Recoverable Private Key Scheme for Consortium Blockchain Based on Verifiable Secret Sharing

Guojia Li, Lin You*, Gengran Hu and Liqin Hu
[1] School of Cyberspace Security, Hangzhou Dianzi University
Hangzhou 310000, China
[e-mail: liguojia@hdu.edu.cn, mryoulin@gmail.com, grhu@hdu.edu.cn, huliqin@hdu.edu.cn]
*Corresponding author: Lin. You

Received February 22, 2021; revised June 5, 2021; accepted July 24, 2021; published August 31, 2021

Abstract

As a current popular technology, the blockchain has a serious issue: the private key cannot be retrieved due to force majeure. Since the outcome of the blockchain-based Bitcoin, there have been many occurrences of the users who lost or forgot their private keys and could not retrieve their token wallets, and it may cause the permanent loss of their corresponding blockchain accounts, resulting in irreparable losses for the users. We propose a recoverable private key scheme for consortium blockchain based on the verifiable secret sharing which can enable the user's private key in the consortium blockchain to be securely recovered through a verifiable secret sharing method. In our secret sharing scheme, users use the biometric keys to encrypt shares, and the preset committer peers in the consortium blockchain act as the participants to store the users' private key shares. Due to the particularity of the biometric key, only the user can complete the correct secret recovery. Our comparisons with the existing mnemonic systems or the multi-signature schemes have shown that our scheme can allow users to recover their private keys without storing the passwords accurately. Hence, our scheme can improve the account security and recoverability of the data-sharing systems across physical and virtual platforms that use blockchain technology.

Keywords: consortium blockchain, private key, threshold secret sharing, verifiability, biometrics.

http://doi.org/10.3837/tiis.2021.08.009 **ISSN : 1976-7277**

1. Introduction

Blockchain uses cryptography algorithm, P2P network architecture, consensus algorithm, and other methods to ensure reliable data transmission, storage, and access functions. As a distributed shared ledger and database, the blockchain can enable peers to establish a trusted distributed system without mutual trust. It has the characteristics of decentralization, non-tampering, traceability, collective maintenance, openness, and transparency. These characteristics ensure the "honesty" and "transparency" of the blockchain and lay the foundation for creating trust in the blockchain. Blockchain applications have extended from cryptocurrency [1] to smart city [2], affiliate systems [3], healthcare [4], supply chain [5, 6], Internet of Things [7, 8], data sharing and storage [9, 10], Information tracking and analysis [11]. However, as a brand-new technology, there are still many difficulties in the combination of blockchain and other technologies. Sáez et al. [12] discussed the challenges faced by blockchain-enabled platforms.

As the underlying technology of decentralized platforms and applications, blockchain has a serious problem in account management: there is no trusted third-party in the blockchain system, and individuals keep the user's private key, if the device has a hard disk crash, data damage, or loss of the device carrying the key, the private key will lose, and the user's blockchain account cannot be retrieved, which causes cause the user's property in the blockchain to be lost. In the blockchain system, the private key is the user's unique identity, and the secure key storage is the core of the security technology in the key management system and the entire blockchain system and is an important guarantee for account security.

Consequently, the current blockchain system requires a safe and effective blockchain account management method to protect users' property safety. The current work on the blockchain account management scheme mainly focused on the security and convenience of the private key generation, storage, and use phase. There is mainly a private key generation scheme combining random seed with a password in the user's private key generation arena, using random seed and password stored in the local device to generate private keys [13]. In the private key storage arena, the main solutions are local storage, offline storage [14], encrypted wallet, account custody [15], and hierarchical deterministic wallet [16, 17]. In the private key use arena, multi-signature [18-20] and threshold signature schemes [21,22] are mainly proposed. The multi-signature transactions require M completed signatures among N members to take effect. If someone wants to change the multi-signature transaction strategy, he needs to generate a new multi-signature transaction address and script. In the threshold signature scheme [21], the account's private key is divided into n shares and stored by n participants. When initiating a transaction, several participants greater than the threshold t are required to sign together. In the Dikshit scheme [22], different weights can be given to participants according to their identities. These threshold signature schemes can complete transactions through the joint participation of multiple people, which enhances the security and reliability of the account to a certain extent, but how to manage the keys of each participant has also become a problem. In the scheme [23], the users mix the private key of the blockchain account with the personal password for secret sharing. Each peer in the public chain stores the generated n shares, and the key recovery request is broadcasted. After the number of peers exceeding the threshold t completes the response, the user can completely recover the private key with the password.

In the scheme of combining random seed and password, the user can recover the private key through the seed and password, but in essence, the user needs to store the random seed and the password safely. The security of this scheme is still related to passwords. The multi-signature and threshold signature schemes are not suitable for individual user account management. The threshold secret sharing scheme also needs to store the user password securely. Once the user password is leaked, more than t peers can steal the user's private key through a collusion attack, and the throughput of the public chain is low. The network is prone to congestion, which does not apply to the future blockchain application environment.

In order to deal with the problem of the secure recovery of blockchain private keys, we propose a recoverable private key scheme for consortium blockchain based on verifiable secret sharing. The consortium blockchain uses the access mechanism to authenticate the user's identity, presets the committer peers, and has advantages over the public chain in terms of efficiency and flexibility. The user shares his private key using the verifiable secret sharing scheme with the committer peers, and each committer peer keeps one share. Once the private key is lost, the user can prove his identity information with the digital certificate and obtain the shares from the committer peers, and the user can confirm the correctness of the shares provided by each committer peer before reconstructing the private key. The private key can be recovered when the number of correct shares satisfies the threshold number.

2. Consortium Blockchain

The consortium blockchain is only aimed at members of a certain group and limited third parties. Multiple pre-selected peers are designated as committers. All pre-selected committer peers determine the generation of each block. Other peers can participate in the transaction, but not the committer process, and any third parties can perform limited queries through the open API of the blockchain.

The consortium blockchain has certain requirements for the configuration of consensus or verification peers and the network environment to obtain better performance. With the access mechanism, the consortium blockchain can improve transaction performance more easily and avoid problems caused by uneven participants.

The main user groups of the consortium blockchain are banks, insurance, securities, business associations, and group companies. When the blockchain was born, these companies have generally completed IT and Internet. They realized the blockchain would be very helpful further to improve the efficiency of the notarization, settlement, clearing business, and value exchange network in the industrial chain of their circle. However, when trying to use the existing blockchain technology, they found that the processing performance, privacy protection, and compliance of the blockchain could not meet their business needs. On the other hand, if these companies fully adopt Bitcoin's public chain design concept, they will subvert their existing business models and inherent interests and bear great risks. So they begin to transform the blockchain system that suits them. Consortium blockchain was born. The form of the consortium blockchain is mostly distributed ledgers. The distributed ledgers and distributed consensus of the blockchain solve the main core problem for them: the trust problem of multiple participants in the consortium.

Regarding the consensus algorithm of the consortium blockchain, the practical Byzantine fault-tolerant algorithm (PBFT) is adopted in the scheme proposed in this article. PBFT is an algorithm based on state machine copy replication that aims to solve how to ensure the consistency and correctness of the final decision even when malicious nodes exist in the entire system. Each state machine copy saves the service state and realizes the legal request of customers. In addition to transactions, it can also complete other operations and has a wide range of applications. And PBFT can still ensure the safety and liveness of the system when there are less than (n-1)/3 number of error peers in the system and correctly reach a distributed consensus.

3. Threshold Secret Sharing Scheme

In this section, we introduce two definitions of SS (Secret Sharing) and TCSS (Threshold Changeable Secret Sharing), Shamir SS scheme based on univariate polynomial, and Harn-Hsu TCSS scheme based on bivariate polynomial.

3.1 Shamir's Threshold Secret Sharing Scheme

In the Shamir secret sharing scheme [24], there are *n* shareholders $\mathrm{U} = \{\mathrm{U}_1, \mathrm{U}_2, \cdots, \mathrm{U}_\mathrm{n}\}$ and a mutually trusted dealer D. In order to share the secret *s* into *n* shares, the dealer D generates a (*t*−1) degree polynomial $f(x) \in Z_P$, where *P* is a prime number. The shared secret is $s = f(0)$, and the dealer computes the secret shares as $y_i = f(x_i)$ for $x_i \neq 0$, then send the pair (x_i, y_i) to the shareholder U_i. When reconstructing the secret, at least *t* shares$(x_i, y_i)$are needed to recover the polynomial $f'(x)$, thus each shareholder can obtain the secret $s = f'(0)$. The scheme consists of two algorithms: share generation and secret reconstruction:

3.1.1 Share Generation

The (*t*−1) degree polynomial is defined as $f(x) = a_0 + a_1x^1 + a_2x^2 + \cdots + a_{t-1}x^{t-1}$ (mod p) and $a_i \in Z_P$, for $0 \leq i \leq \mathrm{t} - 1$ and $a_{t-1} \neq 0$, the secret $s = f(0) = a_0$.

In a (*t*, *n*) secret sharing scheme, *n* points need randomly selected as x_i: $1 \leq i \leq \mathrm{n}$, and $x_i \notin Z_P$, dealer computes $y_i = f(x)$ and sends $s_i = (x_i, y_i)$ to shareholders U_i.

3.1.2 Share Reconstruction

Suppose that *m*(*m* ≥ *t*) shareholders $\mathrm{U}_1, \mathrm{U}_2, \cdots, \mathrm{U}_\mathrm{m}$ team up for secret reconstruction. Each shareholder U_i provides the share s_i to the other shareholders. After that, one shareholder has *m* shares $s_1, \cdots, s_{m-1}, s_m$ and he can use Lagrange interpolation polynomial to recover $f'(x)$ as:

$$f(x) = \sum_{i=1}^{t} s_i \prod_{j=1, j\neq i}^{t} \frac{x_j - x}{x_j - x_i} \bmod p,$$

Thus, the secret *s* can be computed as the following:

$$s = f(0) = \sum_{i=1}^{t} s_i \prod_{j=1, j\neq i}^{t} \frac{x_j}{x_j - x_i} \bmod p.$$

3.2 Harn-Hsu TCSS Scheme

In the Harn-Hsu TCSS scheme [25], there are *n* shareholders $\mathrm{U} = \{\mathrm{U}_1, \mathrm{U}_2, \cdots, \mathrm{U}_\mathrm{n}\}$ and a mutually trusted dealer D. The initial threshold is *t* and it can be increased to the exact number of shareholders who participate in secret reconstruction. This scheme consists of two algorithms: share generation and secret reconstruction.

3.2.1 Share Generation

The dealer D picks a prime number *p* and a random symmetric polynomial $F(x, y)$ with degree $t - 1$ as

$$f(x, y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{2,0}x^2 + a_{1,1}xy + a_{0,2}y^2 + \cdots + a_{t-1,0}x^{t-1} \\ + a_{t-2,0}x^{t-2}y + \cdots + a_{0,t-1}y^{t-1} (\mathrm{mod}\ p),$$

where the coefficient $a_{i,j} \in Z_p$, $a_{i,j} = a_{j,i}$, and $\forall i, j \in [0, t-1]$. The secret $s \in Z_p$ satisfies $s = F(0,0) + bF(1,1)$, where $b \in Z_p$.

The dealer D picks n different positive integers $x_1, \cdots, x_{n-1}, x_n$ from $Z_p(x_i \notin \{0,1\})$ and computes $s_i(y) = F(x_i, y)$, for $i = 1,2, \cdots$, n. Then dealer D distributes each share $s_i(y)$ to the shareholder $\mathrm{U_i}$ securely.

3.2.2 Share Generation

Suppose that $m(t \leq m \leq 1 + t(t+1)/2)$ shareholders, for example, $\mathrm{U_1}, \mathrm{U_2}, \cdots, \mathrm{U_m}$ want to recover the secret. Each shareholder $\mathrm{U_i}$ accesses the public information b and uses its share $s_i(y)$ to compute

$$w_i = s_i(0) \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i} + \mathrm{b}s_i(1) \prod_{j=1, j\neq i}^{m} \frac{x_j - 1}{x_j - x_i} \bmod p.$$

Each shareholder $\mathrm{U_i}$ sends w_i to the other shareholders. After that, a shareholder has $w_1, w_2, \cdots, w_m$ and the secret can be evaluated as

$$s = \sum_{i=1}^{m} w_i \bmod p.$$

The HARN-HSU TCSS scheme indicates that if the threshold m satisfy$(t \leq m \leq t(t+1)/2)$ in the secret reconstruction phase, the threshold can be increased from t to m. In this case, all participants must utilize valid shares to recover the secret. However, the paper [26] has employed linear subspace method to attack Harn-Hsu TCSS scheme successfully. The authors claimed that *t*+1 shares are sufficient to reconstruct the secret even if the threshold is increased beyond *t*+1. Therefore, if an illegal participant without valid share collaborates with more than *t* shareholders, it can receive enough shares to obtain the secret. Thus, the TCSS scheme does not have the threshold changeable property and is still vulnerable to illegal participant attack.

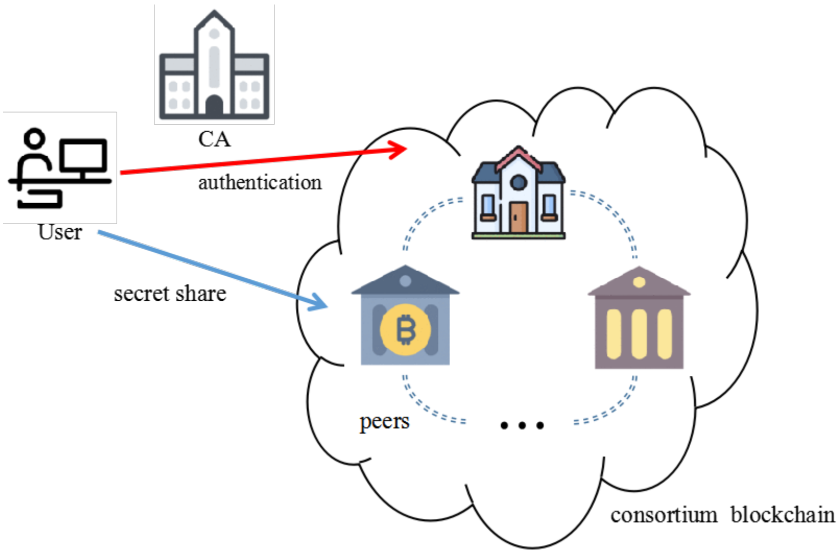

Fig. 1. Application scenario

4. Proposed Scheme

This section proposes a verifiable threshold secret sharing scheme for private key recovery in consortium blockchain.

The consortium blockchain has a user access mechanism, and users need to pass the CA authentication when registering an account on the consortium blockchain to obtain access rights. CA (Certificate Authority) is a certificate authority used to check whether the user's identity is valid and legal. Only the user who is certified by CA can trade on the blockchain. After passing the authentication, the user registers and gets a personal blockchain account, thereby holding the private key *s*, using all the user's blockchain property and the unique identifier of the user's corresponding blockchain identity. The user and all committer peers form a group to share the user's private key. **Fig. 1** shows the application scenario of the proposed scheme.

In our scheme, there are a user U, *n* ($n > t(t+1)/2$), committer peers $\mathrm{P} = \{\mathrm{P}_1, \mathrm{P}_2, \cdots, \mathrm{P}_\mathrm{n}\}$ and the threshold is *t*. The user U is both secret sharer and secret combiner, committer peers who exercise committer power in consortium blockchain are shareholders. The user U splits the secret and sends it to the committer peers. Each committer peer only holds one share of the secret-sharing and it can verify the correctness of the share he received. In the secret reconstruction phase, the user U receives the secret shares that reach the threshold t or more, and each share can be verified for correctness, confirm that all the secret shares are true and effective, then U reconstructs the secret, and finally reconstruct the original secret.

The user's biometric key SK [27] is the key information when recovering the private key. If the user needs to use SK, it can be achieved by extracting personal biometrics, without memory and additional backup. And the private key s can be recovered correctly only by using the SK in the secret recovery phase.

The notations used throughout the presentation are summarized in **Table 1**.

Table 1. Notation used throughout the scheme.

Notation	**Description**
t	Value of the threshold
n	Number of committer peers
m	Number of shares in the secret reconstruction phase
$H(.)$	SHA-256 hash function
$\|\|$	Concatenation Operation
$\oplus$	XOR Operation
SK	The biometric binding key of user
sp_i	First share generated by user
sq_i	Second share generated by user
x_i	Identification of the *i*th committer peer
SP_i	First shadow share for *i*th participant
SQ_i	Second shadow share for *i*th participant
X_i	Shadow id sent to *i*th participant

4.1 Algorithms

The proposed scheme is based on Harn-Hsu TCSS scheme [19] and its consists of three phases: share generation phase, verification phase and secret reconstruction phase.

4.1.1 Share Generation Phase

The user U selects a symmetric polynomial of the degree $t-1$ as the following

$$f(x,y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{2,0}x^2 + a_{1,1}xy + a_{0,2}y^2 + \cdots \\ + a_{t-1,0}x^{t-1} + a_{t-2,0}x^{t-2}y + \cdots + a_{0,t-1}y^{t-1} (\mathrm{mod}\, p), \tag{1}$$

where $a_{i,j} \in Z_p$, $a_{i,j} = a_{j,i}$, and $\forall i,j \in [0, \mathrm{t}-1]$. The secret $\mathrm{s} = F(0,0) + bF(1,1)$, where $\mathrm{s} \in Z_p, b \in Z_p$.

The user U computes his shares $sp_i = f_i(0) = F(x_i, 0)$ and $sq_i = f_i(1) = F(x_i, 1)$, $x_i \notin \{0,1\}$, then U uses SK to compute the shares for committer peer $\mathrm{P_i}$

- $SP_i = sp_i \oplus SK$.
- $SQ_i = sq_i \oplus H(SK) \oplus H(sp_i)$.
- $X_i = x_i \oplus H(SK) \oplus H(sq_i)$.
- U computes the verification message $V_i = H(sp_i \parallel sq_i \parallel x_i)$. Let $m_i = H(SP_i \parallel SQ_i \parallel X_i \parallel V_i)$, then use the improved the ElGamal signature to sign the plaintext m_i. Select a large prime number p and set g to be the generator of the group GF(p). The random number $l \in [1, \mathrm{p}-1]$ and $\gcd(l, p-1) = 1$. Let $y = g^l \bmod p$ as the public key, l is the private key, and computes the modulo inverse of l as $d = l^{-1} \bmod p$. (y, g, p) is the public content.
- The user signs the plaintext, selects a random number $k_i \in [1, \mathrm{p}-1], \gcd(k_i, p-1) = 1$, computes $r_i = g^{k_i} \bmod p$.
- Computes $s_i = (m_i - k_i\, r_i) d \bmod (p-1)$, the signature of m_i is (r_i, s_i).

U packages the share $(SP_i, SQ_i, X_i, V_i, r_i, s_i)$ and sends it to the committer peer $\mathrm{P_i}$.

Fig. 2 shows the share generation phase.

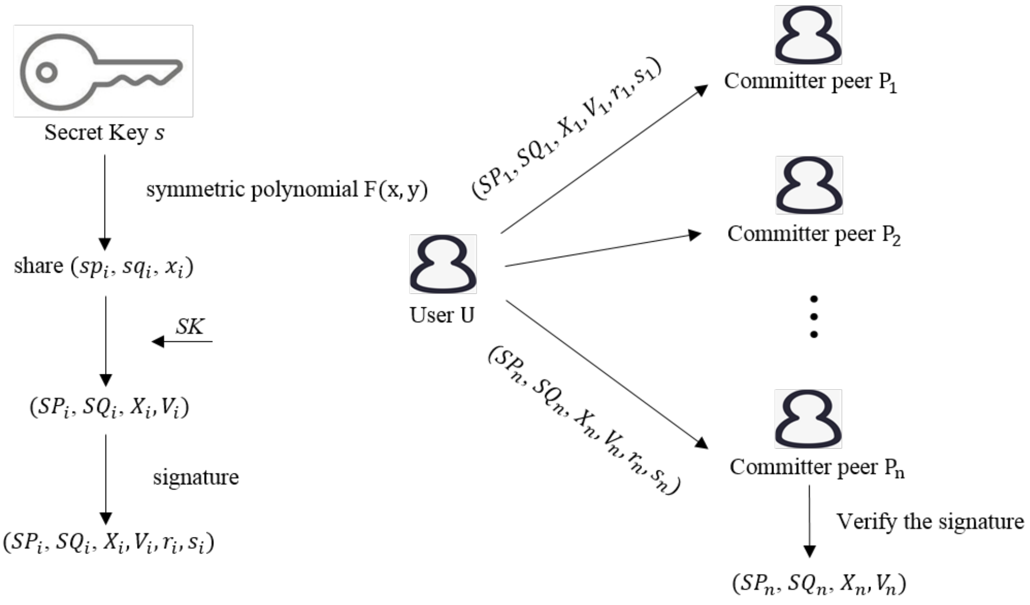
Secret Key s
symmetric polynomial F(x, y)
share (sp_i, sq_i, x_i)
SK
(SP_i, SQ_i, X_i, V_i)
signature
$(SP_i, SQ_i, X_i, V_i, r_i, s_i)$
User U
$(SP_1, SQ_1, X_1, V_1, r_1, s_1)$
$(SP_n, SQ_n, X_n, V_n, r_n, s_n)$
Committer peer $\mathrm{P_1}$
Committer peer $\mathrm{P_2}$
Committer peer $\mathrm{P_n}$
Verify the signature
(SP_n, SQ_n, X_n, V_n)

Fig. 2. Share generation phase.

4.1.2 Verification Phase

The committer peer Pi receives the share $(SP_i, SQ_i, X_i, V_i, r_i, s_i)$ sent by the user, and verifies it

- Pi computes $m'_i = H(SP_i \parallel SQ_i \parallel X_i \parallel V_i)$.
- If $y^{s_i} {r_i}^{r_i} = g^{m'_i} \bmod p$ is true, the signature is valid, and the secret share is correct.

Then Pi stores the share so that the user can get it to recover the secret.

4.1.3 Secret Reconstruction Phase

Once user U loses the private key, he needs to pass CA authentication again to get a new account to join the consortium blockchain and initiate an application to recover the secret key to the committer peers. After the committer peer confirms the user's identity, the committer peer Pi reports to the user to send the share (SP_i, SQ_i, X_i, V_i). After the user receives m $(t \le m \le 1 + t(t+1)/2)$ shares, perform a secret recovery operation

- U enters the biometric extraction key SK to compute the initial share
  $sp_i = SP_i \oplus SK$.
  $sq_i = SQ_i \oplus H(SK) \oplus H(sp_i)$.
  $x_i = X_i \oplus H(SK) \oplus H(sq_i)$.
- U computes the verification information $V'_i = H(sp_i \parallel sq_i \parallel x_i)$, if V'_i is equal to V_i, the committer peer Pi is an honest peer, and the share that Pi provides is correct and valid, otherwise Pi can be judged as malicious peer. The malicious peers will be punished after being confirmed.
- U uses the correct share (sp_i, sq_i, x_i) to compute:

$$w_i = sp_i \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i} + b sq_i \prod_{j=1, j\neq i}^{m} \frac{x_j - 1}{x_j - x_i} \bmod p. \tag{2}$$

Then enters w_i that reaches the threshold number m to recover the secret s by Lagrangian interpolation algorithm:

$$\begin{aligned} s' &= \sum_{i=1}^{m} w_i \bmod p \\ &= \sum_{i=1}^{m} sp_i \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i} + b \sum_{i=1}^{m} sq_i \prod_{j=1, j\neq i}^{m} \frac{x_j - 1}{x_j - x_i} \bmod p \\ &= \sum_{i=1}^{m} F(x_i, 0) \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i} + b \sum_{i=1}^{m} F(x_i, 1) \prod_{j=1, j\neq i}^{m} \frac{x_j - 1}{x_j - x_i} \bmod p \\ &= F(0,0) + bF(1,1) \\ &= s\ . \end{aligned} \tag{3}$$

Fig. 3 shows the secret reconstruction phase.

4.2 Peers Addition and Deletion

When a newly added user peer joins the consortium blockchain network, the (t, n) secret sharing scheme is executed for secret sharing, and his secret shares are stored at each committer peer.

If there are changes such as the addition or deletion of several committer peers, the n of the (t, n) secret sharing scheme is updated to n' which is the new number of existing committer peers in the consortium blockchain network. And users should execute the new (t', n') secret sharing scheme.

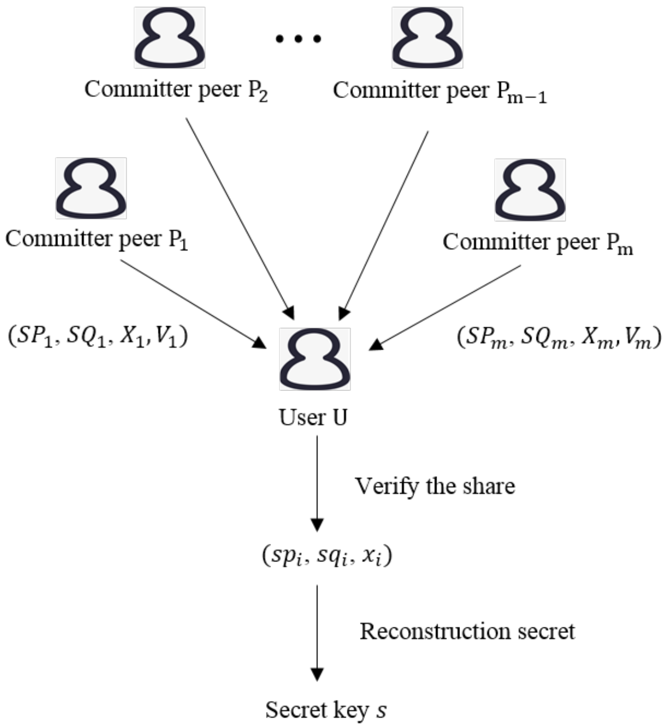

Fig. 3. Secret reconstruction phase.

5. Security Analysis and Scheme Comparison

5.1 Security Analysis

In this section, we will analyze the security to prove the robustness of the proposed scheme against some threats.

Theorem 1. Any subset of participants with *t* members cannot recover the secret.

Proof. In the (t, n) threshold scheme, at least *t* set of shares are required to reconstruct the secret. In the proposed scheme, committer peers only hold the shadow shares (SP_i, SQ_i, X_i) that are generated by U by computing $SP_i = sp_i \oplus SK$, $SQ_i = sq_i \oplus H(SK) \oplus H(sp_i)$, $X_i = x_i \oplus H(SK) \oplus H(sq_i)$. So the shadow share (SP_i, SQ_i, X_i) needs SK to retrieve (sp_i, sq_i, x_i) respectively and SK is the user's biometric extraction key. Thus, (sp_i, sq_i, x_i) can't be retrieved by any participant. Therefore, *t* participants will fail to reconstruct the secret by exchanging information.

Theorem 2. Any adversary cannot recover the secret by performing 'Man-in-the-middle attack'.

Proof. The attacker may intercept the message when the user initiates a secret recovery request to committer peers and simulates the user to apply for the secret sharing share to each peer to get (SP_i, SQ_i, X_i). However, the attacker does not have the key SK, and cannot generate the real share (sp_i, sq_i, x_i) from (SP_i, SQ_i, X_i) to complete the secret recovery which means (SP_i, SQ_i, X_i) has no meaning to the attacker.

Theorem 3. Any attacker cannot recover the secret by bribing the committer peer.

Proof. Even if the attacker bribing peers and gets the (SP_i, SQ_i, X_i) kept by committer peer, he cannot compute the (sp_i, sq_i, x_i) from (SP_i, SQ_i, X_i) to get any secret-related information about the secret due to lack of the biometric binding key SK.

Theorem 4. Illicit peers can be identified in the proposed scheme.

Proof. When the user wants to recover the secret, at least *t* committer peers need to respond and send the shares (SP_i, SQ_i, X_i), and the user utilizes SK to compute t pairs of shares sp_i, sq_i, x_i. Then the correct secret can be recovered.

Suppose that when the committer peer sends information, replace SP_i with SP'_i. The user computes the followings:

$sp'_i = SP'_i \oplus SK$, which is not equal to sp_i.

$sq'_i = SQ_i \oplus H(SK) \oplus H(sp'_i)$, and we get $sq'_i \neq sq_i$.

$x'_i = X_i \oplus H(SK) \oplus H(sq'_i)$, and we get $x'_i \neq x_i$.

The verification information $V'_i = H(sp'_i \parallel sq'_i \parallel x'_i)$ does not equal to V_i, hence the verification fails, and the peer $\mathrm{P_i}$ may judge as an illicit peer.

Suppose the share (SP_i, SQ_i, X_i) that provided by committer peer can pass the user's verification while the share has been changed, which means that the peer $\mathrm{P_i}$ has found two different numbers with the same hash value. And that is not achievable in polynomial time.

Theorem 5. Shareholders in this scheme also have verification capabilities.

Proof. In secret sharing phase, user U shares the secret, generates share content $(SP_i, SQ_i, X_i, V_i, r_i, s_i)$ and sends it to the $\mathrm{P_i}$. If there is an attacker intercepts the share during the communication, tampers the share then sends committer peer $(SP'_i, SQ'_i, X'_i, V'_i, r'_i, s'_i)$ to the committer peer $\mathrm{P_i}$.

The committer peer $\mathrm{P_i}$ receives the share and computes the plaintext content as $m'_i = H(sp'_i \parallel sq'_i \parallel x'_i \parallel V'_i)$ with the public information (y, g, q) then verifies the signature. $\mathrm{P_i}$ may find out the equation $y^{s'_i} {r'_i}^{r'_i} = g^{m'_i} \bmod p$ is not satisfied. The share is considered fake, $\mathrm{P_i}$ refuses to accept the share.

Theorem 6. Our scheme allows the user to update the biometric extraction key SK.

Proof. If the user wants to replace the key SK with SKN, computes:

$$\begin{aligned} M_1 &= SK \oplus SKN. \\ M_2 &= H(SK) \oplus H(SKN). \end{aligned} \tag{4}$$

User sends M_1 and M_2 to every committer peer and every committer peer performs the following computations:

$$\begin{aligned} SP'_i &= SP_i \oplus M_1 = sp_i \oplus SK \oplus SK \oplus SKN = sp_i \oplus SKN. \\ SQ'_i &= SQ_i \oplus M_2 = sq_i \oplus H(SK) \oplus H(sp_i) \oplus H(SK) \oplus H(SKN) = sq_i \oplus H(SKN) H(sp_i). \\ X'_i &= X_i \oplus M_2 = x_i \oplus H(SK) \oplus H(sq_i) \oplus H(SK) \oplus H(SKN) = x_i \oplus H(SKN) \oplus H(sq_i). \end{aligned} \tag{5}$$

The committer peer replaces (SP_i, SQ_i, X_i) with (SP'_i, SQ'_i, X'_i) to achieve the user's update of the biometric key. Here every user can update its key SK without let committer peers knowing about the secret and the committer peers are given relief from storing the secret.

5.2 Scheme Comparison

In this section, the proposed scheme is compared with some existing blockchain private key protection schemes that use other technologies.

Gutoski1 et. al. scheme [12] proposed a new Hierarchical deterministic wallet that solves the vulnerability that the master public key and sub-private key of the HD wallet can reversely recover the master private key. However, the HD wallet essentially relies on the secure storage of the master private key, and there are still private key security issues. Once the master private key is lost, the user cannot retrieve the property.

Goldfeder et. al. [15] proposed a threshold signature scheme compatible with bitcoins signature by using Elliptic Curve Digital Signature Algorithm providing security policy of shared control of a wallet in which each player gets only a single share. Dikshit et al. scheme [16] proposed an extend the weighted threshold ECDSA scheme. These threshold signature schemes can solve the transaction security problem of group decision-making in the blockchain, but they are not suitable for the security protection of the private key of individual users.

Han [20] proposed a new multi-signature wallet that shows better performance, storage efficiency, and privacy than existing blockchain wallets. the proposed wallet involves T-ECDSA and a Bloom-filter and does not require any modification of the blockchain protocol. However, this scheme requires information exchange and multiple encryption and decryption operations between clients, and is only suitable for multi-signature application scenarios between small groups.

We propose a verifiable secret sharing scheme, and it is applied to the consortium blockchain. The scheme achieves the secret-sharing between the user and the committer peers. Only when the user applies for the access qualification of the consortium blockchain there needs a third-party trusted center CA. In the secret recovery process were no trusted center is required to participate and the user behavior is anonymous, and the sharing and recovery of secrets are performed by the user himself. The user can verify the information submitted by the committer peers and identify illegal peers. The scheme adopts a verifiable threshold secret sharing scheme, which has the recoverability of secrets and the ability to resist single-point failures, to realize the two-way verification of user peers and committer peers against collusion attacks, and in the scheme we use the biometric extraction key of the user to achieve secret recovery, without the need to memories and store the password. The comparison of the proposed scheme with some existing methods are presented in **Table 2**.

Table 2. Scheme comparison.

Scheme	Collusion resistance	Single peer failure	Single peer control	Anonymity	Recoverability
Gutoski1's scheme	√	×	√	√	√
Dikshit's scheme	√	√	×	√	√
Han's scheme	√	√	×	√	×
Proposed scheme	√	√	√	√	√

6. Conclusion

We propose a recoverable private key scheme for consortium blockchain based on verifiable secret sharing to overcome the difficulty of recovering the lost private key of the blockchain.

Compared with the currently used mnemonic words and other methods where the user stores the private key separately to recover the key information, this solution can split the private key and share it with each committer peer, and the committer peer participates in the private key reconstruction. With the biometric encryption system, it is guaranteed that the user's private key can be recovered correctly only when the user's biometrics are held. After security analysis, our scheme satisfies the requirements of resistance to collusion attacks, man-in-the-middle attacks, and verifiable threshold secret-sharing. In terms of efficiency, the application scenario of our scheme is the consortium blockchain. Compared with the public chains, the consortium blockchain's transaction cost is cheaper and peers can complete the transaction faster.

In our future work, we will explore how to integrate the threshold secret sharing with the private key system of the blockchain more efficiently and expand the application scenarios from the consortium blockchain for more blockchain applications.

Acknowledgement

This research is partially supported by the National Natural Science Foundation of China (No. 61772166) and the Key Program of the Natural Science Foundation of Zhejiang province of China (No. LZ17F020002).

References

[1] Nakamoto Satoshi, "Bitcoin: A peer-to-peer electronic cash system," *Manubot*, Nov. 2019. Article (CrossRef Link)

[2] C. Esposito, M. Ficco, B. B. Gupta, "Blockchain-based authentication and authorization for smart city applications," *Information Processing and Management*, vol. 58, no. 2, pp. 102468, Mar. 2021. Article (CrossRef Link)

[3] A. Baldominos, JL. López-Sánchez, M. Acevedo-Aguilar, "Blockverse: A Cloud Blockchain-based Platform for Tracking in Affiliate Systems," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, no. 3. Jun. 2020. Article (CrossRef Link)

[4] H. S. Jennath, V. S. Anoop, S. Asharaf, "Blockchain for Healthcare: Securing Patient Data and Enabling Trusted Artificial Intelligence," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, pp. 15-23. Dec. 2020. Article (CrossRef Link)

[5] H. M. Kim, M. Laskowski, "Toward an ontology-driven blockchain design for supply-chain provenance," *Intelligent Systems in Accounting, Finance and Management*, vol. 25, no. 1, pp. 18–27, Mar. 2018. Article (CrossRef Link)

[6] N. K. shetri, "1 blockchain's roles in meeting key supply chain management objectives," *International Journal of Information Management*, vol. 39, pp. 80–89, Apr. 2018. Article (CrossRef Link)

[7] H. N. Dai, Z. Zheng, and Y. Zhang, "Blockchain for internet of things: A survey," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8076–8094, Oct. 2019. Article (CrossRef Link)

[8] A. Ouaddah, H. Mousannif, A. Abou Elkalam, and A. A. Ouahman, "Access control in the internet of things: Big challenges and new opportunities," *Computer Networks*, vol. 112, pp. 237–262, Jan. 2017. Article (CrossRef Link)

[9] C. Feng, K. Yu, A. K. Bashir, Y. D. Al-Otaibi, Y. Lu, S. Chen, D. Zhang, "Efficient and secure data sharing for 5G flying drones: a blockchain-enabled approach," *IEEE Network*, vol. 35, no. 1, pp. 130-137. Feb. 2021. Article (CrossRef Link)

[10] M. El Ghazouani, E. kiram, M. Ahmed, "Efficient Method Based on Blockchain Ensuring Data Integrity Auditing with Deduplication in Cloud," *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, no. 3, pp. 32-38. Sep. 2020. Article (CrossRef Link)

[11] F. Jurado, O. Delgado, Á. Ortigosa, “Tracking News Stories Using Blockchain to Guarantee their Traceability and Information Analysis,” *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, no. 3, pp. 39-46. Sep. 2020. Article (CrossRef Link)

[12] M. Sáez, “Blockchain-Enabled Platforms: Challenges and Recommendations,” *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, no. 3 pp. 73-89. Sep. 2020. Article (CrossRef Link)

[13] Y. Liu, R. Li, X. Liu, J. Wang, L. Zhang, C. Tang, and H. Kang, “An efficient method to enhance bitcoin wallet security,” in *Proc. of the 2017 11th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID)*, Xiamen. pp. 26–29, Oct. 2017. Article (CrossRef Link)

[14] S. Eskandari, J. Clark, D. Barrera, and E. Stobert, “A first look at the usability of bitcoin key management,” in *Proc. of NDSS Symposium 2015*, 2015. Article (CrossRef Link)

[15] M. Guri, “Beatcoin: Leaking private keys from air-gapped cryptocurrency wallets,” in *Proc. of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, Halifax, NS, Canada, pp. 1308-1316, Aug. 2018. Article (CrossRef Link)

[16] D. Khovratovich and J. Law, "BIP32-Ed25519: Hierarchical Deterministic Keys over a Non-linear Keyspace," in *Proc. of the 2017 IEEE European Symposium on Security and Privacy Workshops*, Paris, pp. 27-31, Apr. 2017. Article (CrossRef Link)

[17] G. Gutoski and D. Stebila, “Hierarchical deterministic bitcoin wallets that tolerate key leakage,” in *Proc. of the International Conference on Financial Cryptography and Data Security*, Springer, pp. 497–504, Jul. 2015. Article (CrossRef Link)

[18] G. Maxwell, A. Poelstra, Y. Seurin, and P. Wuille, “Simple schnorr multisignatures with applications to bitcoin,” *Designs, Codes and Cryptography*, vol. 87, no. 9, pp. 2139–2164, Feb. 2019. Article (CrossRef Link)

[19] N. Z. Aitzhan and 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, Oct. 2018. Article (CrossRef Link)

[20] J. Han, M. Song, H. Eom, Y. Son, "An efficient multi-signature wallet in blockchain using bloom filter," in *Proc. of the 36th Annual ACM Symposium on Applied Computing*, pp. 273-281. Mar. 2021. Article (CrossRef Link)

[21] D. Boneh, R. Gennaro, and S. Goldfeder, “Using level-1 homomorphic encryption to improve threshold dsa signatures for bitcoin wallet security,” in *Proc. of the International Conference on Cryptology and Information Security in Latin America*, Springer, pp. 352–377, Jul. 2017. Article (CrossRef Link)

[22] P. Dikshit and K. Singh, “Efficient weighted threshold ecdsa for securing bitcoin wallet,” in *Proc. of the 2017 ISEA Asia Security and Privacy (ISEASP)*, Surat, pp. 1–9, Feb. 2017. Article (CrossRef Link)

[23] J. Zhou and R. Qu, “Study on the healing blockchain wallet protection mechanism against conspiracy attack,” *Computer Engineering*, pp. 1–7, Apr. 2020.

[24] A. Shamir, “How to share a secret,” *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979. Article (CrossRef Link)

[25] L. Harn and C.-F. Hsu, “Dynamic threshold secret reconstruction and its application to the threshold cryptography,” *Information Processing Letters*, vol. 115, no. 11, pp. 851–857, Nov. 2015. Article (CrossRef Link)

[26] S. Jamshidpour and Z. Ahmadian, “Security analysis of a dynamic threshold secret sharing scheme using linear subspace method,” *Information Processing Letters*, vol. 163, p. 105994, Nov. 2020. Article (CrossRef Link)

[27] N. Zhang, Y.L. Zang, and J. Tian, “The integration of biometrics and cryptography-a new solution for secure identity authentication,” *Journal of Cryptologic Research*, vol. 2, no. 2, pp. 159–176, Apr. 2015. Article (CrossRef Link)

Guojia Li is currently a graduate student of the School of Cyberspace Security at Hangzhou Dianzi University. His major is cryptography. His current research interest includes threshold cryptography and blockchain technology.

Lin You is a professor of the School of Cyberspace Security at Hangzhou Dianzi University. He is the director of the Institute of Cryptography and Information Security and also the dean of Blockchain Technology Research Institute at Hangzhou Dianzi University. His research interests include cryptography, biometric recognition and blockchains. He is a member of IEEE, IACR and CACR, respectively.

Gengran Hu received the Ph.D. degree in cryptography from University of Chinese Academy of Sciences, Beijing, China. He is a lecturer of the School of Cyberspace Security at Hangzhou Dianzi University. His research interests include lattice-based cryptography, blockchain and their applications. In addition, he is a member of CACR.

Liqin Hu received the Ph.D. degree in mathematics from the Nanjing University of Aeronautics and Astronautics, Nanjing, China. She is a lecturer of the School of Cyberspace Security at Hangzhou Dianzi University. Her research interests include cryptography and coding theory.