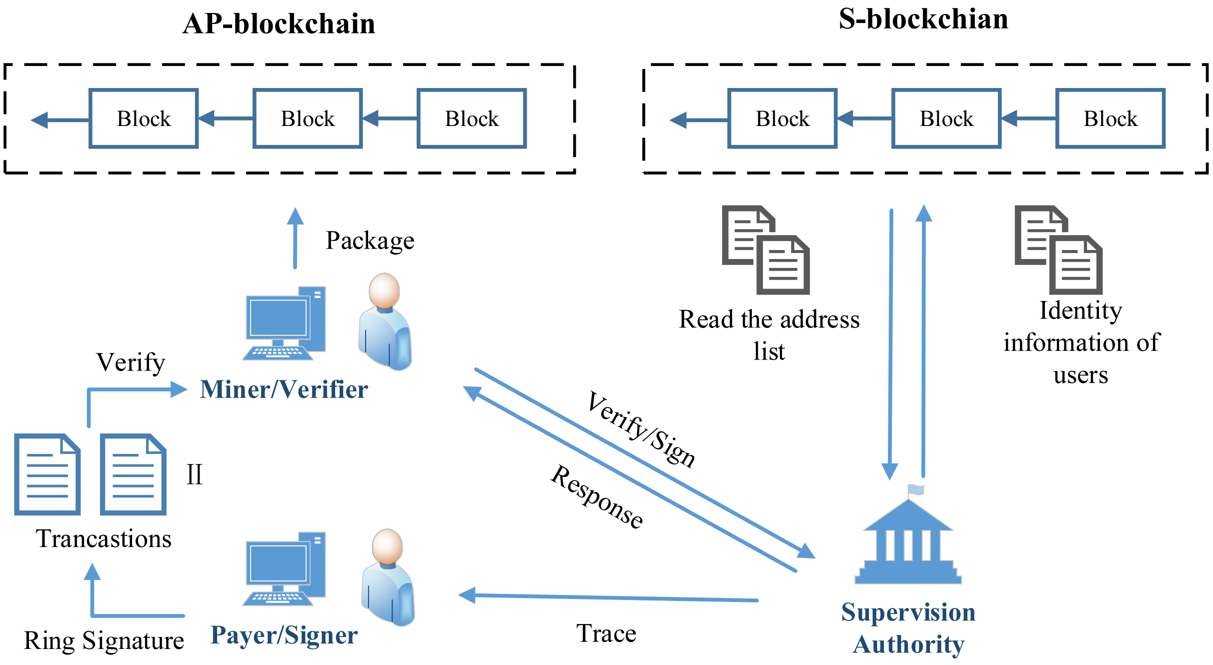

Figure 1. The system model

- *Payer* is a node in *AP-blockchain.* It is responsible for generating transactions.
- *Miner* is a node in *AP-blockchain*. It cooperates with *SA* to complete the legality verification of transactions from *Payer*.
- *SA* is an entity in *S-blockchain*. In addition to verifying the legality of transactions cooperated with *Miner*, it may also trace users' one-time and long-term addresses.
- *AP-blockchain* is a permissionless blockchain, which maintains anonymous transactions of users.
- *S-blockchain* is a permissioned blockchain, which stores users' identity information related to transactions in *AP-blockchain*.

We describe the scheme architecture in brief here. First, *Payer* generates a transaction drawing support with one-time address technology and bilinear ring signature. Subsequently, *Miner* and *SA* jointly verify the legality of the transaction. If the transaction is correctly composed by a honest *Payer*, then the transaction will eventually be packaged into *AP-blockchain* by *Miner*, and the corresponding identity information will be packaged into *S-blockchain* by *SA*. Finally, *SA* may trace users' identity once finding that there are problems with some transactions.

3.2 The Threat Model

Similar to but slightly different from [12], we describe the threat model in our scheme as follows, where the adversary may launch the following attacks. Here we consider that the adversary to launch attacks is an ordinary node with the polynomial time computing capability, rather than a situation where the supervision authority node is corrupted.

- *Double-Spending Attack*: Double-spending attack refers to that the same amount of money has been spent more than once in electronic cash system.
- *Over-Spending Attack*: Over-spending attack means that a user spends more money than the actual amount of the account in a transaction.
- *Anonymity Attack*: Anonymity attack refers that the one but not *SA* can get the identity of a certain payer in a transaction without *SA's* private key.
- *Forgery Attack*: Forgery attack states that a malicious user spends a sum of money in an account without the private key of the corresponding account.
- *Traceability Attack*: Traceability attack indicates that a transaction was published by a *payer* successfully which has been verified and packaged on the blockchain system, but *SA* is unable to trace the payer's actual identity.

3.3 The Security Model

A secure and efficient conditional anonymous scheme for permissionless blockchains should satisfy properties of *Validity*, *Unforgeability*, *Anonymity* and *Traceability*. Specifically, *Validity* aims to prevent from double -spending attack and over-spending attack. *Unforgeability* resist the forgery attack. *Anonymity* aims to prevent from anonymity attack and *Traceability* aims to prevent from the traceability attack. The formal definitions are as follows.

Definition 6 (Validity). Validity means that all transactions generated by honest nodes must pass verification and be packaged into the blockchain, while transactions generated by malicious nodes cannot succeed. It requires that a malicious user cannot (1) spend the money she doesn't have; (2) spend a sum of money more than once; (3) spend a sum of money in excess of the actual amount. A secure and efficient conditional anonymous scheme for permissionless blockchains satisfies validity if for any PPT adversary $\mathcal{A}$,

$$\Pr\left[\begin{array}{cc} & (param) \leftarrow \text{Initialize}(1^{\lambda}); \\ & (P, x) \leftarrow \text{KeyGen}(param); \\ \text{Verify} & (P_{adder}, K, Aux) \\ (Tx, \sigma, \;: & \leftarrow \text{AddrGen}(param); \\ U, I, V) & (m, U) \leftarrow \mathcal{A}(P_{adder,U}); \\ =1 & (Tx, \sigma, V, U) \leftarrow \text{Spend} \\ & (param, m, (x_x, P_s), U,) \\ & R, P_{addr} \end{array}\right] = 1.$$

Definition 7 (Unforgeability). Unforgeability means that an adversary is not able to forge the signature and label of a transaction, without knowing the private key corresponding to a public key in the ring group. A secure and efficient conditional anonymous scheme for permissionless blockchains is unforgeability if for any PPT adversary $\mathcal{A}$

$$\Pr\left[\begin{array}{cc} & (param) \leftarrow \text{Initialize}(1^{\lambda}); \\ \mathcal{A}\ Wins: & (Tx', V', U, \sigma') \leftarrow \\ & \mathcal{A}^{KeyGen,\ AddrGen,\ Spend} \\ & (param) \end{array}\right] \le \text{negl}(\lambda).$$

$\mathcal{A}$ outputs a new tuple (Tx', σ', V', U) with the help of oracles KeyGen, AddrGen, Spend, and $\mathcal{A}$ wins the game if this output satisfies the following conditions:

- $\text{Verify}(Tx', \sigma', V', U) = accept.$
- $P_{Adv} \notin U\ or\ P_{Adv} \in U$ but she doesn't own the corresponding private key.

Definition 8 (Anonymity). A secure and efficient conditional anonymous scheme for permissionless blockchains is anonymity if for any PPT adversay $\mathcal{A}$, it holds that

$$\Pr\left[\begin{array}{cc} & (param) \leftarrow \text{Initialize}(1^{\lambda}); \\ & (P_0, x_0), (P_1, x_1) \leftarrow \\ & \text{KeyGen}(param); \\ b = b': & (\sigma_{x_0}, \sigma_{x_1}) \leftarrow \text{Spend} \\ & (m, (P_0, x_0), (P_1, x_1)); \\ & b \leftarrow \{0, 1\} \\ & b' \leftarrow \text{D}^{\sigma_b, \sigma_0, \sigma_1}(P_0, P_1) \end{array}\right] - \frac{1}{2} \left.\vphantom{\begin{array}{c}1\\1\\1\\1\\1\\1\\1\end{array}}\right| \le \text{negl}(\lambda).$$

In which D is an adversary modeled as a PPT algorithm. P_0, P_1 are the two chosen public keys, where $(P_0, x_0), (P_1, x_1)$ are generated by KeyGen($param$). Before D starts the game with the two public keys (P_0, P_1), the system first generates a challenge number $b \in \{0, 1\}$ which is a random hidden bit. Next, the oracle $\text{Spend}(sk_{i1})$ for $i = \{0, 1, b\}$ generates the three signatures $\sigma_0, \sigma_1, \sigma_b$ with respect to x_0, x_1, x_b on (U, m) where $U = \{P_0, P_1\}$. Finally, D outputs b'.

Definition 9 (Traceability). Traceability means that *SA* can always find the actual payer or signer in a transaction successfully. A secure and efficient conditional anonymous scheme for permissionless blockchains is traceability if for any PPT adversay $\mathcal{A}$,

$$\Pr\left[\begin{array}{cc} & (param) \leftarrow \text{Initialize}(1^{\lambda}); \\ & (Tx', \sigma, V, U) \leftarrow \text{Spend} \\ P_s' = P_s: & (param, m(x_s, P_s), U, R, P_{addr}); \\ & \text{Verify}(Tx, \sigma, U, I, V) = 1; \\ & P_s' \leftarrow \text{Trace}(Tx, \sigma, I, r) \end{array}\right] = 1.$$

4 The Proposed Secure and Efficient Conditional Anonymous Scheme

We first overview the proposed secure and efficient conditional anonymous scheme for permissionless blockchains, and then we present it in more details.

4.1 The High-level Description

We present a secure and efficient conditional anonymous scheme for permissionless blockchains in which users' identities are conditional anonymity. Under normal circumstances, the identity of user is not discovered, just as in the classic anonymous blockchain such as Monero, but the identity also can be revealed when there is a problem with the transaction. Specifically, we achieve this through double-chain structure, one-time address technology and bilinear ring signature. Considering a scenario where *Payer* wants to transfer money form one of her accounts to a payee anonymously. As shown in Figure 2, first, *Payer* generates a one-time address for payee and a *Aux* of this address based on payee's long-term address to esure that the payee can recover the correspongding private key at a later time. Subsequently, *Payer* generates a transaction using bilinear ring signature and publishes it to trading pool. *Miner* packages transactions from the trading pool, then verifies the legality of transactions with *SA*. If the transaction is correctly composed by a honest *Payer*, then the transaction will eventually be packaged into *AP-blockchain* by *Miner*, and the corresponding identity information will be packaged into *S-blockchain* by *SA*. Finally, *SA* may trace users' identities once finding that there are problems with some transactions.

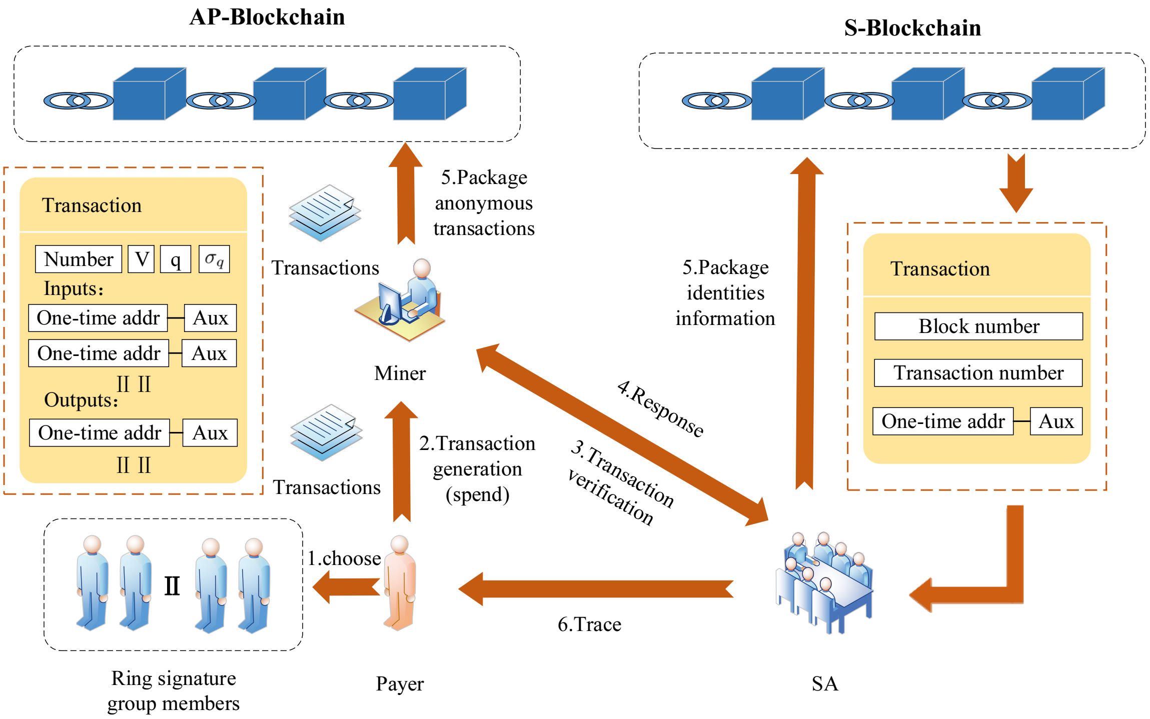

Figure 1. The secure and efficient conditional anonymous scheme for permissionless blockchains

4.2 Detailed Construction of the Proposed Conditional Anonymous Scheme

The secure and efficient conditional anonymous scheme for permissionless blockchains is divided into six algorithms: Initialize, KeyGen, AddrGen, Spend, Verify, and Link & Trace.. The procedure of the scheme is as follows:

- Initialize$(1^\lambda) \to (param)$: : On input the security parameter λ, the system executes as follows:
  - —Choose two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ with prime order p and respective generators g_1 and g_2, the computable isomorphism ψ from $\mathbb{G}_1$ to $\mathbb{G}_2$, and the bilinear map $e: \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, with the target group $\mathbb{G}_T$.
  - —Pick secure hash functions $H_1: \{0,1\}^* \to \mathbb{G}_2$, $H_2: \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{Z}_p$.
  - —Return parameters $param = (\mathbb{G}_1, \mathbb{G}_2, p, e, \psi, g_1, g_2, H_1, H_2)$.
- KeyGen$(param) \to (r, R), (sk_i, pk_i)$: On input *param*, the blockchain system generates key pairs as follows:
  - —*SA* selects private key $r \in_R \mathbb{Z}_p$, calculates public key $R = g_1^r$.
  - —User selects $a_i \in_R \mathbb{Z}_p$, $b_i \in_R \mathbb{Z}_p$ and generates long-term key pair $(sk_i, pk_i) = (a_i, b_i)(g_1^{a_i}, g_1^{b_i})$
  - —Output key pairs of *SA* and users $((r, R), (sk_i, pk_i))$.
- AddrGen$(param, pk_t) \to (P_t, K, Aux)$: On input *param* and the payee's long-term public key $pk_t = (A_t, B_t)$, *Payer* executes as follows:
  - —*Payer* chooses $k \in_R \mathbb{Z}_p$, and calculates $K = g_1^k$.
  - —*Payer* generates the payee's one-time address $P_t = B_t \cdot g_1^{H_2(A_t^k, B_t)}$.
  - —*Payer* encrypts $Aux = Enc_R(B, R^{H_2(K^a, B)})$ of this address with *SA's* pubilc key *R*.
- Spend$(m, (x_s, P_s), U, R) \to (\sigma)$: On input a message *m*, a set of public key group *U* and the public key of *SA*, *Payer* with her secret key x_s executes as follows:
  - —*Payer* picks *n* one-time addresses $U = \{P_i : i \in [1, n]\}$ where $P_s \in U$, and a message m.
  - —*Payer* selects random $\omega_i \in_R \mathbb{Z}_p$ for all $i \neq s$, calculates $h \leftarrow H_1\{m\} \in \mathbb{G}_2$ and sets the bilinear ring signature as:

$$\sigma_i = \begin{cases} \left(\dfrac{h}{\psi \prod\limits_{i \neq s} P_i^{\omega_i}} \right)^{\frac{1}{x_x}} & i = s, \\ g_2^{\omega_i} & i \neq s. \end{cases}$$

  - —*Payer* chooses a random number $q \in_R [1, n]$, produces *V*:

$$V = \begin{cases} \left(\psi \prod\limits_{i \neq s} P_i^{\omega_i} \right) & q = s, \\ \psi(P_q^{\omega_q}) & q \neq s. \end{cases}$$

— *Payer* calculates $I = Enc_R(Sig_{x_s}(R, V), P_x)$, where *Sig* represents a digital signature.

— Output the signature $\sigma = (U, I, V, q(\sigma_1, \ldots, \sigma_n))$

- $\text{Verify}(Tx, \sigma) \rightarrow (1/0)$: On input a transaction Tx and signature σ, *Miner* executes as follows:

— *Miner* calculates $h \rightarrow H_1(m)$, and verifies following two equations:

$$e(g_1, V) = \prod_{i \neq q} e(P_i, \sigma_i),$$
$$e(g_1, h) = e(g_1, V) \cdot e(P_q, \sigma_q).$$

If these two equations are satisfied, *Miner* sends transactions to *SA*.

— *SA* decrypts *Aux* and *I*, gets messages $(Awx_1, Awx_2) = (B_i, R^{H_2(K^a, B_i)})$ and $(I_1, I_2) = (Sig_{x_x}(R, h), P_s)$.

— *SA* verifies

$$\left(\frac{P_{addr}}{Aux_1}\right)^r = Awx_2, (R, h) = Ver_{I_2}(I_1).$$

Ver denotes the verification algorithm of the digital signature.

— *SA* signs transactions and returns them to *Miner*.

— *SA* packages (P_s, I, V) on *S-blockchain*, and *Miner* packages *Tx* on *AP-blockchain*.

- $\text{Line} \,\&\, \text{Trace}(\sigma) \rightarrow ((A_s, B_s) / \perp)$: On input the signature σ of a transaction, *SA* executes as follows:

— *SA* uses its own private key to get the one-time address P_s included in σ.

— *SA* outputs *Link* if P_s has appeared before by finding out from *S-blockchain* then proceeds to the next step, or $\perp$ otherwise.

— *SA* outputs *Payer's* long-term address (A_s, B_s) by decrypting *Aux* of P_s.

This completes the description of the secure and efficient conditional anonymous scheme for permissionless blockchains. Through the specific steps described above, transactions generated by honest *Payers* will eventually be packaged into *AP-blockchain*, and the corresponding identity information will be packaged on *S-blockchain.*

4.3 Security Analysis

In this subsection, we describe the details of the security proofs of unforgeability, anonymity and traceability of the proposed scheme.

Theorem 1. *Our scheme is unforgeable under the co-CDH assumption.*

Proof. The unforgeability proof of bilinear ring signature has been given by Boneh et al. [25], which is ommited here. For the simplification, we just prove that the probability of forging V and passing the verification are negligible.

Given a forger $\mathcal{A}$ against the unforgeability of V in the proposed scheme. We then construct a simulator $\mathcal{B}$ that challenges the co-CDH problem by interacting with $\mathcal{A}$, on input instance (g_1, g_1^a, h). The co-CDH adversary $\mathcal{B}$ sumulates the game for $\mathcal{A}$ as follows.

- Setup. The challenger builds a system to challenge co-CDH problem on input a security parameter λ, in which there is a polynomial time adversary $\mathcal{B}$ to attack the co-CDH problem. Simultaneously, there is a random oracle $\mathcal{O}_s$ queried by polynomial times, which is used to generate the bilinear ring signature.
- Query. In order to use the ability of adversary $\mathcal{A}$ to overcome hard problem, adversary $\mathcal{B}$ trains $\mathcal{A}$, that is, as the challenger of $\mathcal{A}$, adversary $\mathcal{B}$ provides $\mathcal{A}$ with the knowledge required for attack. First, adversary $\mathcal{B}$ picks two public-private key pairs $(P_1, x_1)(P_2, x_2)$ and message m to oracle $\mathcal{O}_s$. Then adversary $\mathcal{B}$ sends $\sigma = (\sigma_1, \sigma_2)$ to $\mathcal{A}$ which is generated by $\mathcal{O}_s$. Finally, adversary $\mathcal{A}$ returns the forged V to $\mathcal{B}$.
- Forgery. In order to attack the co-CDH problem, the challenger generates a challenge co-CDH tuple (g_1, g_1^a, h) for adversary $\mathcal{B}$, in which $g_1 \in \mathbb{G}_1$, $g_2 \in \mathbb{G}_2$. The adversary $\mathcal{B}$ chooses two public - private key pairs $(P_1, x_1)(P_2, x_2) = (g_1^a, a)(g_1^a, b)$, and sends it to oracle $\mathcal{O}_s$. Then the oracle $\mathcal{O}_s$ returns the bilinear ring signature $\sigma = (\sigma_1, \sigma_2)$. The adversary $\mathcal{B}$ picks one of the signatures, replaces it by h and send the new signature $\sigma = (\sigma_1, h)$ to adversary $\mathcal{A}$. The adversary $\mathcal{B}$ sends V to the challenger which is generated by $\mathcal{A}$. Suppose that the non-negligible advantage of adversary $\mathcal{A}$ to forge V successfully is $\Pr[\mathcal{A}\, Wins]$. The advantage for the challenger to successfully challenge co-CDH problem is as follows.

$$Adv_{challenger}^{co-CDH}(\lambda) = \frac{1}{8}\Pr[\mathcal{A}\, Wins]$$

The 1/8 in the above formula is owing to the fact that the challenge can only successfully attack co-CDH problem when the following situations occur. First, only when the oracle $\mathcal{O}_s$ chooses a as the ring signature private key, the adversary $\mathcal{B}$ can construct an attack algorithm based on the returned result. There is 1/2 probability is introduced. Secondly, the adversary $\mathcal{B}$ $\mathcal{B}$ replaces one of the signatures by h. Only when the adversary $\mathcal{B}$ picks the signature $\sigma_1 = \left(\frac{H(m)}{\psi g_1^{b_{\omega 2}}}\right)^{\frac{1}{a}}$, the attack algorithm can proceed. At this time, the public keys and signatures received by adversary $\mathcal{A}$ are

$(P_1,\sigma_1)(P_2,\sigma_2)=(g_1^a,h)(g_1^b,g_2^{\omega_2})$. There is 1/2 probability is introduced. Finally, only when adversary $\mathcal{A}$ chooses $q=2$ and sets $V=\dfrac{H(m)}{h^a}$, the attack algorithm can proceed. There is 1/2 probability is introduced.

The adversary $\mathcal{B}$ calculates $h^a=\dfrac{H(m)}{V}$ and sets it as the answer of the co-CDH challenge tuple. Suppose $\mathcal{A}$ challenges the unforgeability of the scheme successfully with a non-negligible advantage, adversary $\mathcal{B}$ successfully challenges co-CDH problem with 1/8 times of Pr [$\mathcal{A}$ *Wins*] which is still non-negligible. However, the co-CDH problem is difficult to solve under bilinear maps, so the supposition exists contradiction, and the scheme satisfies unforgeability. □

Theorem 2. *Our scheme is anonymous if the bilinear ring signature satisfies anonymity and the asymmetric encryption scheme is IND-CPA.*

Proof. We prove this theorem by feat of the game-based framework. Pr $[Win_i]$, $i\in[0,2]$ denotes the probability of the adversary to win the game in Game 0 to Game 2.

Game 0. This challenge game is defined in Definition 8 and the challenge transaction is denoted as (Tx,σ,U,I,V), where $U=(P_1,P_2)$ with their own $Aux=(Aux_1,Aux_2)$, I is the signcryption of the one-time public key address of the signer and $b\in\{0,1\}$ is serial number of the signer. The adversary $\mathcal{A}$ outputs a guess b' about the actual signer, and we can easily get

$$\Pr[Win_0]=\Pr[b=b'].$$

Game 1. Game 1 is same as Game 0 with one difference which is considering asymmetric encryption with *IND-CPA* security on probability. Since the identity of the real signer is hidden in *I* besides include in the ring signature, we will consider *I* to help the opponent's probability of winning in Game 1. The advantage of Game 1 over Game 0 is only one more public key encryption information, but the asymmetric encryption meets *IND-CPA* security, so the added advantage is negligible. The following equation represents the adversary's probability of winning in Game 1, where λ is the security parameter of this system.

$$|\Pr[Win_1]-\Pr[Win_0]|\le \mathrm{negl}(\lambda)$$

Game 2. Game 2 is same as Game 1 with one difference considering the impact of ring signature on advantage. Game 2 represents the probability of the adversary guessing the real signer in a bilinear ring signature. This is the same as after removing the probability of asymmetric encryption in Game 1.

$$\Pr[Win_2]=\Pr[Win_1].$$

There is only a negligible gap in the winning probability for the adversary between Game 0 and Game 2. Assume that there is no polynomial-time adversary can challenge the anonymity of the bilinear ring signature with a non-negligible probability, our scheme satisfies the property of anonymity. □

Theorem 3. *Our scheme is traceable under the DL assumption.*

Proof. Assume that the one-time address spent by *Payer* is P_s with Aux_s, and the set of ring group is $U_{pk}=\{P_i : i\in[1,n]\}$. The specific steps of the proof are as follows:

- **Correct identity hidden:** In a verified transaction, in order to pass the verification of *SA*, which needs to satisfy the equation $(R,h)=Ver_{I_2}(I_1)$, the *I* must be composed in the correct form $I=Enc_R(Sig_{x_s}(R,V),P_s)=(I_1,I_2)$. The $I_2=P_s$ must meets $P_s\in U_{pk}$, simultaneously.
- **Correct association of long-term address:** Similarly, in order to pass the verification of *SA*, the *Aux*must be correctly formed as $Aux=Enc_R(B,R^{H_2(K^a,B)})$. In order to recognize and spend the one-time address for the payee, *Payer* must correctly generate it. Due to the DL assumption, the probability that a malicious *Payer* constructs the *Aux* that can be verified without *SA's* private key *r* is negligible.

Therefore, the theorem holds when *Payer* runs algorithms AddrGen, Spendand generates (Aux,I) honestly. □

4.4 Efficiency Analysis

In this subsection, we show the efficiency analysis of storeage, communication and computing compared with [12].

(1) Storage and communication efficiency analysis. The overhead of communication and storage analysis is shown in Table 2, where $\mathbb{G}_1$ is the length of the element in group $\mathbb{G}_1$, similarly for $|\mathbb{Z}_p|,|\mathbb{G}_2|,|\mathbb{G}_T|$, $|p|$ is the order of cyclic groups, n and n' are the number of input and output accounts in a transaction and c represents a constant order of magnitude. The details of the results are as follows. In terms of communication overhead, $2(n+n')|\mathbb{G}_1|$ and $(n+1)\mathbb{G}_2$ denote the size of input and output accounts with corresponding *Aux*, respectively. In addition, other auxiliary information such as V and I need to occupy $4|\mathbb{Z}_p|$. However, in the storage overhead, since *AP-blockchain* only needs to store V and σ_q, which is less $n-1$ signatures than communication, the storage overhead is only $(2n+2n')$ $|\mathbb{G}_T|+2|\mathbb{G}_2|+$

$4|\mathbb{Z}_p|+|p|$. It is obvious that our scheme is better than [12] in terms of storage efficiency.

Table 2. Overhead of communication and storage comparison

Schemes	Scheme [12]	Our Scheme
Communication	$c(m+1)(\mathbb{G}_1\vert+\vert\mathbb{Z}_p\vert+\vert\mathbb{G}_q\vert+\vert\mathbb{G}_T\vert)$	$(2n+2n')\vert\mathbb{G}_1\vert+(n+1)\vert\mathbb{G}_2\vert+4\vert\mathbb{Z}_p\vert+\vert p\vert$
Storage	$c(m+1)(\mathbb{G}_1\vert+\vert\mathbb{Z}_p\vert+\vert\mathbb{G}_q\vert+\vert\mathbb{G}_T\vert)$	$(2n+2n')\vert\mathbb{G}_1\vert+2\vert\mathbb{G}_2\vert+4\vert\mathbb{Z}_p\vert+\vert p\vert$

(2) Computational efficiency analysis.

The computational overhead analysis is shown in Table 3, where m is the number of group of input accounts, n is the number of input accounts in each ring signature group. For the sake of simplicity, we denote modular exponentiation operation as E, bilinear pairing operation as P, inversion operation as I_n, isomorphism operation as I_s. We only consider those expensive operations in schemes, such as exponentiation, isomorphism and bilinear pairing, ignoring the cost of other light computations. Regarding the encryption algorithm and signature algorithm used in our scheme, we use ElGamal encryption and BLS signature. The details of the results are as follows. In Spend phase, $(2n-2)E+I_s$ is required in the computation of a bilinear ring signature, in which $(n-1)E+P$ is needed to compute the sth signature and $(n-1)E$ is needed to compute other $n-1$ signatures. Subsequently, $5E+I_s$ is required in the generation of (V,I). In Verify phase, $(n+2)P$, $3E$, $2P$ represent the cost of verification ring signature, decryption, and verification of BLS signature, respectively. Subsequently, $2E+I_n$ are required to decrypt by *SA* in Trace. Since the Trace algorithm in [12] needs to match the ring members orderly, we choose the average time here that needs to perform *n*/2 exponential operations. In addition, our method of computational analysis of comparison scheme is the same as [12].

Table 3. Computational overhead comparison

Schemes	Scheme [12]	Our Scheme
Spend	$n(m+1)E$ $+(m+1)P$	$(3+2n)E$ $+2I_s$
Verify	$n(m+1)E$ $+3(m+1)P$	$3E$ $+(4+n)P$
Trace	$(n/2+2)E$ $+2I_n$	$2E+2I_n$

5 Performance Evaluation

Simulation experiments are conducted on Linux operating system, Intel(R) Xeon CPU E5-2682 v4 @ 2.50GHz processor and 2GB of RAM are used to carry out the following experiments. The GMP library of GNU multi-precision algorithms and PBC library are written in Python language. For security reasons, we set prime q = 1024 bits as the order of cyclic groups.

Figure 3 to Figure 6 show the time cost of our scheme compared with scheme [12]. Figure 3 and Figure 4 describe the time cost comparisons of the Spend and Verify phases. We can see that the time costs of the two schemes in the Spend and Verify phases show a linear growth trend with the increase of the number of input accounts. The difference is that our scheme is better than scheme [12] in this respect. This is because scheme [12] has many bilinear pairing operations in the accumulator with one-way domain and the signature of knowledge used in the Spend and Verify phases. Figure 5 reveals the time cost comparison of transaction verification for users in blockchains. We can see that the user's verification cost of transactions in scheme [12] is the same as the miner's verification cost of transactions, but our scheme has a lower constant time cost in this respect. The reason for this phenomenon is that our scheme aggregates the ring signatures of the payer, and the user only needs to perform three bilinear pairing operations when verifying transactions in the blockchain. Figure 6 shows the time cost comparison of the Trace phases, and experimental result is consistent with theoretical analysis.

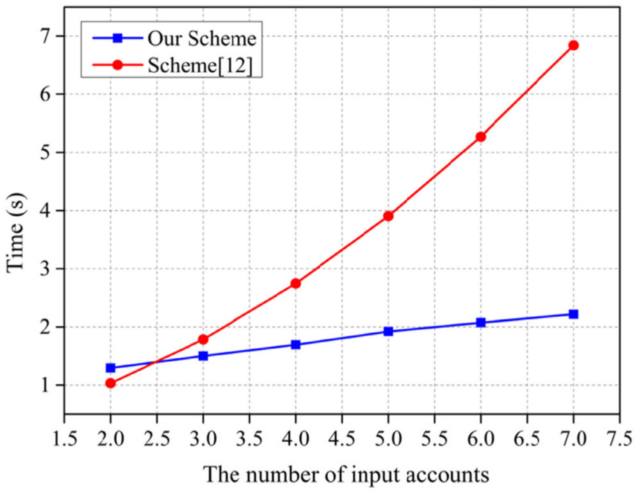

Figure 2. Time cost comparison of the spend phase

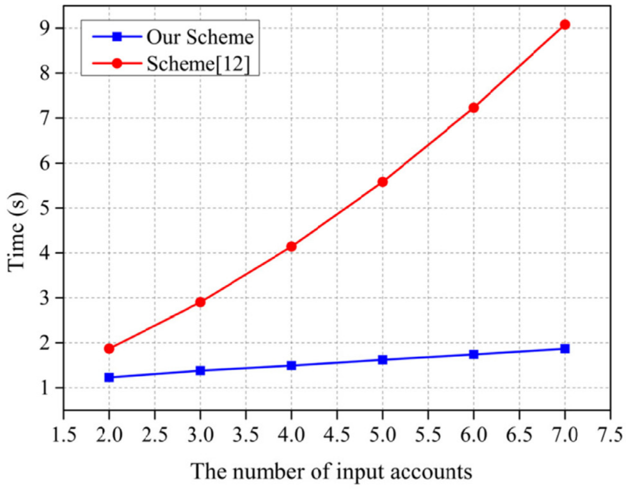

Figure 3. Time cost comparison of the verification phase

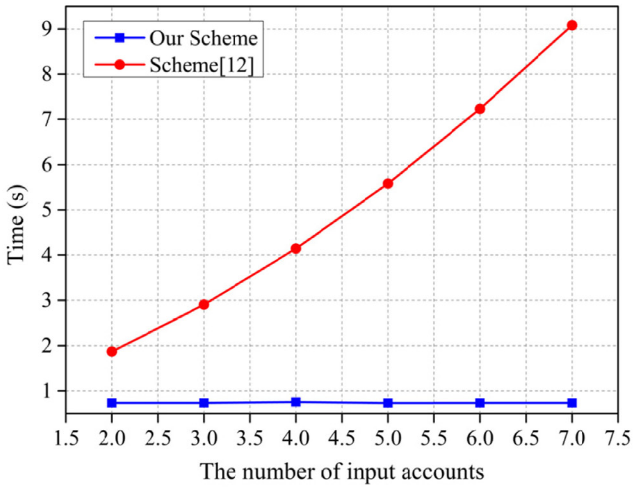

Figure 4. Time cost comparison of transaction verification for users

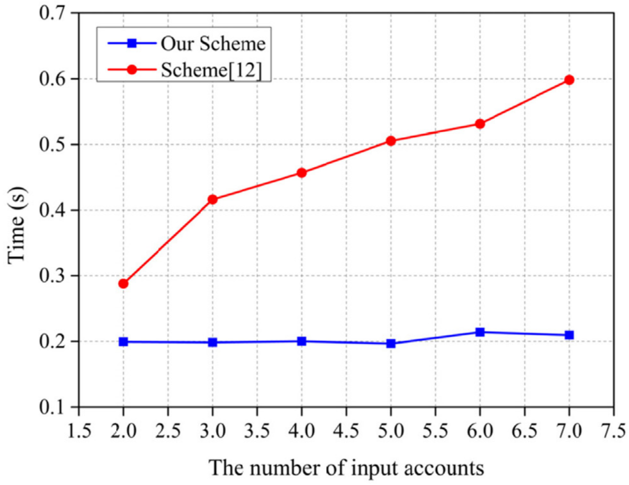

Figure 5. Time cost comparison of the tracing phase

6 Conclusion

In this paper, we propose a secure and efficient conditional anonymous scheme for permissionless blockchains. With the bilinear ring signature and the double-chain structure, it balances anonymity and traceability. Compared with the state of the art, our scheme avoids some security risks brought by uncontrolled rendomness, simultaneously, improve the storage efficiency by reducing the size of transactions. In addition, the scheme satisfies properties of validity, unforgeability, anonymity and traceability. Theoretical analysis and experimental performances show that the proposed scheme is security and efficient.

Acknowledgments

This work is supported by the Key Research and Development Program of Shaanxi (Program No. 2020ZDLGY08-03) and the Shandong provincial Key Research and Development Program of China (2019JZZY020129).

References

[1] D. Chaum, Blind Signatures for Untraceable Payments, *Advances in cryptology*, Santa Barbara, California, USA, 1983, pp. 199-203.

[2] M. Conti, E. S. Kumar, C. Lal, S. Ruj, A Survey on Security and Privacy Issues of Bitcoin, *IEEE Communications Surveys & Tutorials*, Vol. 20, No. 4, pp. 3416-3452, Fourth Quarter, 2018.

[3] H. Chen, M. Pendleton, L. Njilla, S. Xu, A Survey on Ethereum Systems Security: Vulnerabilities, Attacks, and Defenses, *ACM Computing Surveys (CSUR)*, Vol. 53, No. 3, pp. 1-43, June, 2020.

[4] P. Koshy, D. Koshy, P. McDaniel, An Analysis of Anonymity in Bitcoin using P2p Network Traffic, *Financial Cryptography and Data Security-18th International Conference*, Christ Church, Barbados, 2014, pp. 469-485.

[5] N. Aitzhan, D. Svetinovic, Security and Privacy in Decentralized Energy Trading Through Multi-Signatures, Blockchain and Anonymous Messaging Streams, *IEEE Transactions on Dependable and Secure Computing*, Vol. 15, No. 5, pp. 840-852, September-October, 2018.

[6] S. Sun, M. Au, J. Liu, T. Yuen, RingCT 2.0: A Compact Accumulator-based (Linkable Ring Signature) Protocol for Blockchain Cryptocurrency Monero, *European Symposium on Research in Computer Security*, Oslo, Norway, 2017, pp. 456-474.

[7] A. Kumar, C. Fischer, S. Tople, P. Saxena, A Traceability Analysis of Monero's Blockchain, *22nd European Symposium on Research in Computer Security*, Oslo, Norway, 2017, pp. 153-173.

[8] I. Miers, C. Garman, M. Green, A. Rubin, Zerocoin: Anonymous Distributed E-cash from Bitcoin, *2013 IEEE*

Symposium on Security and Privacy, Berkeley, CA, USA, 2013, pp. 397-411.

[9] E. B. Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, M. Virza, Zerocash: Decentralized Anonymous Payments from Bitcoin, *2014 IEEE Symposium on Security and Privacy (SP)*, Berkeley, CA, USA, 2014, pp. 459-474.

[10] J. K. Liu, V. K. Wei, D. S. Wong, Linkable Spontaneous Anonymous Group Signature for Ad Hoc Groups, *Information Security and Privacy: 9th Australasian Conference (ACISP)*, Sydney, Australia, 2004, pp. 325-335.

[11] B. Parno, J. Howell, C. Gentry, M. Raykova, Pinocchio: Nearly Practical Verifiable Computation, *2013 IEEE Symposium on Security and Privacy (SP)*, Berkeley, CA, USA, 2013, pp. 238-252.

[12] Y. Li, G. Yang, W. Susilo, Y. Yu, M. H. Au, D. Liu, Traceable Monero: Anonymous Cryptocurrency with Enhanced Accountability, *IEEE Transactions on Dependable and Secure Computing*, Vol. 18, No. 2, pp. 679-691, March-April, 2021.

[13] N. Alsalami, B. Zhang, Uncontrolled Randomness in Blockchains: Covert Bulletin Board for Illicit Activity, *28th IEEE/ACM International Symposium on Quality of Service (IWQoS)*, Hangzhou, China, 2020, pp. 1-10.

[14] F. Reid, M. Harrigan, An Analysis of Anonymity in the Bitcoin System, 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third International Conference on Social Computing, Boston, MA, USA, 2011, pp. 1318-1326.

[15] D. Ron, A. Shamir, Quantitative Analysis of The Full Bitcoin Transaction Graph, *International Conference on Financial Cryptography and Data Security (FC)*, Okinawa, Japan, 2013, pp. 6-24.

[16] S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G. M. Voelker, S. Savage, A Fistful of Bitcoins: Characterizing Payments Among Men with No Names, *Communications of the ACM*, Vol. 59, No. 4, pp. 86-93, April, 2016.

[17] A. Gaihre, Y. Luo, H. Liu, Do Bitcoin Users Really Care About Anonymity? An Analysis of the Bitcoin Transaction Graph, *IEEE International Conference on Big Data, Big Data 2018*, Seattle, WA, USA, 2018, pp. 1198-1207.

[18] A. Kumar, C. Fischer, S. Tople, P. Saxena, A Traceability Analysis of Monero's Blockchain, *European Symposium on Research in Computer Security*, Oslo, Norway, 2017, pp. 153-173.

[19] C. Garman, M. Green, I. Miers, Accountable Privacy for Decentralized Anonymous Payments, *Financial Cryptography and Data Security - 20th International Conference*, Christ Church, Barbados, 2016, pp. 81-98.

[20] L. Zhang, H. Li, Y. Li, Y. Yu, M. H. Au, B. Wang, An Efficient Linkable Group Signature for Payer Tracing in Anonymous Cryptocurrencies, *Future Generation Computer Systems*, Vol. 101, pp. 29-38, December, 2019.

[21] D. Boneh, H. Shacham, Group Signatures with Verifier-local Revocation, *Proceedings of the 11th ACM conference on Computer and communications security*, Washington, DC, USA, 2004, pp. 168-177.

[22] A. Wu, Y. Zhang, X. Zheng, R. Guo, Q. Zhao, D. Zheng, Efficient and Privacy-preserving Traceable Attribute-based Encryption in Blockchain, *Annals of Telecommunications*, Vol. 74, No. 7-8, pp. 401-411, August, 2019.

[23] H. Huang, X. Chen, J. Wang, Blockchain-based Multiple Groups Data Sharing with Anonymity and Traceability, *Science China Information Sciences*, Vol. 63, No. 3, pp. 1-13, March, 2020.

[24] T. Ma, H. Xu, P. Li, SkyEye: A Traceable Scheme for Blockchain, *IACR Cryptology ePrint Archive*, Vol. 2020, Article No. 34, January, 2020.

[25] D. Boneh, C. Gentry, B. Lynn, H. Shacham, Aggregate and Verifiably Encrypted Signatures from Bilinear Maps, *International Conference on the Theory and Applications of Cryptographic Techniques*, Warsaw, Poland, 2003, pp. 416-432.

[26] E. Fujisaki, K. Suzuki, Traceable Ring Signature, *10th International Conference on Practice and Theory in Public-Key Cryptography*, Beijing, China, 2007, pp. 181-200.

[27] J. K. Liu, M. H. Au, W. Susilo, J. Zhou, Linkable Ring Signature with Unconditional Anonymity, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 26, No. 1, pp. 157-165, January, 2014.

Biographies

Ruiyang Li received the B.S. degree from Jinan University, Guangzhou, China, in 2017. He is currently a M.S. candidate of the School of Cyber Engineering, Xidian University, Xian, China. His research interests include blockchains and data security.

Jun Shen received the B.S. and M.S. degrees from the Nanjing University of Information Science and Technology, Nanjing, China, in 2015 and 2018, respectively. She is currently pursuing the Ph.D. degree with the School of Cyber Engineering, Xidian University, Xian, China. Her research interests include cloud computing security and blockchains.

Shichong Tan received the B.S. degree in Telecommunications Engineering in 2002, the M.S. and Ph.D. degrees in Cryptography from Xidian University, China, in 2005 and 2009, respectively. Since 2005, he has been with Xidian University, where he is now an Associate Professor. His research interests include cloud computing and blockchains.

Kuan-Ching Li (Senior Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering, in 1996 and 2001, respectively. He is currently a Distinguished Professor with the Department of Computer Science and Information Engineering, Providence University, Taiwan His research interests include parallel and distributed processing, GPU/many-core computing, and big data.