A Voting Scheme in Blockchain Based on Threshold Group Signature

Lipeng Wang(✉), Mingsheng Hu, Zijuan Jia, Bei Gong(✉),
Yanyan Yang, Xinxin Liu, and Wenjun Cui

School of Information Science and Technology, Zhengzhou Normal University,
Zhengzhou 450044, China
wlpscu@126.com, tekkman_blade@126.com

Abstract. Traditional voting schemes are used for the credit evaluation and authentication. During the voting process, the contents need to be verified through the signature algorithms. Traditional signature schemes for voting scenes exist several drawbacks such as distrust of central nodes for the group signature and inefficiency for the ring signature. A trusted center selection scheme is proposed based on Dynamic Bayesian Network, which can be adapted in the isomerized blockchain. By introducing the historical interaction window, the aging factor, and the penalty factor, the adaptive trusted metrics can be obtained through aggregating the direct credibility and the indirect credibility. A new threshold group signature scheme is introduced through collaboration between users and the trusted centers. In order to protect the user identities, the blinding process is proposed. In case of compromising, the trusted centers create redundant backup, and can be updated with the proposed selection scheme. Security analysis shows that the proposed signature, whose difficulty is equivalent to the discrete logarithm of the elliptic curve, achieves a high level of anonymity and can resist impersonation attacks. Computational complexity analysis shows that the new method with low computational cost and transmission efficiency can be effectively adapted to the isomerized blockchain scene.

Keywords: Blockchain · Confidential computation
Threshold group signature · Dynamic Bayesian Network

1 Introduction

Blockchain is a distributed database technology which can record transaction. The characteristics of blockchain are "decentralization", "anonymization" and "de-trusted" [1]. It handles distrust among nodes and has been widely applied in the fields of e-money, financial investment, internet of things, medical care, and energy network [2]. The blockchain falls into three categories: public chain, permissioned chain, and private chain. At present blockchain-based voting schemes existing on permissioned chain and private chain have been used in credit assessment and decision making. The paper is devoted to develop a safe and transparent online voting mechanism on the blockchain: coordinating participants to ensure the fairness, allowing the voting manager to check the voting result and cancel illegal ballots [3]. Compared with existing voting

H. Zhang et al. (Eds.): CTCIS 2018, CCIS 960, pp. 184–202, 2019.
https://doi.org/10.1007/978-981-13-5913-2_12

schemes, blockchain-based voting methods have the ability of being irrevocable and non-repudiation, and the process can be automated in accordance with the statute, without human interference. Applications based on the blockchain with neutrality and security have shown the wide prospects.

The identities in the blockchain node, similar to the bankcard account number, are the pseudonym information used by the user to participate in the voting process. The temporary identify labels are generated by the participants using the public key encryption algorithm with anonymity. Recent studies [4, 5] have shown that the blockchain has the risk of identity leakage. The attackers exploit the message propagation vulnerabilities and trace out the initiator identities. It means that the native blockchain architecture cannot guarantee the anonymity. How to design a highly anonymous and non-repudiation blockchain architecture is the focus of this study.

To implement the voting scheme based on the group signature in blockchain, the native blockchain architecture needs to be redesigned to pick out the trusted centers. However, once the trusted centers are breached, the user secrets will be stolen. What's more, attackers can tamper with the voting results. How to guarantee the security of the trusted centers in the isomerized blockchain is the prerequisite for the voting scheme.

To select the trusted centers on the blockchain, the metrics must be updated in order to feed back the credibility to other nodes in time. When initiating, all nodes need to generate their own private keys to calculate the share signatures which will be passed to the trusted centers. The trusted centers will accumulate the share signatures to composite the group signature. To trace the abnormal behaviors, the trusted centers must be allowed to open the signature and locate the corresponding user identities. When some nodes are unavailable, their signatures are allowed to revoke. How to design such a revocable and traceable threshold group signature scheme is the main contribution of this study.

2 Related Researches

In 1994, Marsh firstly proposed the concept of trusted computing and elaborated on the canonical representation of trust metrics [6]. The key of implementing the signature scheme in blockchain is selection of the trusted centers.

To solve the problems that the user identities on general devices are unsafe, [7] builds a mathematical model based on the D-S evidence theory and proposes the trustworthiness metrics in Ad-hoc and P2P networks. [8] proposes a dynamic trusted model based on Bayesian Network. The credibility metrics depend on the neighbors with the dynamic objective variables. However, the model still needs to be optimized in terms of prediction accuracy and time continuity. [9] quantifies the credibility based on the information theory and measures the trust propagation path in the Ad-hoc network. Since the degree of trust is regarded as the uncertainty, the entropy is introduced to quantify it. [10] proposes the trusted security discovery model TSSD in pervasive circumstances. The model is a composite architecture that supports discovery functions in both secure and insecure scenarios. [11] proposes a penalty-based incentive mechanism based on the repeated game, which can adjust the penalty according to the reputation.

Due to heterogeneity and complexity of the blockchain, the trusted metrics need to meet the requirements of dynamic adaptability, high efficiency, and strong timeliness. When adapting the traditional measurements to the blockchain, existing schemes mainly include several drawbacks [12]:

1. Assumptions made by those methods limit their applications.
2. The computing power of blockchain nodes is uneven, which leads existing methods to be inferior in terms of interaction timeliness and adaptability.
3. Autonomy of blockchain makes the node behaviors unpredictable. Protection methods without cooperation among nodes make them vulnerable when facing up to the attacks.
4. Existing researches lack robustness whose accuracy will be declined when adapting to the blockchain.

The threshold group signature schemes, according to the way of key distribution, are mainly divided into the ones with the trusted center and the ones without the trusted center. For the group signature scheme without the trusted center, each node has a high degree of autonomy, which enhances the security level with the cost of increasing computation. The trusted center for the threshold group signature scheme is considered as the management node and can trace user identities which is promising for the blockchain based voting scenes.

Traditional signature schemes based on the large number decomposition and discrete logarithm problems cannot meet the requirements of low computation and bandwidth required in the blockchain. The group signatures scheme based on the elliptic curve with lower resource consumption are more suitable for the blockchain scenarios [13].

In 2004, Tzer-Shyong et al. [14] integrated the short key feature with a threshold method and proposed a signature scheme (YC scheme) based on the elliptic curve encryption algorithm. However, it lacks the operations of tracking and cancellation of the signatures. [15] points out that arbitrary member conspiracy in the YC scheme can be able to obtain the private keys. Therefore, [16] proposes an ECC threshold signature scheme with the elliptic curve discrete logarithm difficulty, which could track the signer identity with the help of the trusted center and can effectively resist the members conspiring to forge the signature. The above threshold group signatures are based on the Shamir secret sharing scheme. Some other secret sharing methods will be discussed later.

In [17], a threshold group signature scheme is proposed based on the bilinear mapping and secret sharing. The keys are distributed in the members of the signature set. In [18], the threshold group signature scheme with the elliptic curve discrete logarithm difficulty is proposed, including signature verification, thresholds modification, etc. [19] proposes a threshold group signature scheme based on ECC scheme, which has a shorter key length, lower computation and bandwidth requirement. [20] and [21] propose the ECDSA threshold signature schemes, whose participants can reconstruct keys. Goldfeder et al. proposes a threshold signature scheme to realize the multi-party control functions on bitcoin [22], which applies the threshold cryptography for key management. However, when to recover the keys, multiple parties must be present.

To address the above problems, the paper redesigns the native blockchain, and selects the trusted centers based on Dynamic Bayesian Network, which correlates the time factor to perform the dynamic trusted metrics. In addition, this paper proposes a threshold group signature scheme. Through cooperation between users and the trusted centers, the share signatures will be synthesized to form the group signature. In order to protect the user identities, the proposed method will blind the user identities. In case of the trusted centers unavailable, the new scheme will back up the user signatures and update the trusted centers dynamically.

3 Background

3.1 Digital Signature

Digital signature is a password protection scheme, which utilizes cryptography to confirm source and integrity of the data. It is mainly used in asymmetric key encryption and digital summarization. The sender first signs the message, then others verify it. The signature is temporally valid for a single process. The steps are as follows [23]:

1. $G(p)\overset{generate-keys}{\longrightarrow}(sk,pk)$, where *sk* is the private key and *pk* is the public key.
2. $S(sk,m)\overset{generate-signatures}{\longrightarrow}sig$, where *m* is the plaintext, and *sig* is the result signature.
3. $Verify(pk,m,sig)\overset{verrify-signatures}{\longrightarrow}\{True,False\}$, which verifies integrity of the data through the public key, the plaintext, and the signature.

The typical signature schemes mainly include the elliptic curve digital signature [24] and the partial blind signature [25]. The elliptic curve digital signature will be the focus in the paper.

The elliptic curve digital signature algorithm is mainly designed based on the elliptic curve discrete logarithm. The elliptic curve E on the finite field $\mathrm{F_q}$ is expressed as [24]:

$$y^2 = x^3 + ax + b(\bmod q),$$

where $a,b,x,y \in \mathrm{F_q}$. Assuming $\mathrm{E(F_q)}$ represents the point set of the elliptic curve, and the base point is $\mathrm{G} \in \mathrm{E(F_q)}$, $|G| = n$. The parameters are $(\mathrm{F_q},\ a,\ b,\ \mathrm{G}, n)$. The steps of the elliptic curve algorithm are as follows:

1. d is a randomly selected value, where $\{d|d \in Z, d \in [1,n-1]\}$. $\mathrm{Q} = d\mathrm{G}$, where the public key is Q, and the private key is d.
2. Randomly select $k \in [1,n-1]$ to obtain $k\mathrm{G} = (x,y)$.
3. $\mathrm{r} = x$ mod n, and verify $r = 0$. If it holds, return to step 2.
4. $e = \mathrm{H(m)}$, where $\mathrm{H}(\cdot)$ is a custom hash function.
5. $s = k^{-1}(e + dr)$ mod n, if $s = 0$, go back to the step 2 to continue, otherwise return the result (r,s).

6. The formula to verify the signature (r, s) with the plaintext m and the public key G is $s^{-1}(eG + Qr) = k(e + dr)^{-1}(eG + dGr) = kG$. If it does not hold, it means that the message may be tampered.

3.2 Secret Sharing Scheme

The concept of secret sharing was first proposed by Shamir [26] and Blackey [27]. The idea is to split the secret into N copies and send each copy to different participants for management. When to recover the secret, it needs to involve a certain number of participants, whose number must be greater than or equal to the specified threshold.

The classic secret sharing scheme is the Shamir sharing scheme. The scheme firstly divides the secret S into n copies. Any k copies can recover S, and $k - 1$ ones could not. The whole process is divided into three steps:

1. Initialization: Assuming n participants $(\mathrm{P}_1 \ldots .\mathrm{P}_n)$ with the threshold k, let p be a prime number, the trusted center coding range be the finite field GF(p), and each participant be $x_i \in \mathrm{GF(p)}(i = 1, 2, \ldots .n)$.
2. Encryption: The trusted center selects a $k-1$ order polynomial $f(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{k-1}x^{k-1}$, where $a_i \in \mathrm{GF}(P)(i = 1, 2, \ldots, k - 1)$, $a_0 = S$. Put $x_i \in \mathrm{GF(p)}(i = 1, 2, \ldots .n)$ into the above equation to obtain $(x_1,f(x_1)), \ldots, (x_n,f(x_n))$ which will be sent to each participant.
3. Decryption: k pairs of $(x,f(x))$ will be optionally collected from n participants to reconstruct the polynomial $f(x)$, from which we can obtain $f(0) = a_0 = S$ with Lagrange interpolation.

3.3 Threshold Signature Scheme

The (t, n) threshold signature algorithm mainly includes three steps, which are key generation, signing and verification. For n members, the threshold key generation step returns the public key pk and the private key $sk_i\{1 \leq i \leq n\}$ with the system parameter I. The threshold signature process takes on the input m and I with the private key sk_i, and outputs the signature σ [28]. The threshold verification process can be able to verify the signature σ with the public key pk.

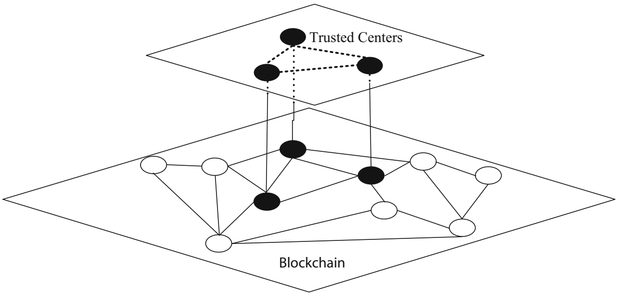
The threshold signature algorithm is considered as safe when meeting two requirements [29]:

1. Unforgeability: Given the system parameter I, an attacker can destroy up to a maximum of $t - 1$ nodes through issuing at most k questions, where k is generally equal to 2^n times the length of the message.
2. Robustness: When the attacker breaks through up to $t - 1$ nodes, the correct signature can still be generated.

4 Trusted Center Selection Criterion

The threshold group signature scheme needs the trusted center to trace user identities. Its corresponding measurements must obey several principles which will be discussed later. This paper proposes a trusted center selection scheme based on Dynamic Bayesian Network. The new scheme introduces the historical interaction window, the aging factor, the penalty factor, and aggregates the direct and indirect credibility to form a set of dynamic and adaptive trusted metrics. When the reliability for the current centers is declined to a certain level, they will be re-selected as soon as possible. This paper adopts the selection scheme to the blockchain as shown in Table 1, which reduces the computational complexity and is proved to be equivalent to [12].

Table 1. Diagram of trusted centers construction



4.1 Definitions

Definition 1: Assuming $C = \{c_i | i \in N^*, 1 \le i \le n\}$ represents a set of network nodes. For the current node c, if $\forall c_i \in C$, $c \ne c_i$, the interactive factor r_i exists to make $r_i(c, c_i)$ not empty, and the set $R = \{r_1, r_2, \ldots, r_n\}$ is the local relational database.

Definition 2: For the local relational database $R = \{r_1, r_2, \ldots, r_n\}$, if a relational window w exists to make $R_w = \{r_{n-w}, r_{n-w-1}, \ldots, r_n\}$ be the trusted metrics for the target entity, w is said to be valid.

Definition 3: For the valid relationship window w, at time t, the corresponding context is denoted as $W(w, t)$.

Definition 4: With n trust levels, the credibility set of the current network is denoted as $L = \{l_1, l_2, \ldots, l_n\}$ where $i < j$, $l_i < l_j$. The greater l, the higher the credibility.

Definition 5: Assuming $X = \{x_1[t], x_2[t], \ldots x_n[t]\}$ denotes the attribute set at time t, and $x[t]$ denotes a serial of values for the random variable $x_i[t]$. Assuming continuous events occur at discrete time and the attribute evolving process satisfies the Markov chain model, that is $P(x[t+1]|x[0], \ldots x[t]) = P(x[t+1]|x[t])$, the resulting network is called Dynamic Bayesian Network.

Definition 6: Let the priori network be N_0 and the transfer network be N. N_0 is the distribution of $x[0]$, and N represents the transition probability $P(x[t+1]|x[t])$.

4.2 Trusted Metrics

For the credibility measurement, this paper adopts a strategy combining the static and dynamic behaviors to perform the trusted authentication, which considers the objective and subjective factors to make the comprehensive measurement. The proposed method tries to minimize the resource requirements when applied to the blockchain.

The static trusted metrics are the basis which includes the operating system integrity, additional protection, and hardware performance. The nodes in the blockchain who own more computing resources are able to carry out more tasks, so a high credibility will be assigned to them. The static measurement of this paper does not adopt the traditional identity authentication metric because of anonymity of the blockchain.

The dynamic trusted metrics provide the fine-grained quantitative standards for the measurement in the paper. Through real-time monitoring the node status and timely feedback to others, real-time credibility of the trusted center is ensured. The dynamic behavior metrics mainly include mining efficiency, abnormal connection attempts, and node reliability. Because each node in the blockchain needs to communicate with others during a certain time interval, the paper introduces the dynamic behavior assessment scheduling to the original communication frequency which can effectively utilize the system resource. The evaluation indicators are shown in Fig. 1.

When a new node attempts to join the blockchain, its credibility needs measurement. This paper considers the trustworthiness as a measure of the satisfaction degree for the task. The measurement relies on the correlation among nodes with a combination of subjective and objective factors. The proposed method first initializes the measurement module, calculates the direct credibility (Dc) and the indirect credibility (Ic), then obtains the comprehensive credibility Sc based on Dc and Ic.

Definition 7: Assuming $\delta'(t)$ is the aging factor, which represents the attenuation factor of the credibility with time. The formula is:

$$\delta'(t) = 1 - \frac{\Delta t \times \zeta'}{t - t_0},$$

where $\zeta' \in (0,1)$, $\delta'(t) \in (0,1)$, t_0 denotes the start time, and t represents the current time. ζ' is used to adjust the decay rate. As the value increases, the decay rate will become faster. Δt is the interval between two operations.

Theorem 1: In the blockchain scenario, $\delta'(t)$ is equivalent to $\delta(t)$, and $\delta(t)$ is:

$$\delta(t) = 1 - \frac{\zeta}{t - t_0},$$

where $\zeta \in (0,1)$, $\delta(t) \in (0,1)$.

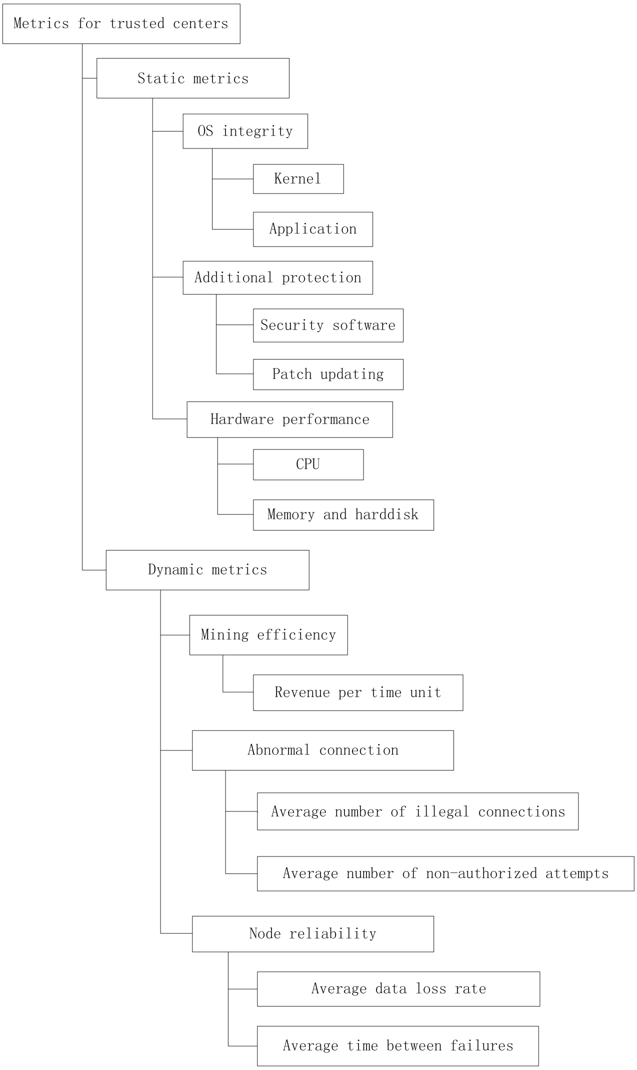


Fig. 1. Metrics for selection of the trusted centers

Proof: In the blockchain scenario, nodes in the network send heartbeat packets at a certain interval T_h. In order to make full use of resources, the time interval to update the aging factor can be set as $\Delta t = T_h$. Due to the fluctuation of the network, there is a certain disturbance during the process of message transmission. Assuming Δt obeys the normal distribution: $\Delta t \sim N(T_h, \sigma^2)$, at intervals $\Delta t_1, \Delta t_2, \ldots .\Delta t_n$ for n entities in the blockchain, we obtain $E(\Delta t_i) = T_h$ and $D(\Delta t_i) = \sigma^2$. For any real number x, we can obtain

$$\lim_{n\to\infty} P\left\{ \frac{\sum_{i=1}^{n} \Delta t_i - nT_h}{\sqrt{n}\sigma} \le x \right\} = \Phi(x),$$

where $E(\Phi(x)) = 1$.

For the blockchain scenario, the node number n is large and Δt can be approximated as a fixed value. Assuming $\zeta = \Delta t \times \zeta'$, we can obtain $\delta(t) = 1 - \frac{\zeta}{t-t_0}$.

Definition 8: Let the penalty factor be ψ with the formula:

$$\psi = \begin{cases} 1, \Delta Dc_n \ge 0 \\ 0 < \psi < 1, \Delta Dc_n < 0 \end{cases},$$

where $\psi \in (0, 1)$. Dc_n is the direct credibility of the n-th entity, where $\Delta Dc_n = Dc_n(\mathrm{t}) - Dc_n(\mathrm{t}-1)$. When $\Delta Dc_n < 0$, it indicates that the current entity's direct credibility drops down, and needs to reduce ψ to increase the punishment, which can lower the entity's credibility to a certain value in order to remove the node from the trusted set as soon as possible.

Definition 9: Let the target node be x_j. $\forall x_i \in X$, the corresponding relationship between x_i and x_j in the window W at time t corresponds to the context condition $\mathrm{W}(\mathrm{w}, t)$. w represents the correlation between $\mathrm{x_i}$ and x_j in the valid relationship window. The direct credibility is:

$$Dc_n(\mathrm{x_i}, \mathrm{x_j}, \mathrm{W}(\mathrm{w}, \mathrm{t}), \mathrm{t}) = \begin{cases} \left(\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right), t = 0 \\ Dc_n(\mathrm{x_i}, \mathrm{x_j}, \mathrm{W}(\mathrm{w}, \mathrm{t}-1), \mathrm{t}-1) \times \delta(t), \mathrm{W}(\mathrm{w}, \mathrm{t}) - \mathrm{W}(\mathrm{w}, \mathrm{t}-1) = \Phi, \\ Dc_n(\mathrm{x_i}, \mathrm{x_j}, \mathrm{W}(\mathrm{w}, \mathrm{t}), \mathrm{t}) \times \Psi, \textit{ else} \end{cases}$$

where $\sum_{i=1}^{n} Dc_i = 1$.

When initializing the direct credibility, the credibility for target node's neighbors are set to be the same value. If there is no change at a certain interval, the node's credibility decays with time. When the increment of the direct credibility is positive, the penalty factor $\psi = 1$, and the direct credibility remains unchanged. Otherwise, the direct credibility needs to be punished.

Definition 10: $\forall x_i \in X$, let the target node be x_j, and the context condition at time t for the node x_i and x_j within the valid window w be $W(w,t)$. We can obtain the indirect credibility for the target node with the formula:

$$Ic_n\left(\mathrm{x_i},\ \mathrm{x_j},\ \mathrm{W(w,t),t}\right)=\begin{cases} \left(\frac{1}{n},\frac{1}{n},\ldots,\frac{1}{n}\right), t=0 \\ Ic_n(\mathrm{x_i},\ \mathrm{x_j}, \mathrm{W(w,t-1)},\ \mathrm{t-1}) \times \delta(t),\ \mathrm{W(w,t)\ \text{-}\ W(w,t-1)} = \Phi \\ \frac{\sum_{Z\in X} Dc_n(\mathrm{x_i},\mathrm{x}_k,\mathrm{W(w,t\text{-}1),t\text{-}1}) \times Sc_n(\mathrm{x}_k,\mathrm{x}_j,\mathrm{W(w,t\text{-}1),t\text{-}1})}{\sum_{Z\in X} Dc_n(\mathrm{x_i},\mathrm{x}_k,\mathrm{W(w,t\text{-}1),t\text{-}1})} \times \Psi, else \end{cases},$$

where Z represents neighbors of the target node. The indirect credibility of its neighbors is initialized as $\frac{1}{n}$. For the subsequent process, if the target node does not generate a new context condition, the indirect credibility decays over time. Conversely, we can obtain the target node credibility combining the direct credibility and the comprehensive credibility of the target node and its neighbors. When the increment of the indirect credibility is positive, the penalty factor $\psi = 1$, and the indirect credibility remains unchanged. Otherwise, the indirect credibility of the target node needs to be punished.

Definition 11: Let μ be the weight factor for the direct credibility with the formula:

$$\mu = 0.5 + \frac{\mathrm{w}}{2\Omega},$$

where Ω is the length of the valid relationship window, $\mu \in (0.5, 1]$. μ is used to adjust the weight between the direct credibility and the indirect credibility when calculating the comprehensive credibility. Since its value is always greater than 0.5, it means the priority will be assigned to the direct credibility.

Definition 12: $\forall x_i \in X$, let the target node be x_j, the corresponding context condition of x_i and x_j is $\mathrm{W}(\mathrm{w},t)$. The comprehensive credibility is:

$$Sc_n\left(\mathrm{x_i},\ \mathrm{x_j},\ \mathrm{W(w,t),t}\right) = \begin{cases} Ic_n(\mathrm{x_i},\mathrm{x_j},\mathrm{W(w,t),t}), \mathrm{W(w,t)} = \Phi \\ Dc_n(\mathrm{x_i},\ \mathrm{x_j},\mathrm{W(w,t),t}), \mathrm{W(w,t)} = \mathrm{W}(\Omega,\ \mathrm{t}) \\ Sc_n(\mathrm{x_i},\ \mathrm{x_j},\ \mathrm{W(w,t-1),t-1}) \times \delta(t), \mathrm{W(w,t)} - \mathrm{W(w,t-1)} = \Phi \\ [\mu \times Dc_n(\mathrm{x_i},\ \mathrm{x_j},\ \mathrm{W(w,t),t})\ +\ (1-\mu)Ic_n(\mathrm{x_i},\ \mathrm{x_j},\ \mathrm{W(w,t),t})] \times \Psi, else \end{cases}.$$

The comprehensive credibility is considered as the ultimate credibility of the target node. If no change of context condition between the time $t-1$ and t, the comprehensive credibility decays. Otherwise the priority will be given to the direct credibility compared with the indirect credibility. If the increment is negative, the comprehensive credibility will be punished.

5 Blockchain Based Threshold Group Signature Scheme

The participants for the proposed scheme mainly include the blockchain nodes (US), the trusted centers (TC), and the backup node (BK), where the trusted centers are retrieved through Dynamic Bayesian Network as above. The proposed threshold group signature scheme adapts the signature to the blockchain scenario, which blinds user identities, and improves the effectiveness with security guarantees. In addition, the new scheme adds the backup operation to avoid the corruption of the trusted centers.

As shown in Fig. 2, the blockchain based threshold group signature is divided to eight steps which comprise selecting the trusted centers, registration, generation of the share signature, synthesizing the group signature, signature verification, signature backup, signature openness, and signature revoking. They will be discussed later.

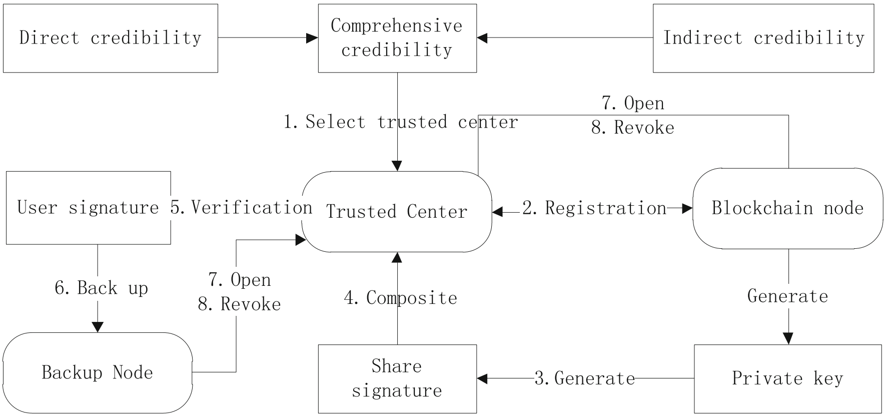


Fig. 2. Architecture diagram of the proposed scheme

For the convenience, the symbols are defined as Table 2.

Table 2. Symbols of the proposed scheme

Symbol	Description	Symbol	Description
TC	trusted centers	Id_i	identity of i th node
N_i	i th node	u_s	private key for users
c_s	private group key	u_p	public key for users
c_p	public group key	P	set comprising t members
TC_s	private key of trusted center	Id	set comprising t member identities
TC_p	public key of trusted center	UL	the user signatures
ID_i	blinded identity of i th node		

1. Select the trusted centers
   When the blockchain is initializing, the trusted center is selected based on Dynamic Bayesian Network as above.
2. Registration
   The parameters of the proposed threshold group signature scheme need to be set to generate the corresponding keys and hash functions. When a node attempts to join the blockchain, it needs to interact with the trusted center, blind the identities, and verify the keys.

First, select the appropriate parameters to generate the elliptic curve. p is a large prime number and F_p is a finite field. Select a and b at random, where $a, b \in F_p$, to construct the non-singular elliptic curve: $y^2 = x^3 + ax + b$, $4a^3 + 27b^2 \neq 0(\mathrm{mod}\, p)$, where G represents generators, $ord(G) = \Upsilon$, and Υ represents a large prime.

If the private key of the trusted center $\mathrm{TC_s} = s$ has been decided, its public key is $\mathrm{TC_p} = sG$. The signature is based on the Shamir secret sharing scheme with the polynomial:$f(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{t-1}x^{t-1}$, where $a_i \in \mathrm{GF}(p)(i = 1, 2, \ldots, t-1)$. Set the private group key as $c_s = a_0 = f(0)$ and the public group key as $c_p = c_sG = f(0)G = a_0G$. Select a one-way hash function $h(x)$ to blind the user identities.

$< a, b, G, c_p, p, h(x) >$ are denoted as the public information with global accessibility in the proposed scheme. $< c_s, f(x) >$ serves as the confidential information, which is stored on the trusted center and backed up to prevent the central node from malfunctioning or being offline to affect the availability of the blockchain.

When a certain node i attempts to join the blockchain, it needs to apply to the trusted centers, including identities of verification and blinding. The trusted centers then issue the partial key which will be sent to the corresponding user node after being encrypted. The specific process is as follows:

- The node sends the randomly generated partial key $u \in Z_p^*$ and its own identity Id_i to the trusted center. The trusted center then obtains $X_i = uG$ according to the value u of the node i. The trusted center searches the user signatures database to determine whether the node has joined the blockchain or not. If it does, the trusted center rejects its application, otherwise the trusted center obtains $U = uG = (x_u, y_u)$, $ID_i = (x_u + s)h(Id_i) + u \bmod p$ where ID_i is the blinded user identity and $< U, ID_i >$ will be sent to the node i.
- After the node i receives $< U, ID_i >$, it needs to verify $ID_iG = (x_uG + TC_p)h(Id_i) + U$. If it holds, the node is required to resend its application. Otherwise its private key will be set as $x_i = u$, $X_i = x_iG$, and $< X_i, U, ID_i, Id_i >$ will be sent to the trusted center.
- When receiving $< X_i, U, ID_i, Id_i >$, the trusted center verifies $ID_iG = (x_uG + TC_p)h(Id_i) + X_i$. If it does not hold, the node will be refused to join the blockchain, otherwise it can allow to enter the network and add itself to UL.
- The trusted center obtains another part of the private key for the node $i : y_i = f(ID_i)$, and y_i will be sent to the corresponding node via a secret channel. a_iG will be broadcasted in the network by the trust centers. After the user receives y_i, it needs to verify $y_iG = Gf(\mathrm{ID_i}) = \sum_{i=0}^{t-1} a_iGID_i^i$. If it does not hold, the trusted center needs to re-execute this step.
- The private key for node i is $u_s = x_i + y_i$ and the public key is $u_p = u_sG$. Blinded identity ID_i of the node i and the public key u_p are broadcasted to the blockchain.

1. Generate the share signature
   Assuming the set of participants as $\Lambda = \{\mathrm{N_1}, \mathrm{N_2}, \ldots, \mathrm{N_t}\}$, with corresponding blinded identities as $ID = \{\mathrm{ID_1}, \mathrm{ID_2}, \ldots, \mathrm{ID_t}\}$. First, the node i randomly selects $k_i \in Z_p^*$, then obtains $r_i = k_iG = (x_{ri}, y_{ri})$ and the hash value $z = h(m)$ of the

message m. The node can obtain its share signature $s_i = k_i x_{ri} - z u_s I_i \bmod p$ where $I_i = \prod_{i \neq j} \frac{ID_i}{ID_i - ID_j}$. It means that the share signature of the node i is (r_i, s_i), which will be sent to the trusted center through a secret channel.

2. Synthesize the group signature
   After the trusted center receives the share signature (r_i, s_i), it should be verified. Calculate $I_i = \prod_{i \neq j} \frac{ID_i}{ID_i - ID_j}$ and z with the member set Λ and its corresponding identities $ID = \{ID_1, ID_2, \ldots, ID_{\mathrm{t}}\}$. Verify $s_i G + z u_p I_i = r_i x_{ri}$. If it holds, it shows that the share signature is legal, otherwise return to the step 3 to continue.
   The way to generate the threshold group signature is as follows: First we can obtain $R = \sum_{i=1}^{t} r_i x_{ri} \bmod p$, and then merge the share signatures with $S = \sum_{i=1}^{t} s_i$ to get $W = \sum_{i=1}^{t} I_i X_i$. (R, S) is the final threshold group signature. Add the signature $< r_i, s_{i,} ID_i >$ to UL which can allow to open the signatures.
3. Verify the signature
   The trusted center obtains (R, S) which needs to be verified. The verification steps are as follows: First, calculate $z = h(m)$ and verify $SG + z(c_p + W){=}R$. If it does not hold, reject the signature; otherwise, perform the backup operation.
4. Signature backup
   In order to prevent the user signatures from tampering, it needs to select the sub-optimal node as backup which can be obtained with the above scheme based on Dynamic Bayesian Network to duplicate the user signatures.
5. Open the signature
   It needs to connect the trusted centers to perform the open operation. For the signature (R, S), the trusted centers first searches $< r_i, s_i, ID_i >$, then locates the target identity from $< X_i, Id_i, ID_i >$ indexed by ID_i.
6. Revoke the signature
   When a node in the blockchain leaves the network, the trusted center is required to delete its information. First, reinitialize $f(x) = a_0 + \alpha_1 x + \mathbf{a}_2 x^2 + \ldots + \mathbf{a}_{t-1} x^{t-1}$, where a_0 remains unchanged. Then we can obtain the partial key $y_i = f(ID_i)$, re-execute from step 2 to determine whether it needs to receive y_i.

6 Security Analysis

6.1 Correctness Analysis

Theorem 2: When the node i is being registered to the trusted center, two authentication equations $ID_i G = (x_u G + TC_p) h(Id_i) + U$ and $ID_i G = (x_u G + TC_p) h(Id_i) + X_i$ hold.
Proof: Since $U = uG = (x_u, y_u)$, $ID_i = (x_u + s) h(Id_i) + u$ and $\mathrm{TC_p} = sG$, we can obtain $ID_i G = ((x_u + s) h(Id_i) + u) G = (x_u G + sG) h(Id_i) + uG = (x_u G + \mathrm{TC}_p) h(Id_i) + U$.

Since $x_i = u$, $X_i = x_iG$, we can obtain $ID_iG = ((x_u + s)h(Id_i) + u)G = (x_uG + \mathrm{TC}_p)$ $h(Id_i) + X_i$.

Theorem 3: The equation $y_iG = Gf(\mathrm{ID_i}) = \sum_{i=0}^{t-1} a_iGID_i^i$ holds which is used to verify the key of the trusted center by node i.
Proof: Since $y_i = f(ID_i)$, $f(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{t-1}x^{t-1}$, we can obtain $y_iG = Gf(ID_\mathrm{i}) = a_0G + a_1GID_i + \ldots + a_{t-1}GID_i^{t-1} = \sum_{i=0}^{t-1} a_iGID_i^i$.

Theorem 4: When synthesizing the group signature, the verification formula $s_iG + zu_pI_i = r_ix_{ri}$ holds.
Proof: Since $s_i = k_ix_{ri} - zu_sI_i$, $u_p = u_sG$, $r_i = k_iG = (x_{ri}, y_{ri})$, we can obtain $s_iG = k_ix_{ri}G - zu_sI_iG = r_ix_{ri} - zu_pI_i$ and $s_iG + zu_pI_i = r_ix_{ri}$.

Theorem 5: When the trusted center obtains the signature (R, S), the verification equation $SG + z(c_p + W) = R$ holds.
Proof: Since $S = \sum_{i=1}^{t} s_i$, $s_i = k_ix_{ri} - zu_sI_i$, we obtain $SG = \sum_{i=1}^{t} s_iG = \sum_{i=1}^{t} (k_ix_{ri}G - zu_sI_iG)$. As $r_i = k_iG$, $u_s = x_i + y_i$, $y_i = f(ID_i)$, $R = \sum_{i=1}^{t} r_ix_{ri}$, we infer $SG = \sum_{i=1}^{t} s_iG = \sum_{i=1}^{t} (r_ix_{ri}) - \sum_{i=1}^{t} \{z(x_i + f(ID_i))I_iG\} = R - \sum_{i=1}^{t} (zx_iI_iG + zf(ID_i)I_iG)$. Since $X_i = x_iG$, we get $SG = R - \sum_{i=1}^{t} (zX_iI_i + zf(ID_i)I_iG)$. According to the Lagrange's theorem, $I_i = \prod_{i \neq j} \frac{ID_i}{ID_i - ID_j}$, we deduce $\sum_{i=1}^{t} (f(ID_i)I_iG) = \sum_{i=1}^{t} ((a_0 + a_1ID_i + a_2ID_i^2 + \ldots + a_{t-1} ID_i^{t-1})I_iG) = f(0)G = c_p$. Since $W = \sum_{i=1}^{t} I_iX_i$, $SG = R - \sum_{i=1}^{t} (zX_iI_i) - zc_p = R - zW - zc_p$, we obtain $SG + z(c_p + W) = R$.

6.2 Security Analysis

6.2.1 Threshold Safety Analysis

The threshold group signature scheme is adapted to the blockchain. For the blockchain with n nodes, at least t nodes must cooperate to obtain the resulting signature. For a well-designed threshold group signature, if an attacker breaks through a certain number of nodes, as long as the number of valid nodes is greater than or equal to t, the voting result still be correct.

For n participants, when generating the share signatures, each node in the blockchain utilizes its own private key u_s to sign the message and generate the share signature with the formula $s_i = k_ix_{ri} - zu_sI_i \bmod p$. After the trusted center receives the share signature (r_i, s_i) for each participant, the trusted centers synthesize them with the formula $S = \sum_{i=1}^{t} s_i = \sum_{i=1}^{t} (k_ix_{ri} - zu_sI_i)$, and need to verify the result with the formula $SG + z(c_p + W) = R$.

At least t nodes are required to synthesize the share signatures. If the number of available signatures is less than t, the synthesis will fail. After obtaining the group signature, it is necessary to verify $SG + z(c_p + W) = R$. According to the Lagrange's theorem, we can obtain $\sum_{i=1}^{t} (f(ID_i)I_iG) = f(0)G = c_p$, which needs at least t identities to pass the verification. In practice, because the public key c_p and G is well-known, the attacker can obtain them. When we want to obtain the private group key $f(0)$, the difficulty is equivalent to the discrete logarithm problem of the elliptic curve which has been proved hard, it shows that the proposed scheme is safe.

6.2.2 Anonymity Analysis

For the blockchain applications, recent studies [4, 5] exploit the vulnerabilities of propagation and transaction mechanism of the blockchain, and successfully obtain the address of the targeted transaction to infer user identities, which shows the native blockchain cannot guarantee anonymity.

The proposed scheme blinds the user identities with $ID_i = (x_u + s)h(Id_i) + u \bmod p$, which relies on the partial key x_u of the current node i and the private key s of the trusted center. The values are respectively grasped by the users and the trusted center, other unauthorized nodes are not able to access them. Even the attacker obtains x_u and s, it is still difficult for him to retrieve the user identities through the one-way hash function $h(x)$.

6.2.3 Unforgeability Analysis

Unforgeability means that the node in the blockchain cannot impersonate others to generate the signature. In the blockchain scenario, the trusted center and the users may impersonate other identities to sign the messages. For the convenience, the current node is defined as i, and one of other nodes is defined as j. Unforgeability of the trusted center and other nodes will be analyzed separately.

For the first case, the trusted center impersonates one user identity to sign. Since the trusted center is updated based on Dynamic Bayesian Network, the metrics of the trusted center depends on feedback from other nodes. If the attacker breaks through the trusted center, it will inevitably show some abnormal behavior characters, such as multiple re-entry, unauthorized access, etc. The credibility of the trusted center will be adjusted, which may strip the trusted center out of the blockchain. Because of $Sc_n(\mathrm{x_i},\ \mathrm{x_j},\ \mathrm{W(w,\ t),\ t}) = Sc_n(\mathrm{x_i},\ \mathrm{x_j}, \mathrm{W(w,\ t-1),\ t-1}) \times \delta(t), \mathrm{W(w,\ t)} - \mathrm{W(w,\ t-1)} = \Phi \times \delta(t), \mathrm{W(w,t)} - \mathrm{W(w, t-1)} = \Phi$, it means that the credibility of the trusted center will decline without other positive incentive. The rigorous measurement ensures the security of the trusted center.

If the trusted center personally conducts the attack, and impersonate the user node i, he can obtain some related information such as $< c_s, f(x) >$ and UL. The trusted center TC randomly selects $u'_i \in Z_p^*$, and because of $x'_i = u'_i$, he can obtain the private key $u'_s = x'_i + y'_i$, and $X'_i = u'_iG$. After that, he can obtain the public key $u'_p = u'_sG$. $< r_i, s_{i,} ID_i >$ and $< X_i, Id_i, ID_i >$ for UL have been backed up, which need to check the integrity. However, the prerequisite to pass the verification must satisfy $X'_i = X_i$, which denotes that the trusted center TC must obtain $X_i = X_i' = u_i'G$ and u_i'. The difficulty is equivalent to the discrete logarithm of the elliptic curve which is computationally hard.

For the second scenario, the node j poses as the node i for signature. At this time, the node j only knows ID_i of the node i. The node j randomly selects $u_i' \in Z_p^*$, and generates the private key $u_s' = x_i' + y_i'$, $X_i' = u_i'G$, and the public key $u_p' = u_s'G$. However, the trusted center has backed up the information whose integrity will be checked. It shows that the process is computationally hard.

7 Performance Analysis

The difficulty of the blockchain threshold group signature scheme proposed in the paper is equivalent to the discrete logarithm of the elliptic curve. Compared with those schemes based on the large number decomposition and discrete logarithm, the proposed scheme can obtain the higher security level with the shorter key length. In order to compare with existing signature schemes, the following symbols are defined (Table 3).

Table 3. Symbols of operations in the proposed scheme

Symbols	Description
C_m	Modular multiplication
C_{ec_m}	Elliptic curve scalar multiplication
C_{ec_a}	Elliptic curve addition
C_i	Modular inverse
C_h	Hash

Due to low overhead of modular addition and modular subtraction, those operations will not be considered in the paper. According to [14], $C_{ec\_m} \approx 29C_m$, $C_{ec\_a} \approx 0.12C_m$, the following parts will compare the proposed scheme with other schemes uniformly.

The paper will conduct the performance analysis from three aspects that are registration, generation and verification of the signature. In addition, when calculating the complexity of the hash function, it counts only once for the same tasks. The computational complexity is shown in Table 4.

Table 4. Computational complexity of the proposed scheme

Steps	Computational complexity	Total
Register	$(3+t)C_{ec\_m} + tC_{ec\_a} + C_h$	$(87 + 29.12t)C_m + C_h$
Signature generation	$4tC_{ec\_m} + 2(t-1)C_{ec\_a} + 3tC_m + (t+1)C_h$	$(119.24t - 0.24)C_m + (t+1)C_h$
Signature verification	$C_{ec\_m} + C_{ec\_a} + C_h$	$29.12C_m + C_h$

When comparing with existing threshold signature methods, two aspects that are generation and verification of the signature will be considered. The paper utilizes the comparison method presented in [13]. It should be noted that only the cryptographic schemes based on the elliptic curve are considered.

Table 5 is the comparison result of the computational complexity. From the table, we can infer that for the signature generation, [15] is superior to the proposed scheme, and for the signature verification, it is inferior. In the blockchain, the trusted center must undertake most of the computation. The proposed metrics consider the computing power, which makes the selected one own the abundant resource to increase the system throughout. On the other hand, the latter one does not realize the cancellation operation, which is not suitable for the blockchain.

Table 5. Comparison of computational complexity

Algorithms	Signature generation	Signature verification
Proposed	$(119.24t - 0.24)C_m + (t+1)C_h$	$29.12C_m + C_h$
Algorithm in [30]	$(209.36t - 87.88)C_m + C_h + tC_i$	$145.36C_m + C_h$
Algorithm in [14]	$(90.24t - 0.24)C_m + (t+1)C_h + tC_i$	$(58.12 + t)C_m + C_h$
Algorithm in [16]	$(235.12t - 0.12)C_m + (t+1)C_h$	$87.24C_m + C_h$
Algorithm in [15]	$(88.24t - 0.24)C_m + (t+1)C_h$	$58.12C_m + C_h$
Algorithm in [13]	$(119.24t - 0.24)C_m + (t+1)C_h$	$58.24C_m + C_h$

Compared with [13], the proposed scheme introduces the strategy based on Dynamic Bayesian Network to construct the trusted centers to enhance the security. For uneven distribution of computing resource in the blockchain, the proposed scheme is able to assign those tasks with high complexity to the trusted center, which can reduce the overhead of communication and effectively increase the system throughput. In addition, the identities have been blinded in the proposed design to achieve anonymity.

The signature generation process in [14, 30] contains the modular inverse operation, which has the high computational complexity, and [30] does not have the revoking operation. The proposed scheme can achieve the better result than [16] in the process of signature generation and signature verification.

In summary, the proposed method has the higher degree of anonymity and unforgeability that involves opening and revoking signatures. Compared with existing threshold group signature schemes, the proposed one has the lower computational complexity.

8 Conclusion

When applying the threshold group signature scheme to the blockchain to achieve the voting function, we will face up to the untrustworthy of the trusted center and the leakage of the user signatures. The paper proposes a trusted center selection scheme based on Dynamic Bayesian Network, which considers the time factor to perform the trusted metrics. The paper introduces the history interaction window, the aging factor and the penalty factor, and aggregates the direct and indirect credibility to form the dynamic metrics.

The voting scheme based on the threshold group signature for the isomerized blockchain is proposed. Through cooperation of users and the trusted centers to generate the group signature, the computational difficulty is equivalent to the discrete logarithms of the elliptic curve. The new design blinds and backs up the user identities. Security analysis shows that the proposed scheme can achieve the higher level of anonymity and effectively resist the impersonation attack. Computational complexity analysis shows that the cost of this scheme is low when adapting to the blockchain.

Acknowledgments. This work was supported by the National Natural Science Funds (U1304614, U1204703), the construct program of the key discipline in Zhengzhou Normal University, aid program for Science and Technology Innovative Research Team of Zhengzhou Normal University, Henan Province Education Science Plan General Topic “Research on Trusted Degree Certification Based on Blockchain” ((2018)-JKGHYB-0279).

References

1. Liehuang, Z., et al.: Survey on privacy preserving techniques for blockchain technology. J. Comput. Res. Dev. **54**(10), 2170–2186 (2017)
2. Miers, I., Garman, C., Green, M., Rubin, A.D.: Zerocoin: anonymous distributed E-Cash from Bitcoin. In: IEEE Symposium on Security and Privacy, San Francisco, pp. 397–411 (2013)
3. Baohua, Y., Chang, C.: Principle. Design and Application of Blockchain. China Machine Press, Beijing (2017)
4. Biryukov, A., Khovratovich, D., Pustogarov, I.: Deanonymisation of Clients in Bitcoin P2P Network. Eprint Arxiv, 15–29 (2014)
5. Koshy, P., Koshy, D., Mcdaniel, P.: An analysis of anonymity in Bitcoin using P2P network traffic. In: International Conference on Financial Cryptography and Data Security, Barbados, pp. 469–485 (2014)
6. Marsh, S.P.: Formalising trust as computational concept (1994)
7. Almenarez, F., Marin, A., Diaz, D., Sanchez, J.: Developing a model for trust management in pervasive devices. In: IEEE International Conference on Pervasive Computing and Communications Workshops, p. 267 (2006)
8. Melaye, D., Demazeau, Y.: Bayesian dynamic trust model. In: International Central and Eastern European Conference on Multi-Agent Systems, Budapest 2005, pp. 480–489 (2005)
9. Sun, Y.L., Yu, W., Han, Z., Liu, K.J.R.: Information theoretic framework of trust modeling and evaluation for ad hoc networks. IEEE J. Sel. Areas Commun. **24**(2), 305–317 (2006)
10. Ahamed, S.I., Sharmin, M.: A trust-based secure service discovery (TSSD) model for pervasive computing. Comput. Commun. **31**(18), 4281–4293 (2008)
11. Chun-Mei, G., Qiang, J., Huai-Min, W., Quan-Yuan, W.: Repeated game theory based penalty-incentive mechanism in internet-based virtual computing environment. J. Softw. **21** (12), 3042–3055 (2010)
12. Liang, H., Wu, W.: Research of trust evaluation model based on dynamic Bayesian network. J. Commun. **34**(9), 68–76 (2013)

13. Liquan, C., Zheng, Z., Muyang, W., Xiaoyan, S.: A threadhold group signature scheme for mobile internet application. Chin. J. Comput. **41**(5), 1–18 (2018)
14. Chen, T.S., Hsiao, T.C., Chen, T.L.: An efficient threshold group signature scheme. In: TENCON 2004, pp. 13–16 (2004)
15. Ya, P.: Research on Threshold Digital Signature Theory and Application. Zhongshan University, China (2010)
16. Dong, X., Jiajia, L., Zhonghua, S.: A new threshold signature scheme based on elliptic curve cryptosystem. J. Hangzhou Norm. Univ. (Nat. Sci. Ed.) **12**(1), 57–60 (2013)
17. Liu, H.W., Xie, W.X., Jian-Ping, Y.U., Peng, Z.: Efficiency identity-based threshold group signature scheme. J. Commun. **30**(5), 122–127 (2009)
18. Jie, Y., Xuri, Y., Wujun, Z.: Research on group signature with threshold value based on elliptic curve. J. Southeast Univ. (Nat. Sci. Ed.) **38**, 43–46 (2008)
19. Chung, Y.F., Chen, T.L., Chen, T.S., Chen, C.S.: A study on efficient group-oriented signature schemes for realistic application environment. Int. J. Innov. Comput. Inf. Control. Ijicic **8**(4), 2713–2727 (2012)
20. Gennaro, R., Jarecki, S., Krawczyk, H., Rabin, T.: Robust threshold DSS signatures. In: Maurer, U. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 354–371. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-68339-9_31
21. Gennaro, R., Jarecki, S., Krawczyk, H., Rabin, T.: Secure distributed key generation for discrete-log based cryptosystems. In: International Conference on Theory and Application of Cryptographic Techniques, pp. 295–310 (1999)
22. Bonneau, J., et al.: Securing Bitcoin wallets via a new DSA/ECDSA threshold signature scheme (2015). https://www.cs.princeton.edu/~stevenag/threshold_sigs.pdf
23. Si, C.: Research on Anonymity and Key Management of Bitcoin. Xidian University, Xi'an (2017)
24. Stinson, D.R., Dengguo, F.: Cryptography Theory and Practice. Publishing House of Electronics Industry, Beijing (2016)
25. Abe, M., Fujisaki, E.: How to date blind signatures. In: Kim, K., Matsumoto, T. (eds.) ASIACRYPT 1996. LNCS, vol. 1163, pp. 244–251. Springer, Heidelberg (1996). https://doi.org/10.1007/BFb0034851
26. Shamir, A.: How to Share a Secret. Commun. ACM **22**(11), 612–613 (1979)
27. Blakley, G.R.: Safeguarding cryptographic keys. In: AFIPS, p. 313 (1979)
28. Boldyreva, A.: Threshold signatures, multisignatures and blind signatures based on the Gap-Diffie-Hellman-Group Signature Scheme. In: International Workshop on Public Key Cryptography, pp. 31–46 (2003)
29. Cai, Y., Zhang, X., Jiang, N.: A novel identity-based threshold signature. Acta Electron. Sin. **37**(4A), 102–105 (2009)
30. Dahshan, H., Kamal, A., Rohiem, A.: A threshold blind digital signature scheme using elliptic curve dlog-based cryptosystem. In: Vehicular Technology Conference, Glasgow, pp. 1–5 (2015)