Unconditionally Secure Social Secret Sharing Scheme

Mehrdad Nojoumian, Douglas R. Stinson, and Morgan Grainger

David R. Cheriton School of Computer Science
University of Waterloo, Waterloo, ON, N2L 3G1, Canada
fmnojoumi, dstinson, mjgraingg@uwaterloo.ca

Abstract. We introduce the notion of a *Social Secret Sharing Scheme*, in which shares are allocated based on a player's reputation and the way he interacts with other participants. During the social tuning phase, weights of players are adjusted such that participants who cooperate will end up with more shares than those who defect. Alternatively, newcomers are able to be enrolled in the scheme while corrupted players are disenrolled immediately. In other words, this scheme proactively renews shares at each cycle without changing the secret, and allows trusted participants to gain more authority. Our motivation is that, in real world applications, components of a secure scheme may have different levels of importance (i.e., the number of shares a player has) as well as reputation (i.e., cooperation with other players for the share renewal or secret recovery). Therefore, a good construction should balance these two factors respectively. In the proposed schemes, both the passive and active mobile adversaries are considered in an unconditionally secure setting. [1]

1 Introduction

The growth of Internet has created amazing opportunities for *secure multiparty computations* where various users, intelligent agents, or computer servers cooperate in order to conduct computation tasks based on the private data they each provide [8]. Since these computations could be among untrusted participants or competitors, consequently, the privacy of each participant's input is an important factor.

As stated in the literature, a fundamental method used in secure multiparty computations is the *secret sharing scheme* [19, 3], where a secret is divided into different shares for distribution among participants (private data), and a subset of participants then cooperate in order to reveal the secret (computation result). In particular, Shamir proposed the $(t; n)$*-threshold secret sharing scheme*, in which the secret is divided into n T971(and290-3dvtion)-3eis curious to learn the secret information. On the other hand, in the *active adversary* model, players may deviate from protocols while trying to learn the secret data.

In addition, the passive or active adversary might be classified in a *static* or *mobile* setting. The former refers to the adversary who corrupts players ahead of time, while in the latter case, the adversary may corrupt different players at different stages of the protocols' executions. Finally, the entire security model might be *computational* if one relaxes any of these assumptions, then he can decrease the computation and communication complexities. For instance, by using a trusted authority, or constructing the proposed scheme by relying on computational assumptions, or considering the simple passive adversary model without mobility.

1.3 Organization

This paper is organized as follows. Section 2 provides some preliminaries. Section 3 creates a general picture of our social secret sharing scheme. Section 4 demonstrates the first construction under the passive mobile adversary model. Section 5 extends the first scheme to the active mobile adversary model. Finally, Section 6 contains concluding remarks.

2 Preliminaries

In the following discussions, secret sharing schemes and trust management are quickly reviewed in order to create the required foundations for our proposed social secret sharing scheme.

2.1 Secret Sharing

As mentioned earlier, in a *$(t; n)$-threshold secret sharing scheme*, the secret is divided into n shares to be distributed among players. Consequently, the secret is reconstructed if at least t players cooperate with each other. On the other hand, any subsets of $t - 1$ players cannot learn anything about the secret.

In a *verifiable secret sharing scheme* [6], participants can verify that their shares are consistent with those of other participants. The authors in [1] present an unconditionally secure VSS when $t < \frac{n}{3}$. They only assume the existence of secure private channels between each pair of players. The proposed scheme in [16] uses the same communication model along with a broadcast channel to construct a new VSS when $t < \frac{n}{2}$. The former construction has a zero probability of error while the latter one has a negligible probability of error.

The authors in [11] illustrate the notion of the *proactive secret sharing scheme*, where the shares of players are updated without changing the secret. This solution is proposed for the mobile adversary model [15], where the adversary can infiltrate and gather the shares of an increasing number of participants over timnum0ubpad-333(o)28(v)27(etaFche)]T3(o)28(v)19091 Tf 153.9Sr-31he secre4.

Definition 1. *We define* $T_i^j(p)$

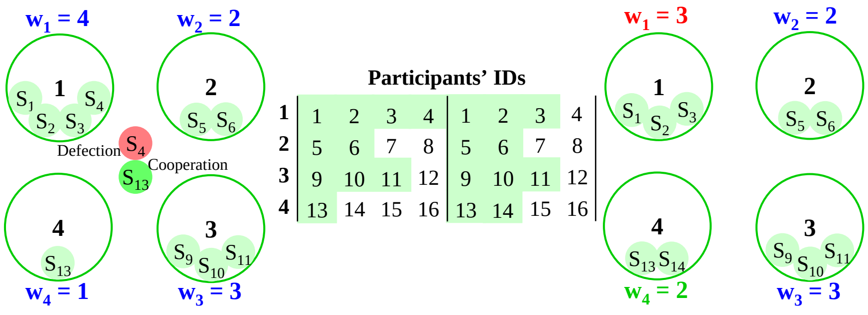
3 Social Secret Sharing Scheme

The proposed model consists of n participants, $P_1, P_2, \ldots, P_n$, and a dealer who is available only during the initialization phase. We assume the existence of private channels between each pair of participants (to be used during the share renewal step), and that the dealer can communicate privately with participants in the dealing stage. We also assume the existence of a synchronized broadcast channel, on which information is transmitted instantly and accurately to all participants. Let $\mathbb{Z}_q$ be a finite field and let ω be a primitive element in this field; all computations are performed in the field $\mathbb{Z}_q$.

Our intention is to construct unconditionally secure schemes, i.e., schemes that do not rely on computational assumptions. We consider both the passive and active adversaries with mobility, i.e., who are able to change the set of corrupted players from time to time during the execution of protocols. In the first construction, players correctly follow all protocols but are curious to learn the secret, while in the second one, players may deviate from the protocols.

In social secret sharing, each participant initially receives a constant number of shares. As time passes, players are assigned weights based on their behaviors in the scheme. Consequently, each participant receives a number of shares corresponding to his trust value which is the representation of a player's reputation over time. In fact, weights of participants are adjusted such that cooperative players receive more shares compared to non-cooperative ones. Alternatively, newcomers can join the scheme while corrupted players are disenrolled immediately. The reason for a corruption might be an active attack or a computational failure. Therefore, the corrupted server is able to re-enroll in the scheme only after being fixed, and in that case, he is treated as a newcomer.

Example 2. We consider a matrix $M_{n \times m}$ for the participants' identifiers, where n is the maximum number of participants and m is the maximum weight of any participant. As an example shown in Figure 1, assume we have four participants with different weights. After some period of time, suppose we observe defection (e.g., not being available to send S_4) from the first participant and cooperation from the fourth player. In that case, the scheme decreases w_1 to 3 and increases w_4 to 2. That is, disenrollment of $i = 4$ and enrollment of $i = 14$ take place.



Participants' IDs

1	1	2	3	4	1	2	3	4
2	5	6	7	8	5	6	7	8
3	9	10	11	12	9	10	11	12
4	13	14	15	16	13	14	15	16

Fig. 1. Social Secret Sharing Scheme

To further illustrate the proposed scheme, different possible behaviors are defined. After that, the required conditions are illustrated in order to ensure the scheme is working correctly. Finally, the formal definition of a social secret sharing scheme is presented.

Definition 3. *Cooperation $P_i(C)$: P_i is available at the time of share renewal or secret recovery and sends correct information. Defection $P_i(D)$: P_i is not available at the required time or probably responds with delay. Corruption $P_i(X)$: P_i has been compromised by a passive or active adversary and may send incorrect information.*

Definition 4. *To recover the secret, the total weight of authorized players 2 (uncorrupted) must be equal or greater than the threshold, i.e., $\text{P}_{P_i 2} w_i$ t. On the other hand, the total weight of colluders 2 r (corrupted) must be less than the threshold, i.e., $\text{P}_{P_i 2 r} w_i < t$. Finally, the weight of each player is bounded to a parameter much less than t, i.e., w_i m t for 1 i n.*

Definition 5. *The Social Secret Sharing Scheme S^4 is a three-tuple denoted as $S^4(Sha; Tun; Rec)$ consisting of secret sharing, social tuning, and secret recovery respectively. The only difference compared to the threshold scheme is the second stage, in which the weight of each player P_i is adjusted based on the player's reputation $T_i(p)$, after that, shares are updated accordingly.*

4 Passive Adversary Model Construction

In this construction, we consider the *passive adversary model*, where players follow all protocols but an unauthorized subset of them may collude to gather information and attempt to reconstruct the secret.

4.1 Secret Sharing ($\mathcal{S}ha$)

Suppose, the dealer initiates a secret sharing scheme by generating a polynomial $f(x)$ 2 $\mathbb{Z}_q$

in order to be the main shareholder and form a monopoly. In other words, it protects the scheme in a scenario where a malicious player cooperates for a while in order to gather most of the shares for a severe damage.

Furthermore, consider the scenario in which a player cooperates in the share renewal stage for several times (cheap cooperations) until reaching a high trust value, at which point he may defect the secret recovery stage (an expensive defection) without significant effect on his reputation value. The authors in [14] define the parameter as the *transaction cost*. In that case, the scheme would be able to fairly deal with the players' cooperation and defection.

Finally, since players' weights and consequently trust values are public information, therefore, scheme. For each participant P_i, consider the ratio of a player's trust value $T_i(p)$ to the number of shares he is holding $w_i(p)$. This ratio

Share Renewal. This stage consists of two phases. First, initial shares for newcomers or newly activated *ids* of existing players are generated. After that, players proactively update their shares, while disenrolled *ids* do not receive any more shares. As a result, old shares corresponding to those inactivated *ids* would be useless.

Phase-(I):

To update shares in a proactive scheme, a participant must have his previous shares. Suppose we intend to activate a new *id* in period p while we do not have its corresponding share in period $p-1$. For the sake of simplicity, assume each participant has one identifier, in that case, this problem can be resolved only if t participants cooperate together in order to generate the old share for the newcomer, where t is the threshold.

The initial solution to this problem, named *share recovery*, was proposed in [11]. That solution is not efficient due to its random shuffling procedure. Saxena et al. [18] propose a non-interactive solution by using bivariate polynomials, named *bivariate admission control*, but this protocol is secure only under the discrete logarithm assumption. Our solution, called *enrollment protocol*, is an efficient new construction with unconditional security under the passive adversary model. We assume that this protocol is executed in a single time slot in our social secret sharing scheme.

We first show the Lagrange interpolation formula [20], and then present the enrollment protocol. Suppose q is a prime number and $x_1, x_2, \ldots, x_t$ are distinct elements in $\mathbb{Z}_q$. In addition, suppose $f_1, f_2, \ldots, f_t$ are elements in $\mathbb{Z}_q$. Then, there is a unique polynomial $f(x) \in \mathbb{Z}_q[x]$ of degree at most $t-1$ such that $f(x_i) = f_i$ for $1 \leq i \leq t$:

$$f(x) = \sum_{i=1}^{t} \prod_{1 \leq j \leq t, i \neq j} \frac{x - x_j}{x_i - x_j} \cdot f_i \tag{1}$$

1. First, each player P_i for $1 \leq i \leq t$ computes his corresponding Lagrange interpolation constant.

$$\gamma_i = \prod_{1 \leq j \leq t, i \neq j} \frac{k - j}{i - j} \quad \text{where } i, j, k \text{ represent players' } ids \tag{2}$$

2. After that, each participant P_i multiplies his share φ_i by his Lagrange interpolation constant, and randomly splits the result into t portions, i.e., a row in a *share-exchange matrix*.

$$\varphi_i \cdot \gamma_i = \partial_{1i} + \partial_{2i} + \cdots + \partial_{ti} \quad \text{for } 1 \leq i \leq t \tag{3}$$

3. Players exchange ∂_{ji}'s accordingly through pairwise channels. Therefore, each P_j holds t values, i.e., a column in the share-exchange matrix. P_j adds them together and sends the result to P_k.

$$\sigma_j = \sum_{i=1}^{t} \partial_{ji} \quad \text{where } \partial_{ji} \text{ is the } j^{th} \text{ share-portion of the } i^{th} \text{ participant} \tag{4}$$

4. Finally, player P_k adds these values σ_j for $1 \leq j \leq t$ together to compute his share φ_k.

$$\varphi_k = \sum_{j=1}^{t} \sigma_j \tag{5}$$

First, suppose $t-1$ of t cooperating participants collude. In this case, colluders have access to all entries of $t-1$ rows. In addition, they also know $t-1$ entries of the single unknown row because $t-1$ columns belong to them. Therefore, just one entry remains unknown which prevents colluders to find the newcomer's share and consequently the secret (as presented in Example 7, if P_1 and P_2 collude, $@_{33} = 5$ in the third row remains unknown).

Second, suppose $t-2$ of t cooperating participants plus the newcomer collude. In this case, colluders have access to all entries of $t-2$ rows, in addition, they also know $t-2$ entries of two unknown rows because $t-2$ columns belong to them. Therefore, four entries remain unknown. On the other hand, the newcomer also knows the summation of column's entries for all columns, as a consequence, he can just construct two equations with four unknowns which does not reveal any information about the secret (as presented in Example 7, if P_1 and the newcomer P_4 collude, $@_{22} = 1$ and $@_{32} = 2$ in the second row and $@_{23} = 2$ and $@_{33} = 5$ in the third row remain unknown and P_4 can only construct the following two equations: $1 + @_{22} + @_{23} = 4$ and $1 + @_{32} + @_{33} = 8$). □

Phase-(II):

1. To update shares, each player P_u generates a random polynomial $g^u(x) \in \mathbb{Z}_q[x]$ of degree $t-1$ with a zero constant term.
2. Player P_u then sends w_i shares to P_i for $1 \le i \le n$, as shown below, where $\#_{ij} = im - m + j$ and m is the maximum weight of any participant.

$$ {}^{u}_{ij} = g^u(\#_{ij}) \text{ for } 1 \le j \le w_i $$

3. Finally, each player P_i updates his share by adding up the auxiliary shares ${}^{u}_{ij}$ to his share $'_{ij}$.

$$ '_{ij} = '_{ij} + \sum_{u=1}^{n} {}^{u}_{ij} \text{ for } 1 \le j \le w_i $$

Since the constant terms of $g^u(x)$-s are zero, therefore, the secret remains the same and shares of players are updated in order to overcome the mobile adversary [11]. As we mentioned, inactivated *ids* do not receive any shares at this stage, i.e., they are disenrolled.

4.3 Secret Recovery ($\mathcal{R}ec$)

As stated earlier, authorized players are able to recover the secret if their total weight is equal or greater than the threshold, i.e., $\sum_{P_i \in} w_i \ge t$. In this case, players $P_i \in$ send their shares $'_{ij}$ for $1 \le j \le w_i$ to a selected participant to reconstruct $f(x)$ by Lagrange interpolation, consequently, the secret $f(0) =$ is recovered.

Theorem 9. *The social secret sharing scheme $S^4(Sha; Tun; Rec)$ presented in Section 4 is unconditionally secure under the passive mobile adversary model.*

Proof. The security of Sha and Rec is the same as the security of the Shamir's secret sharing scheme [19]. The security of the Tun depends on the share renewal step. The first phase is secure as shown in Theorem 8. The second phase is also proven to be secure as illustrated in [11].

5 Active Adversary Model Construction

In this section, we consider the active adversary model, where players may deviate from protocols or collude to reconstruct the secret. We review the verifiable and proactive secret sharing scheme, first proposed in [21] (a flaw in the scheme was fixed in [7]). We modify those protocols accordingly to fit them to our social secret sharing scheme.

First of all, the pairwise check is changed since each participant has multiple shares rather than a single share. Second, the *recovery* protocol is used to generate new shares for newly activated *ids*.

4. Each player P_i computes a subset $f1; :::; ng$ such that any ordered pair $(i; j)$ 2 is not broadcasted. If j j n jrj, then P_i outputs $ver_i = 1$, otherwise, P_i outputs $ver_i = 0$.

The dealer erases all the data on his end if at least n jrj players output $ver_i = 1$, otherwise, he reboots the system for another initialization. It is worth mentioning that, this scheme can tolerate a dishonest dealer. Moreover, a corrupted player may act honestly during Sha because of a future harmful plan, therefore, consists of uncorrupted players and possibly malicious players who act honestly during the initialization.

5.2 Social Tuning ($\mathcal{T}un$)

This section is similar to its counterpart in the passive adversary model construction (Section 4). The only difference is the share renewal stage.

Share Renewal. This stage consists of two phases. In the first one, initial shares for newcomers or newly activated *ids* of existing players are generated, i.e., they are enrolled. Then, in the second phase, players proactively update their shares, while disenrolled *ids* do not receive any updates.

Phase-(I):

1. Each player P_i where i 2 sends $'_{ik}(!^{\#_{jl}})$ for 1 k w_i to P_j in order to generate his l^{th} shares, that is, $'_{jl}(x)$.
2. After that, player P_j computes a polynomial $'_{jl}(x)$ such that $'_{jl}(!^{\#_{ik}}) = '_{ik}(!^{\#_{jl}})$ for at least n $2jrj$ values of i.

s2J/F21 10.9091 1step/F12 7.9701 8 5f Td05 [50 59] TJ/F(1#5)]U1/F15 10.9091 Tf

In fact, share $'_{jl}(x)$ is constructed through the interpolation of pairs $(!^{\#_{ik}}; '_{ik}(!^{\#_{jl}}))$ in the second step. We explain the main reason behind the condition n $2jrj$ in Section 5.3.

Phase-(II):1. EacT(e)-382473dateir

6. Then, other players P_j excluding the conflicting parties P_u and P_i, check ${}^{u}_{ic}(!^{\#_{jl}}) \overset{?}{=} {}^{u}_{jl}(!^{\#_{ic}})$ and broadcast yes or no. If, for every broadcasted ${}^{u}_{ic}(x)$, *at least* $j\ \ j\ \ \ jrj$ 1 players broadcast yes, then P_u is not malicious. In this case, if P_i has a share ${}^{u}_{ic}(x)$ different from the one that P_u has broadcasted, he stores the broadcasted one.
7. Finally, each participant P_i first updates the list of good players who are not found guilty in the previous step, and then updates his shares for 1 k w_i as follows:

$$'_{ik}(x) = '_{ik}(x) + (x + !^{\#_{ik}}) \mathop{X}_{u2} {}^{u}_{ik}(x)$$

5.3 Secret Recovery ($\mathcal{R}ec$)

Players are able to recover the secret at any time by performing the following recovery protocol.

1. Each player P_i where $i\ 2$ sends $'_{ik}(0)$ for 1 k w_i to a selected participant P_j, that is, the constant terms of shares.
2. After that, the selected player P_j computes a polynomial $f^0(0; y)$ such that $f^0(0; !^{\#_{ik}}) = '_{ik}(0)$ for at least n $2jrj$ values of i.
3. In fact, $f^0(0; y)$ is part of the original symmetric polynomial $f(x; y)$, therefore, the selected P_j computes the secret $= f^0(0; 0)$.

As we mentioned, the scheme itself can tolerate jrj dishonest players. In addition, a dishonest dealer may cheat on jrj of honest players during Sha in order to eliminate them from the scheme. As a result, the set of good players has at least n $2jrj$ members. Therefore, an error correction technique, such as the one proposed in [17], can be used to find the maximum consistent set of shares for the interpolation of $f^0(0; y)$.

Theorem 10. *The social secret sharing scheme* $S^4(Sha; Tun; Rec)$ *presented in Section 5 is unconditionally secure under the active mobile adversary model.*

Proof. The security proofs of the modified protocols are the same as the ones presented in [21, 7].

6 Conclusion

We introduced the notion of a *social secret sharing scheme*, in which a player's weight are adjusted based on his reputation and behaviors over time. We demonstrated two constructions based on the passive and active mobile adversary models.

The proposed construction has a variety of desirable properties: it is *unconditionally secure*, meaning that it does not rely on any computational assumptions; *proactive*, refreshing shares at each cycle without changing the secret; *dynamic*, allowing changes to the access structure after the initialization; *weighted*, allowing the cooperative players to gain more authority in the scheme; and *verifiable* in the case of the active adversary model.

In addition, the proposed scheme gradually reduces the influence of unreliable participants due to the *self-reinforcement* property of social interactions among players. In other words, players collaborate with those whom they really trust; conversely, they tend not to cooperate with those whom they do not trust. This issue creates an increasing gap between reliable and unreliable players unless a participant undergoes a sustained change in his behavior. Applications of such a paradigm are: electronic auctions with private bids running by intelligent agents, joint signature, and shared decryption keys.

References

[1] Ben-Or, M., Goldwasser, S., and Wigderson, A. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *STOC* (1988), ACM, pp. 1{10.

[2] Benaloh, J. C., and Leichter, J. Generalized secret sharing and monotone functions. In *CRYPTO* (1988), S. Goldwasser, Ed., vol. 403 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 27{35.

[3] Blakley, G. R. Safeguarding cryptographic keys. In *National Computer Conference, New York* (Montvale, NJ, USA, 1979), R. E. Merwin, J. T. Zanca, and M. Smith, Eds., vol. 48 of *AFIPS Conference proceedings*, AFIPS Press, pp. 313{317.

[4] Blundo, C., Cresti, A., Santis, A. D., and Vaccaro, U. Fully dynamic secret sharing schemes. *Theoretical Computer Science 165*, 2 (1996), 407{440.

[5] Boneh, D., and Franklin, M. Efficient generation of shared rsa keys. *Journal of ACM 48*, 4 (2001), 702{722.

[6] Chor, B., Goldwasser, S., Micali, S., and Awerbuch, B. Verifiable secret sharing and achieving simultaneity in the presence of faults. In *FOCS* (1985), IEEE, pp. 383{395.

[7] D'Arco, P., and Stinson, D. R. On unconditionally secure robust distributed key distribution centers. In *Advances in Cryptology, Proceedings of ASIACRYPT '02, Lecture Notes in Computer Science* (2002), Springer-Verlag, pp. 346{363.

[8] Du, W., and Atallah, M. J. Secure multi-party computation problems and their applications: a review and open problems. In *NSPW '01: Proceedings of the workshop on new security paradigms* (2001), ACM, pp. 13{22.

[9] Goldwasser, S. Multi party computations: past and present. In *PODC '97: Proceedings of the 16th Annual ACM Symposium on Principles of Distributed Computing* (1997), ACM, pp. 1{6.

[10] Harkavy, M., Tygar, J. D., and Kikuchi, H. Electronic auctions with private bids. In *WOEC'98: Proceedings of the 3rd Conference on USENIX Workshop on Electronic Commerce* (1998), USENIX Association, pp. 61{74.

[11] Herzberg, A., Jarecki, S., Krawczyk, H., and Yung, M. Proactive secret sharing or: How to cope with perpetual leakage. In *CRYPTO* (1995), D. Coppersmith, Ed., vol. 963 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 339{352.

[12] Jøsang, A., Ismail, R., and Boyd, C. A survey of trust and reputation systems for online service provision. *Decision Support Systems 43*, 2 (2007), 618{644.

[13] Mui, L., Mohtashemi, M., and Halberstadt, A. A computational model of trust and reputation. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences* (2002), pp. 2431{2439.

[14] Nojoumian, M., and Lethbridge, T. A New Approach for the Trust Calculation in Social Networks. In *E-business and Telecommunication Networks: 3rd International Conference on E-Business, Selected Papers* (2008), vol. 9, Springer, pp. 64{77.

[15] Ostrovsky, R., and Yung, M. How to withstand mobile virus attacks (extended abstract). In *PODC '91: Proceedings of the 10th Annual ACM Symposium on Principles of Distributed Computing* (1991), ACM, pp. 51{59.

[16] Rabin, T., and Ben-Or, M. Verifiable secret sharing and multiparty protocols with honest majority. In *STOC* (1989), ACM, pp. 73{85.

[17] Rees, R. S., Stinson, D. R., Wei, R., and van Rees, G. H. J. An application of covering designs: determining the maximum consistent set of shares in a threshold scheme. *Ars Comb. 53* (1999), 225{237.

[18] Saxena, N., Tsudik, G., and Yi, J. H. Efficient node admission for short-lived mobile ad hoc networks. In *Proceedings of the 13th IEEE International Conference on Network Protocols* (2005), IEEE Computer Society, pp. 269{278.

[19] Shamir, A. How to share a secret. *Communications of the ACM 22*, 11 (1979), 612{613.

[20] Stinson, D. R. *Cryptography: Theory and Practice,Third Edition*. CRC Press, 2005.

[21] Stinson, D. R., and Wei, R.