Hindawi
Security and Communication Networks
Volume 2019, Article ID 6989383, 8 pages
https://doi.org/10.1155/2019/6989383

Research Article

Outsourcing Hierarchical Threshold Secret Sharing Scheme Based on Reputation

En Zhang, Jun-Zhe Zhu, Gong-Li Li, Jian Chang, and Yu Li

College of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China

Correspondence should be addressed to En Zhang; zhangenzdrj@163.com

Received 11 April 2019; Accepted 24 August 2019; Published 10 October 2019

Guest Editor: Mehdi Hussain

Secret sharing is a basic tool in modern communication, which protects privacy and provides information security. Among the secret sharing schemes, fairness is a vital and desirable property. To achieve fairness, the existing secret sharing schemes either require a trusted third party or the execution of a multiround protocol, which are impractical. Moreover, the classic scheme requires expensive computing in the secret verification phase. In this work, we provide an outsourcing hierarchical threshold secret sharing (HTSS) protocol based on reputation. In the scheme, participants from different levels can fairly reconstruct the secret, and the protocol only needs to run for one round. A cloud service provider (CSP) uses powerful computing resources to help participants complete homomorphic encryption and complex verification operations, and the CSP cannot be aware of any valuable information. The participants can obtain the secret with a small number of operations. To avoid collusion, we suppose that participants have their own reputation value, and they are punished or rewarded according to their behavior. The reputation value of a participant who deviates from the protocol will decrease; therefore, the participant will choose a cooperative strategy to obtain better payoffs. Lastly, our scheme is proved to be secure, and experiments indicate that our scheme is feasible and efficient.

1. Introduction

Secret sharing is an important cryptographic primitive and has a widespread application in secure multiparty computation, image encryption, and attribute-based encryption. Secret sharing, an idea proposed by Shamir [1] and Blakley [2], allows a dealer to distribute different shares among a set of participants. The method guarantees any authorized subsets of t or more participants can reconstruct the secret. However, it is hard to guarantee that the dealer and participants are absolutely honest. To address this problem, verifiable secret sharing (VSS) schemes [3–5] guarantee additionally any cheating behavior can be detected, which can check the validity of shares. Subsequently, a series of protocols [6–9] is studied sharing multiple secrets at a time. In these schemes, participants only need to submit a pseudoshare rather than a real share to recover multiple secrets. Secret sharing has become an important research topic, and a large quantity of studies have been proposed. A multistage secret sharing scheme was introduced by Pilaram and Eghlidos [10], which was based on Lattice and could resist quantum attacks. Zhang et al. [11] presented an outsourcing secret sharing scheme based on homomorphic encryption, but the scheme could not effectively resist collusion. Recently, secret sharing has stronger privacy requirements. Although information about shares is leaked, the adversary still has no access to information about secret. Fehr and Yuan [12] constructed a robust secret sharing scheme with security against a rushing adversary. Benhamouda et al. studied leakage resilience of the MPC protocol [13]. A nonmalleable scheme concerning secret sharing was presented by Goyal and Kumar [14]. The scheme can resist adversary of someone who arbitrarily tampers with shares. Later, Goyal and Kumar [15] proposed nonmalleable secret sharing schemes for more general access structures.

In real life, everyone is not exactly equal in status or privileges. It would be an endless task to cite such living examples. For example, in a research and development department of a company, the shares of the private key of confidential files may be distributed among employees.

Some are accountants, and some are department managers. The company's policy requires 3 employees to be in attendance at the same time to open confidential files, but at least one of them must be a department manager. Such a setting requires a special secret sharing method. Therefore, the concept of HTSS was proposed. Tassa [16] introduced the structure of HTSS. In the scheme, a secret is shared among participants that are divided into different levels. Only participants who meet a certain level can reconstruct the secret. If the specific level is not met, the participants learn nothing about the secret. Later, Traverso et al. [17] proposed an HTSS scheme that supports verifiability and dynamics, which can add, remove, and renew shares. Recently, Mohamed and Arockia [18] introduced an HTSS scheme for color images. Bhattacharjee et al. [19] presented a hierarchical image scheme for bandwidth efficient transmission and offered a great degree of robustness in compressed sensing.

In the classic secret sharing scheme, fairness is a desirable property that guarantees each participant can gain the secret simultaneously. For the purpose of the goal, Tompa and Woll [20] firstly introduced a fair reconstruction scheme. The main idea of the scheme is to hide the real secret value, and the cheater has to guess the secret location. However, it is impractical for all participants to release their shares synchronously. A novel fair threshold scheme was presented by Tian et al. [21]. In the work, the real secret value was hidden in the sequence for the sake of decreasing the probability of the cheater achieving a successful guess. Combining the approach with game theory, Halpern and Teague [22] introduced a rational cryptographic protocol. In the rational scheme, the participants are rational players whose behavior aims are to maximize their profit. To achieve fairness, existing schemes require either a trusted third party or the execution of a multiround protocol, which are impractical.

The reputation system plays a key role in the online community, such as auction markets, trusted content delivery, and e-commerce. By publicizing the reputation value, participants can choose trusted peers with whom to cooperate. Reputation systems can effectively combat selfish, dishonest, and malicious behavior. Xiong and Liu [23] presented a detailed explanation. Combining with reputation systems, Zhang et al. [24] proposed a PSI protocol against social rational participants in which the parties who defect the PSI protocol will be penalized. Nojoumian and Stinson [25] introduced a socio-rational protocol. In this paper, participants are invited to execute an unknown number of protocols based on their reputation. Recently, a series of works were proposed. Litos and Zindros [26] created a reputation network in which the reputation value is quantifiable and expressed in monetary terms. Clark et al. [27] presented a dynamic, privacy-preserving decentralized reputation system.

At present, the vast amount of data stored in the cloud has led to explosive growth in the data volume. People are entering the era of big data, and everything will be digitized. According to the statistics of the Millet cloud storage service, the number of customers at the end of 2015 reached 97 million, with 46.5 billion photos and 504 million videos. It is estimated that by 2020, the global data volume will reach 44 ZB. At the same time, cloud outsourcing computing is also very common. More and more devices with poor computing power such as smart phones, pads, and sensors can outsource computing to a CSP with powerful computing power so that users can enjoy unlimited computing resources. However, in the face of outsourcing computing, users are reluctant to disclose their personal sensitive data. Therefore, we need to find a practical approach to implement an HTSS scheme.

1.1. Our Contribution. We provide an outsourcing HTSS protocol based on reputation, as is demonstrated in Figure 1. In this protocol, secret shares are distributed to different levels of participants. The participants can obtain the secret fairly with a small quantity of operations. Expensive computing is outsourced to a CSP, and the CSP can gain nothing about the secret. Moreover, the reputation system can effectively prevent participants from colluding with the server. Compared with previous schemes, our scheme has the following advantages:

(1) The participants are not required to always be online, which avoids multiple interactions between the participants and the server.

(2) The protocol could accurately check the malicious behavior of the participants or the server.

(3) Expensive computing is outsourced to a CSP. With the CSP's computing power, the CSP can execute homomorphic encryption and complex verification operations, and the server can gain nothing about the secret.

(4) Through a combination with the reputation system, we design a social game model for the hierarchical secret sharing scheme, which can resist collusion between the participant and the server. Assuming that participants have their own reputation value, they are punished or rewarded according to their behavior. Moreover, all participants are rational players whose behavior aims are to maximize their profit. The reputation value of a participant deviating from the protocol will decrease. In our model, a participant who chooses a cooperative strategy can obtain better payoffs. Therefore, each participant will honestly abide by the protocol.

We formally describe preliminaries in Section 2. We construct an outsourcing HTSS scheme based on reputation in Section 3. We indicate the security of the scheme in Section 4 and compare our scheme with previous schemes in Section 5. Finally, the conclusion of our paper is presented in Section 6.

2. Preliminary

2.1. Secret Sharing Homomorphisms. Benaloh [28] described the homomorphic property of secret sharing. For example, consider two secrets k_1 and k_2, which are shared by

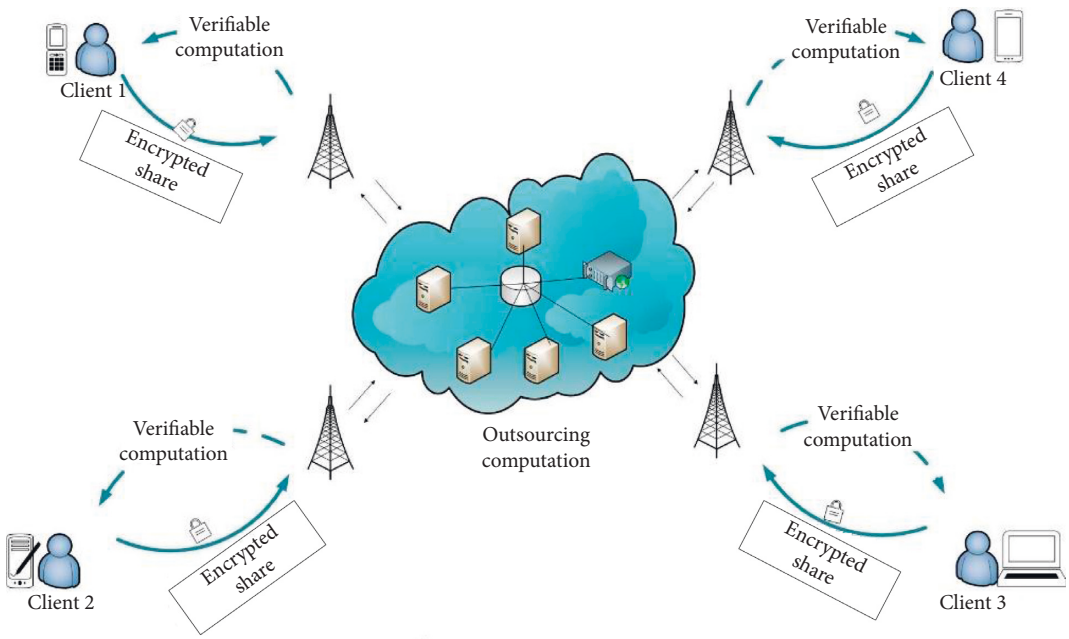

Figure 1: Outsourcing the hierarchical secret sharing scheme based on reputation.

polynomials $\varphi(x)$ and $\phi(x)$. If we add the shares $f(i) = \varphi(i) + \phi(i)$, $1 \le i \le n$, each of $f(i)$ can be viewed as a subshare of secret $k_1 + k_2$. Suppose that K is defined as the secret domain, and Σ is defined as the share domain. A set of functions $F_I: \Sigma^t \longrightarrow K$ can be determined, where $I \subseteq \{1, 2, \ldots, n\}$ and $|I| = t$. Given any set of t values $k_{i_1}, \ldots, k_{i_t}$, the following equation can define the secret k:

$$k = F_I\left(k_{i_1}, \ldots, k_{i_t}\right), \quad \text{for } I = \{i_1, \ldots, i_t\}. \tag{1}$$

Definition 1. Suppose $\oplus$ and $\otimes$ are two operations on the secret domain K and share domain Σ, respectively. There are

$$\begin{aligned} k &= F_I\left(k_{i_1}, \ldots, k_{i_t}\right), \\ k' &= F_I\left(k'_{i_1}, \ldots, k'_{i_t}\right), \end{aligned} \tag{2}$$

then,

$$k \oplus k' = F_I\left(k_{i_1} \otimes k'_{i_1}, \ldots, k_{i_t} \otimes k'_{i_t}\right). \tag{3}$$

From the above definition, Shamir's polynomial is $(+,+)$-homomorphic, which implies that the sum of the shares is equivalent to shares of the sum.

2.2. Tassa's $(\mathbf{t}, n)$ Hierarchical Threshold Scheme. In HTSS, a set of participants $P = \{P_1, \ldots, P_n\}$ are split into multiple levels $U_0, U_1, \ldots, U_m$, where U_0 is the highest level and U_m is the lowest level. For all $0 \le i < j \le m$, there is $P = \cup_{i=0}^{m} U_i$, where $U_i \cap U_j = \varnothing$. Supposing that n_h is the number of participants associated with level U_h, we can obtain $n = |P| = \sum_{h=0}^{m} n_h$. Then, we define a threshold t_h associated with level U_h, for $h = 0, \ldots, m$, which satisfies $0 < t_0 < \ldots < t_m$. In addition, we set $\mathbf{t} = \{t_h\}_{h=0}^{m}$, $t = t_m$, and $t_{-1} = 0$. Therefore, the $(\mathbf{t}, n)$ hierarchical access structure Γ is described as follows:

$$\Gamma = \left\{ A \subset P: \left| A \cap \left(\bigcup_{j=0}^{i} U_j \right) \right| \ge t_i, \quad \forall i \in \{0, 1, \ldots, m\} \right\}. \tag{4}$$

Next, we describe in detail how the Birkhoff interpolation reconstructs the secret.

The Birkhoff interpolation problem is to find a polynomial $F(x) = \sum_{r=0}^{t-1} a_r x^r \in \mathbb{R}_{t-1}[x]$ that satisfies the equalities $F(i) = \sigma_{i,j}$, where $F^j(i)$ is the j-th derivative of $F(x)$ at position i. Suppose that an authorized subset $R \in \Gamma \subset P$ can reconstruct the secret. E associated with R is a matrix with binary entries. If there is participant $p_{i,j}$ with share $\sigma_{i,j}$, then the entry $e_{i,j}$ is set to "1". In addition, we set $\varphi = \{1, x, \ldots, x^t\} = \{\omega_0, \omega_1, \ldots, \omega_{t-1}\}$ and define ω_r^j as the j-th derivative of ω_r. The matrix $A(E, X, \varphi)$ can be expressed as follows:

$$A(E, X, \varphi) = \begin{pmatrix} \omega_0^{j_1}(i_1) & \omega_1^{j_1}(i_1) & \cdots & \omega_{t-1}^{j_1}(i_1) \\ \omega_0^{j_2}(i_2) & \omega_1^{j_2}(i_2) & \cdots & \omega_{t-1}^{j_2}(i_2) \\ \vdots & \vdots & \vdots & \vdots \\ \omega_0^{j_r}(i_r) & \omega_1^{j_r}(i_r) & \cdots & \omega_{t-1}^{j_r}(i_r) \end{pmatrix}, \quad \text{where } r = 0, \ldots, t-1. \tag{5}$$

The polynomial $F(x)$ can be reconstructed:

$$F(x) = \sum_{r=0}^{t-1} \frac{\left|A(E, X, \varphi_r)\right|}{|A(E, X, \varphi)|} x^r, \tag{6}$$

in which we can obtain $A(E, X, \varphi_r)$ by replacing the $(r+1)$-th column with the shares $\sigma_{i,j}$ in the lexicographic order.

Definition 2. Let M be a message space, Σ be a share space, and Γ be an access structure where t_h is the threshold associated with level U_h. Suppose that the pair (i, j) is the identity of participant $p_{i,j} \in U_h$. Then, an HTSS scheme contains the *share phase* and *reconstruction phase*.

Share Phase. A dealer outputs n shares $\sigma_{i,j} \in \sum$ that is distributed to participant $p_{i,j} \in U_h$.

Reconstruction Phase. An authorized subset R of t participants, which satisfies $R \in \Gamma$, can reconstruct the secret $k \in M$ using Birkhoff interpolation.

2.3. Social Game Model of Secret Sharing. Reputation systems can provide an incentive for honest behavior and help people decide who is trustworthy. Several reputation systems have been deployed in practical applications, such as encouraging compliance with e-commerce contracts. Next, we briefly review the related concepts and methods in [25].

Definition 3. Let $T_i^j(p)$ be the trust value assigned by participant P_j to P_i during period p. Let $T_i : \mathbb{N} \longmapsto \mathbb{R}$ be the trust function computing the reputation of P_i:

$$T_i(p) = \frac{1}{n-1} \sum_{j \neq i} T_i^j(p), \quad \text{where } -1 \leq T_i(p) \leq +1 \text{ and } T_i(0) = 0. \tag{7}$$

The monotonically increasing function $\mu(x)$ and the monotonically decreasing function $\mu'(x)$ are used to update reputation values recursively, that is, computing $T_i(p)$ by $T_i(p-1)$. If participant P_i has a choice of cooperating during period p, then $T_i(p) = T_i(p-1) + \mu(x)$. If participant P_i has a choice of defecting during period p, then $T_i(p) = T_i(p-1) - \mu'(x)$.

Subsequently, we review the payoff assumption. Let $u_i(a)$ be P_i's payoff by considering future action, let $u_i'(a)$ be P_i's payoff by considering current action, let $l_i(a) \in \{0, 1\}$ define whether the participant is aware of secret during period p, and define $\text{num}(a) = \sum l_i(a)$. The generalized payoff assumptions of social games are as follows:

(A) $l_i(a) = l_i(a')$ and $T_i^{a'}(p) > T_i^{a'}(p) \Longrightarrow u_i(a) > u_i(a')$

(B) $l_i(a) > l_i(a') \Longrightarrow u_i(a) > u_i(a')$

(C) $l_i(a) = l_i(a')$ and $\text{num}(a) < \text{num}(a') \Longrightarrow u_i(a) > u_i(a')$

Remark 1. A, B, and C have impact factors ρ_1, ρ_2, and ρ_3, respectively, where $\rho_1 \gg \rho_2 \geq \rho_3$.

Let

$$\omega_i(a) = \frac{3}{2 - T_i^a(p)}. \tag{8}$$

We can obtain the current payoff $u_i'(a)$ and the future payoff $u_i(a)$ as follows:

$$\begin{aligned} u_i'(a) &= \rho_2 l_i(a) + \rho_3 \frac{l_i(a)}{\text{num}(a) + 1}, \\ u_i(a) &= \rho_1 \frac{\left| T_i^a(p) - T_i^a(p-1) \right|}{T_i^a(p) - T_i^a(p-1)} \times \omega_i(a) + u_i'(a). \end{aligned} \tag{9}$$

3. The HTSS Scheme Based on Reputation

In this section, combining an outsourcing computation and the reputation system, we propose a novel outsourcing HTSS protocol based on reputation. In the protocol, t or more parties from different levels can recover the secret. The scheme contains five phases: an initialization phase, a secret distribution phase, an outsourcing phase, a reconstruction phase, and a reputation update phase. We formally defined some parameters during the initialization phase. In the secret distribution phase, a dealer distributes encrypted shares and broadcasts verification information and participants receive a random value and encrypted shares. Then, the participants send shares to a CSP, and the CSP returns the results to the participants where the CSP cannot be aware of any valuable information about the secret. Next, the participants can obtain the secret fairly in the reconstruction phase. Finally, we can update the participant's reputation value. To avoid collusion, participants have their own reputation value and they are punished or rewarded according to their behavior. For example, if a participant wants to collude with the CSP and sends a collusion invitation to the CSP, then we can penalize the participant according to the reputation system.

3.1. Initialization Phase. Let p and q, such as $(q \mid p-1)$, be two large primes, g be a generator of the q-th order subgroup $\mathbb{F}_q$ of $\mathbb{F}_p^*$, and $H(x)$ be a collision-resistant hash function.

A secret k is shared among n-parties, and a set of parties denoted by $P = \{P_1, \ldots, P_n\}$ are split into multiple levels $U_0, U_1, \ldots, U_m$. n_h is the number of participants associated with level U_h, and t_h is the threshold associated with level U_h, for $h = 0, \ldots, m$. The pair (i, j) is the identity of participant $p_{i,j} \in U_h$, for $i = 1, \ldots, n_h$, $j = t_{h-1}$, and $t_{-1} = 0$.

3.2. Secret Distribution Phase. The trusted dealer distributes shares by performing the following stages:

Step 1. The dealer randomly chooses $t-1$ coefficients $a_1, \ldots, a_{t-1} \in \mathbb{F}_q$ and generates a polynomial with $t-1$ degree:

$$f(x) = \sum_{r=0}^{t-1} a_i x^r \bmod q, \tag{10}$$

where a_0 is a secret value, i.e., $k = a_0$. The corresponding shares are $\sigma_{i,j} = f^j(i)$, where $f^j(i)$ is the j-th derivative of the polynomial $f(x)$ at position i.

Step 2. The dealer randomly chooses $t-1$ coefficients $a_1', \ldots, a_t' \in \mathbb{F}_q$ and generates a polynomial with $t-1$ degree:

$$f'(x) = \sum_{r=0}^{t-1} a_i' x^r \bmod q, \tag{11}$$

where a_0' distributed to all participants is a random value. The corresponding shares are $\sigma_{i,j}' = f'^j(i)$.

Step 3. According to the $(+,+)$-homomorphic property, the sum of the shares is equivalent to the shares of the sum, and the dealer performs the following operation:

$$\xi_{i,j} = \sigma_{i,j} \otimes \sigma_{i,j}' = f^j(i) \otimes f'^j(i). \tag{12}$$

Step 4. The dealer distributes $(\xi_{i,j}, H(a_0))$ to participant $p_{i,j} \in U_h$, for $i = 1, \ldots, n_h$, $j = t_{h-1}$, and $h = 0, \ldots, m$.

Step 5. The dealer broadcasts verification information:

$$c_r = g^{a_r \otimes a_r'} \bmod p, \quad r = 0, \ldots, t-1. \tag{13}$$

3.3. Outsourcing Phase. Suppose that t or more participants from different levels commit their shares, and then they will perform the following stages:

Step 1. An authorized subset of t participants sent $(\xi_{i,j}, c_r)$ to the CSP.

Step 2. According to following equation, the CSP checks whether $(\xi_{i,j}, c_r)$ is correct:

$$g^{\xi_{i,j}} \equiv \prod_{r=j}^{t-1} c_r^{(r!/(r-j)!)i^{r-j}} = g^{f^j(i)} \bmod p, \tag{14}$$

where $r = 0, \ldots, t-1$. The CSP performs Step 3 if the above equation is held; otherwise, the protocol is terminated and the deception of participant $p_{i,j}$ will be disclosed.

Step 3. The CSP uses Birkhoff interpolation to reconstruct $f(x)$ with $\xi_{i,j}$:

$$f(x) = \sum_{r=0}^{t-1} \frac{|A(E, X, \varphi_r)|}{|A(E, X, \varphi)|} x^r. \tag{15}$$

According to the above equation, the CSP can learn $k' = F(0) = k \oplus a_0'$ and send k' to t participants.

3.4. Reconstruction Phase. Each participant can obtain the secret with a small amount of computation according to the following steps:

Step 1. The participant can obtain the secret k by $k = k' \oplus a_0'$.

Step 2. The participant can verify secret k according to the following equation:

$$H(a_0) = H(k). \tag{16}$$

If the equation is true, CSP's calculation is correct; otherwise, it is wrong.

3.5. Reputation Update Phase. The reputation value updates as follows:

Case 1. If $P_k (1 \le k \le n+1)$ sends a collusion to $P_{j \ne k} (1 \le k \le n+1)$ and P_j has a choice of colluding with P_k, then the colluder earns ρ_4, where $\rho_1 \gg \rho_2 \ge \rho_3 \ge \rho_4$ and $P_{n+1} = \text{CSP}$.

Case 2. If P_j has a choice of not to collude with P_k and broadcasts his malicious behavior, then P_j's reputation value will increase. In contrast, P_k's reputation value will decrease.

Case 3. If each participant has a choice of cooperating, then the reputation value will increase; otherwise, the reputation value will decrease.

4. Security Analysis

In the section, we give the analysis of the protocol.

Theorem 1. *The outsourcing HTSS scheme is secure and any $t-1$ or fewer participants get nothing about the secret.*

Proof. (a) Any $t-1$ or fewer participants get nothing about the secret.

In the scheme, any $t-1$ or fewer participants' collusion from different levels cannot obtain the secret with their subshares $\xi_{i,j}$ for $i = 1, \ldots, n_h$, $j = t_{h-1}$, and $h = 0, \ldots, m$ because the Birkhoff interpolation requires t values to determine the unique solution.

(b) The CSP cannot be aware of any valuable information about the secret.

The scheme protects the participant's privacy, and the CSP does not know the participant's input and output. An authorized subset of t participants sends encrypted share $\xi_{i,j}$ to the CSP. Therefore, the CSP cannot be aware of any valuable information about the secret. □

Theorem 2. *The outsourcing HTSS scheme can verify malicious behavior, and the malicious behavior can be detected in time.*

Proof. (a) The participants and the CSP can check invalid shares.

The public verification information $c_r = g^{a_r \oplus a_r'}$ can check shares whether is correct, and a commitment to the $\xi_{i,j}$ can be expressed by the following equation:

$$g^{\xi_{i,j}} = g^{\sigma_{i,j} \otimes \sigma'_{i,j}} = g^{\left(a_0 + a_1 i + \cdots + a_{t-1} i^{t-1}\right)^{(j)} \otimes \left(a'_0 + a'_1 i + \cdots + a'_{t-1} i^{t-1}\right)^{(j)}}$$
$$= g^{\left(a_0 \otimes a'_0\right) + \left(a_1 \otimes a'_1\right) i^{(j)} + \cdots + \left(a_{t-1} \otimes a'_{t-1}\right) i^{t-1\,(j)}}$$
$$= (\alpha_0)^{(j)} (\alpha_1) i^{(j)} \cdots (\alpha_{t-1}) i^{t-1\,(j)}. \tag{17}$$

Thus, the validity of $\xi_{i,j}$ can be checked:

$$g^{\xi_{i,j}} \equiv \prod_{r=j}^{t-1} c_r^{(r!/(r-j)!) i^{r-j}} = g^{f^j(i)} \bmod p, \tag{18}$$

and the malicious behavior can be detected in time.

(b) The participants can verify the CSP's calculation result.

The participants can verify the calculation result by a collision-resistant hash function. If $H(a_0) = H(k)$, the participants can confirm that the CSP's calculation is correct; otherwise, the result is incorrect. Moreover, the participants can detect the CSP's malicious behavior in time. □

Theorem 3. *The scheme is a social Nash equilibrium and collusion-free if the rational participant chooses a cooperation strategy.*

Proof. (a) The scheme is not secure if the participants collude with the CSP.

The scheme cannot resist collusion between the server and other participants. In the scheme, if P_i receives the CSP's collusion invitation and sends a'_0 to the CSP, then the CSP can obtain the real secret k instead of $k' = k \oplus a'_0$.

(b) Following the method in [25], we consider all participants are rational. Let Coop_j define that participant P_j chooses a cooperation strategy where $1 \le j \le n+1$ and $P_{n+1} = \text{CSP}$, let Coll_j define that P_j chooses a collusion strategy, let Coop_{-j} denote that all participants choose a cooperation strategy except for P_j, and let $\text{Coop}_{-i||j}$ denote that all the participants choose a cooperation strategy except for P_i and P_j.

If all the participants have a choice of cooperating denoted by $(\text{Coop}_j, \text{Coop}_{-j})$, then the payoff functions for choosing cooperation strategy are $u_i^{(\text{Coop}_j, \text{Coop}_{-j})} = \Omega(\rho_1 \omega_i + \rho_2 + (\rho_3/(n+1)))$ and $u_{n+1}^{(\text{Coop}_j, \text{Coop}_{-j})} = \rho_1 \omega_{n+1}$ where $\omega_i = 3/(2 - T_i(p))$.

If P_i invites CSP to collude and the CSP has a choice of colluding with P_i with a probability of 0.5, then the payoff functions for choosing colluding strategy are $u_i^{(\text{Coll}_i, \text{Coll}_{n+1}, \text{Coop}_{-i\|n+1})} = \rho_1 \omega_i + \rho_2 + (\rho_3/(n+2)) + \rho_4$ and $u_{n+1}^{(\text{Coll}_i, \text{Coll}_{n+1}, \text{Coop}_{-i\|n+1})} = \rho_1 \omega_{n+1} + \rho_2 + (\rho_3/(n+2)) + \rho_4$ where $\rho_1 \gg \rho_2 \ge \rho_3 \ge \rho_4$; otherwise, if CSP has a choice of not to collude with P_i with a probability of 0.5 and publishes his malicious behavior, then $u_i^{(\text{Coll}_i, \text{Coop}_{-i})} = -\rho_1 \omega_i$ and $u_{n+1}^{(\text{Coll}_i, \text{Coop}_{-i})} = \rho_1 \omega'_{n+1}$, where $\omega'_i = 3/(2 - (T_i(P) + \mu(x)))$. If the CSP invites P_i to collude and P_i has a choice of colluding with CSP, then $u_{n+1}^{(\text{Coll}_{n+1}, \text{Coll}_i, \text{Coop}_{-i\|n+1})} = \rho_1 \omega_{n+1} + \rho_2 + (\rho_3/(n+2)) + \rho_4$ and $u_i^{(\text{Coll}_{n+1}, \text{Coll}_i, \text{Coop}_{-i\|n+1})} = \rho_1 \omega_i + \rho_2 + (\rho_3/(n+2)) + \rho_4$; otherwise, if P_i has a choice of not to collude with CSP and publishes his malicious behavior, then $u_{n+1}^{(\text{Coll}_{n+1}, \text{Coop}_{-(n+1)})} = -\rho_1 \omega_{n+1}$ and $u_i^{(\text{Coll}_{n+1}, \text{Coop}_{-(n+1)})} = \rho_1 \omega'_i$. The payoff function of P_i choosing a collusive strategy is $u_i = 1/2(\rho_2 + (\rho_3/(n+2)) + \rho_4)$, and the payoff function of P_i choosing a cooperative strategy is $u_i = 1/2(\rho_1 \omega_i + \rho_2 + (\rho_3/(n+2)) + \rho_1 \omega'_i)$. The payoff function of the CSP choosing a collusive strategy is $u_{n+1} = 1/2(\rho_2 + (\rho_3/(n+2)) + \rho_4)$, and the payoff function of the CSP choosing a cooperative strategy is $u_{n+1} = 1/2(\rho_1 \omega'_{n+1} + \rho_1 \omega_{n+1})$. The payoff function of cooperative strategy is larger than that of collusive strategy. From the above statements, we can conclude that choosing cooperation is the optimal strategy. □

Table 1: Computation time (in ms).

Client	$t=3$	$t=4$	$t=5$	$t=6$
Time of secret verification	791.52	868.21	1103.42	7370.42
Time of secret reconstruction	2.17	2.19	2.31	2.74

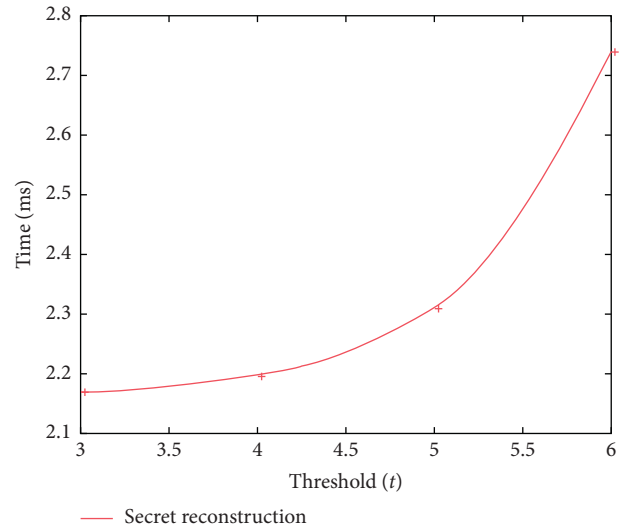

Figure 2: Secret reconstruction time.

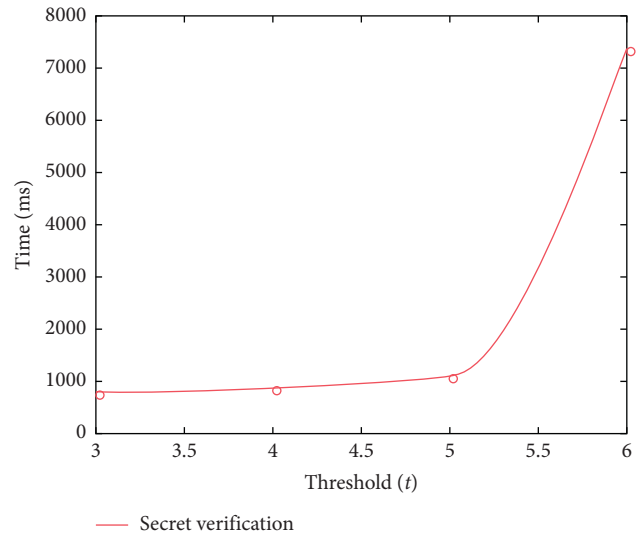

Figure 3: Secret verification time.

Table 2: Feature comparison of schemes.

	Maleka et al. [29]	Harn et al. [30]	Pilaram and Eghlidos [10]	Traverso et al. [17]	Our scheme
Fairness	No	Yes	Yes	No	Yes
Number of rounds	Multiple rounds	Multiple rounds	One round	One round	One round
Trusted third party	No	No	Yes	Yes	No
Interactive	Yes	Yes	No	Yes	No
Computation	User	User	User	User	CSP
Communication cost	O (tk)	O (t)	O (1)	O (t)	O (1)

5. Performance Analysis

We evaluated the prototype on a PC which has an Intel Core i7-6700 CPU (4-core 2.60 GHz) and 8 GB of RAM. To ignore network latency, we run the server and all clients on the same host. The times of the secret verification and secret reconstruction are given in Table 1. In Figure 2, the curve shows the reconstruction time of the scheme. According to the test results, the time varies from 2.17 ms to 2.74 ms. Figure 3 shows the time of the verification, and as the number of participant increases, the verification time increases exponentially. According to the test results, the time varies from 791.52 ms to 7370.42 ms. We conclude that the secret reconstruction requires less time than the verification algorithm.

In addition, we listed our comparison results in Table 2. Maleka et al. [29] analyzed a finite repeated game and an infinite repeated game, but the scheme could not effectively guarantee fairness. Traverso et al. [17] proposed an HTSS scheme that supports verifiability and dynamics, which can add, remove, and renew shares. Although the scheme can check invalid shares, the scheme cannot effectively guarantee fairness. A multistage secret sharing scheme was introduced by Pilaram and Eghlidos [10], which was based on Lattice and could resist quantum attacks. But this scheme requires a trusted third party. In order to achieve desire of fairness, Harn et al. [30] proposed a fair secret sharing scheme, but the scheme requires multiple protocol rounds and cannot be effectively applied to devices with poor computing capabilities.

In contrast, our scheme only needs to execute the protocol once. The participants only need to perform the decryption operation, and the communication cost is O (1). In the proposed scheme, complex operations such as homomorphic encryption and verification are outsourced to the CSP. Moreover, our scheme does not require participants to always be online.

6. Conclusion

Combining outsourcing computation and a reputation system, we provide an outsourcing HTSS protocol based on reputation. The participants can obtain the secret fairly with a small number of operations in this work. Expensive computing is outsourced to a CSP, and the CSP could learn nothing about the secret. The reputation system can effectively prevent participants from colluding with the server. Participants have their own reputation value, and they are punished or rewarded according to their behavior. Moreover, our protocol could accurately check the malicious behavior of the participants or the server and does not require multiple interactions between the participants and the server, which applies to cloud computing environments and mobile networks.

Data Availability

All data generated or analyzed during this study are included in this published article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (U1604156, 61772176, and 61602158) and Science and Technology Research Project of Henan Province (172102210045 and 192102210131).

References

[1] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612-613, 1979.

[2] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of the American Federation of Information Processing Societies (AFIPS'79) National Computer Conference*, vol. 48, pp. 313–317, CA, USA, February 1979.

[3] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch, "Verifiable secret sharing and achieving simultaneity in the presence of faults," in *Proceedings of the 26th Annual Symposium on Foundations of Computer Science*, pp. 383–395, IEEE, Portland, OR, USA, October 1985.

[4] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *Proceedings of the 28th Annual Symposium on Foundations of Computer Science*, pp. 427–438, IEEE, Los Angeles, CA, USA, October 1987.

[5] T. P. Pedersen, "Distributed provers with applications to undeniable signatures," in *Advances in Cryptology-EUROCRYPT*, pp. 221–242, Springer, Berlin, Germany, 1991.

[6] C. Blundo, A. De Santis, and U. Vaccaro, "Efficient sharing of many secrets," in *Proceedings of the Annual Symposium on Theoretical Aspects of Computer Science*, pp. 692–703, Springer, Würzburg, Germany, February 1993.

[7] H.-Y. Chien, J.-K. Jan, and Y.-M. Tseng, "A practical (t, n) multi-secret sharing scheme," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 83, no. 12, pp. 2762–2765, 2000.

[8] L.-J. Pang and Y.-M. Wang, "A new (t, n) multi-secret sharing scheme based on Shamir's secret sharing," *Applied Mathematics and Computation*, vol. 167, no. 2, pp. 840–848, 2005.

[9] C.-C. Yang, T.-Y. Chang, and M.-S. Hwang, "A (*t*, *n*) multi-secret sharing scheme," *Applied Mathematics and Computation*, vol. 151, no. 2, pp. 483–490, 2004.

[10] H. Pilaram and T. Eghlidos, "An efficient lattice based multi-stage secret sharing scheme," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 1, pp. 2–8, 2017.

[11] E. Zhang, J. Peng, and M. Li, "Outsourcing secret sharing scheme based on homomorphism encryption," *IET Information Security*, vol. 12, no. 1, pp. 94–99, 2018.

[12] S. Fehr and C. Yuan, "Towards optimal robust secret sharing with security against a rushing adversary," in *Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Darmstadt, Germany, May 2019.

[13] F. Benhamouda, A. Degwekar, Y. Ishai, and T. Rabin, "On the local leakage resilience of linear secret sharing schemes," in *Proceedings of the Annual International Cryptology Conference*, pp. 531–561, Springer, Santa Barbara, CA, USA, August 2018.

[14] V. Goyal and A. Kumar, "Non-malleable secret sharing," in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pp. 685–698, ACM, Los Angeles, CA, USA, June 2018.

[15] V. Goyal and A. Kumar, "Non-malleable secret sharing for general access structures," in *Proceedings of the Annual International Cryptology Conference*, pp. 501–530, Springer, Santa Barbara, CA, USA, August 2018.

[16] T. Tassa, "Hierarchical threshold secret sharing," *Journal of Cryptology*, vol. 20, no. 2, pp. 237–264, 2007.

[17] G. Traverso, D. Demirel, and J. Buchmann, "Dynamic and verifiable hierarchical secret sharing," in *Proceedings of the International Conference on Information Theoretic Security*, pp. 24–43, Springer, Tacoma, WA, USA, August 2016.

[18] F. P. Mohamed and R. J. P. Arockia, "Hierarchical threshold secret sharing scheme for color images," *Multimedia Tools and Applications*, vol. 76, no. 4, pp. 5489–5503, 2017.

[19] T. Bhattacharjee, S. P. Maity, and S. R. Islam, "Hierarchical secret image sharing scheme in compressed sensing," *Signal Processing: Image Communication*, vol. 61, pp. 21–32, 2018.

[20] M. Tompa and H. Woll, "How to share a secret with cheaters," *Journal of Cryptology*, vol. 1, no. 3, pp. 133–138, 1989.

[21] Y. Tian, C. Peng, Q. Jiang, and J. Ma, "Fair (*t*, *n*) threshold secret sharing scheme," *IET Information Security*, vol. 7, no. 2, pp. 106–112, 2013.

[22] J. Halpern and V. Teague, "Rational secret sharing and multiparty computation," in *Proceedings of the Thirty-Sixth Annual ACM symposium on Theory of computing*, pp. 623–632, ACM, Chicago, IL, USA, June 2004.

[23] L. Xiong and L. Liu, "Peertrust: supporting reputation-based trust for peer-to-peer electronic communities," *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 7, pp. 843–857, 2004.

[24] E. Zhang, F. Li, B. Niu, and Y. Wang, "Server-aided private set intersection based on reputation," *Information Sciences*, vol. 387, pp. 180–194, 2017.

[25] M. Nojoumian and D. R. Stinson, "Socio-rational secret sharing as a new direction in rational cryptography," in *Proceedings of the International Conference on Decision and Game Theory for Security*, pp. 18–37, Springer, Budapest, Hungary, November 2012.

[26] O. S. T. Litos and D. Zindros, "Trust is risk: a decentralized financial trust platform," in *Proceedings of the International Conference on Financial Cryptography and Data Security*, pp. 340–356, Springer, Sliema, Malta, April 2017.

[27] M. R. Clark, K. Stewart, and K. M. Hopkinson, "Dynamic, privacy-preserving decentralized reputation systems," *IEEE Transactions on Mobile Computing*, vol. 16, no. 9, pp. 2506–2517, 2017.

[28] J. C. Benaloh, "Secret sharing homomorphisms: keeping shares of a secret," in *Proceedings of the Conference on the Theory and Application of Cryptographic Techniques*, pp. 251–260, Springer, Santa Barbara, CA, USA, August 1986.

[29] S. Maleka, A. Shareef, and C. P. Rangan, "Rational secret sharing with repeated games," in *Proceedings of the International Conference on Information Security Practice and Experience*, pp. 334–346, Springer, Sydney, Australia, April 2008.

[30] L. Harn, C. Lin, and Y. Li, "Fair secret reconstruction in (*t*, *n*) secret sharing," *Journal of Information Security and Applications*, vol. 23, pp. 1–7, 2015.