# Collusion Free Protocol for Rational Secret Sharing

**Amjed Shareef**
Department of Computer Science and Engineering
Indian Institute of Technology Madras, Chennai, India
Email: amjedshareef@gmail.com

**Abstract**

We consider the *rational secret sharing problem* introduced by Halpern and Teague[1], where players prefer to get the secret rather than not to get the secret and with lower preference, prefer that as few of the other players get the secret. Some positive results have been derived by Kol and Naor[3] by considering that players only prefer to learn. They have proposed an efficient $m$-out-of-$n$ protocol for rational secret sharing without using cryptographic primitives. Their solution considers that players are of two types; one player is the short player and the rest of the players are long players. But their protocol is susceptible to coalitions if the short player colludes with any of the long players. We extend their protocol, and propose a completely collusion free, $\varepsilon$-Nash equilibrium protocol, when $n \geq 2m-1$, where $n$ is the number of players and $m$ is the number of shares needed to construct the secret.

## 1 Introduction

In a secret sharing scheme there is a unique player called the *dealer* (player 0) who wants to share a secret $s$ among $n$ players, $p_1, \ldots, p_n$. The dealer sends every player a share of the secret in such a way that any group of $m$ (threshold value) or more than $m$ players can together reconstruct the secret but no group of fewer than $m$ players can reconstruct the secret. Such a system is called an $(m, n)$-threshold scheme or an $m$-out-of-$n$ Shamir scheme[4]. Halpern and Teague[1] introduced the problem of *rational secret sharing* assuming that the players are rational, where each player behaves in a selfish manner. Each player has his own preferences and utility function (the profit he gets). He always tries to maximize his profits and behaves accordingly. In brief, every player primarily prefers to get the secret rather than to not get it and secondarily, prefers that the fewer of the other players that get it, the better. A rational player follows the protocol only if it increases his expected utility.

This impossibility result is proved by Halpern and Teague[1]. They show that rational secret sharing is not possible with any mechanism that has a fixed running time. This was shown using iterated deletion of weakly dominated strategies (the strategy of not sending the share weakly dominates the strategy of sending the share). Kol and Naor[3] proposed an efficient solution to the problem in the form of a strict rational secret sharing protocol in the presence of a simultaneous broadcast channel (on which all players broadcast simultaneously). The solution is non-cryptographic, i.e., it does not use any cryptographic primitives, and it also considers the learning preferring property where players prefer to learn rather than to not learn. The protocol in [3] is collusion free, even if $(m-1)$ long players collude they cannot get any advantage. But if a short player colludes with even one long player, then they can learn which iteration is the definitive iteration and thus, can learn the which iteration contains the secret. Indeed, colluded players prevent other players from learning the secret by not following the protocol in the sense that they do not broadcast in the iteration prior to the definitive iteration, causing the other players to abort. A completely collusion free protocol for rational secret sharing works even if any $m-1$ players collude. The completely collusion free protocol is presented in [2] by Kol and Naor by making cryptographic assumptions. We propose a completely collusion free, $\varepsilon$-Nash equilibrium protocol, when $n \geq 2m-1$ without making any cryptographic assumptions, by incorporating changes in the original protocol by Kol and Naor[3]. Our protocol is completely collusion free if $n \geq 2m-1$, otherwise it is equivalent to Kol and Naor's[3] protocol. $\varepsilon$-Nash equilibrium has defined as follows.

**$\varepsilon$-Nash Equilibrium:** A behavioural strategy profile $\sigma$ for the game $\Gamma$ is said to be an $\varepsilon$-Nash equilibrium if for every $i \in N$ and every behavioural strategy $\sigma_i'$, it holds that $u_i(\sigma_i, \sigma_{-i}) + \varepsilon \geq u_i(\sigma_i', \sigma_{-i})$.

**Authentication:** We use authentication methods similar to those used in the paper [3]. Let us say that the dealer chooses the message $x$ as player $i'$s value. The dealer randomly choose $s_i, b_i \in \mathbb{F}$, $b_i \neq 0$, such that $c_i = (b_i x + s_i) \in \mathbb{F}$. The player $i$ gets the value $s_i$, the tag and each and every other player gets $< b_i, c_i >$. The player $i$ broadcasts his value $s_i$ along with the message $x$. The other players can verify the correctness of the value $x$ with $s_i, b_i$ and $c_i$.

## 2 Protocol

We consider a Synchronous Broadcast Channel (SBC) that is present and connects all the $n$ players in such a way that at every clock tick, every player may broadcast a message over the channel and receive other players' messages. The reason the STOC 08 protocol was not collusion-resilient was that if any long player colludes with the short player, then they both will get the secret and prevent others from doing so by not broadcasting in the iteration prior to the definitive iteration. But by distributing the masked secret as an $(m-1)$-out-of-$n$ Shamir share, we protect the protocol against this type of collusion. Consider the case where one short player and $(m-2)$ long players collude. If the players in the coalition do not send their shares in the iteration before the definitive iteration, then they cannot get the shares of the masked secret and hence reconstruct the masked secret and by extension the secret. Therefore they will their shares in the iteration prior the to definitive iteration, but they may not send their shares in the definitive iteration. Even if they decide to remain quite, the remaining $m$ players will broadcast and in this way everyone will get the secret. In the case where $(m-1)$ long players collude, they will not know which iteration is definitive, and will continue to broadcast in each iteration because they are rational. Thus, in any case, all players will get the secret regardless of whether or not a collusion is formed and also regardless of how many members form that collusion and who is present in that collusion.

**The dealer's protocol:**

We make two changes to the dealer's protocol. One change is that instead of distributing the masked secret to each player, we distribute an $(m-1)$-out-of-$n$ Shamir share of the masked secret to each player. The second change pertains to the authentication information. The "tag" of the player's element, which allows him to prove the authenticity of the previous elements in the present cell, and the "hash function", which allows him to check the veracity of the elements in the corresponding cells of the other vectors, are chosen such that the probability that the element supplied by another players is at least $1-\varepsilon'$ for $\varepsilon' = min\{\beta, \frac{\varepsilon}{U_{max}}\}$. Here, $U_{max}$ is an upper bound on the payoffs that the player may receive and $\beta$, which depends on the utility functions, is the parameter to a geometric distribution used to calculate share size as well as number of stages in each iteration.

**The player's reconstruction protocol:**

This is similar to Kol and Naor's[3] protocol. Please refer to Table 1 for a more detailed account.

Our protocol is an $\varepsilon$-rational protocol for the following reason. When a coalition containing the short player is formed, the coalition will know which iteration is the definitive iteration. Since they will definitely get the secret in the definitive iteration provided that the other players follow the protocol, they have no need to be truthful in the final iteration. Therefore they will try to cheat by sending incorrect messages, which they hope will pass the authentication check, in an effort to fool the other players into thinking that the current iteration is not the definitive iteration. Our scheme incorporates an authentication mechanism which ensures that the attempt by the coalition to cheat will fail with a probability at least $1-\frac{\varepsilon}{U_{max}}$, where $U_{max}$ is an upper bound on the payoffs that the players may receive. Hence our protocol is $\varepsilon$-rational.

**Theorem 1** *Let $Y$ be a finite set of secrets with distribution $\mathcal{D}$, and let $(u_i)_{i\epsilon N}$ be learning preferring utility functions. For all $2 \le m \le n$, where $n \ge 2m-1$, the scheme described above is a simultaneous $\varepsilon$-**rational** $m$-out-of-$n$ secret sharing scheme for $Y$ with respect to linger avoiding strategies. It has expected running time $O(\frac{1}{\beta^2})$, and expected share size $O(\frac{1}{\beta}(\log\frac{1}{\beta} + \log\frac{U_{max}}{\varepsilon}))$, where $\beta$ is a parameter derived from utilities and distribution $D$, similar to [3].*

*Proof:* If $n < 2m-1$ then the protocol gracefully reduces to the one in [3]. In any case, the coalition, if it includes the short player, will know the definitive iteration. However, as the secret can only be reconstructed using $m-1$ shares, any deviation by the coalition prior to the definitive iteration leads to the quitting of the protocol by all the players,and hence the coalition, along with the other players, fail to learn the secret with probability 1. As players follow learning preferring property, it is strictly dominating for the colluded players, to follow the protocol correctly. □

**Acknowledgement:** The authors would like to acknowledge William Kumar Moses, Jr., for his insightful remarks on theorem.

## References

[1] Joseph Halpern and Vanessa Teague. Rational secret sharing and multiparty computation: extended abstract. In *STOC '04: Proceedings of the thirty-sixth annual ACM symposium on Theory of computing,*

**Player$_i$(share)**
Set secret_revealed ← FALSE and cheater_detected← FALSE.
**Repeat until** secret_revealed←TRUE **or** cheater_detected←TRUE.

- **If your share ended:**
  - Keep silent.
  - If at least $m-1$ people have broadcasted and passed their messages passed the authentication check, secret_revealed←TRUE.
  - If less than $m-1$ people have broadcasted correctly, cheater_detected←TRUE.
- **If your share did not end:** use the corresponding cell of share to check whether this is the last stage of the present iteration.
  - If this is not the last stage:
    - Keep silent
    - If anyone has broadcasted, cheater_detected←TRUE.
  - If this is the last stage:
    - Broadcast the player's tag and shares of the masked secret, random mask, and indicator as they appear in the corresponding cell of the share.
    - If all players sent:
      - If the reconstructed indicator shows that this is the definitive iteration, secret_revealed←TRUE.
    - If one or more players did not send:
      - If number of valid messages broadcasted by others $< m-1$, cheater_detected←TRUE.
      - Else
        If the reconstructed indicator shows that this is not the definitive iteration, cheater_detected←TRUE.
        Else, secret_revealed←TRUE.
- **Leave the game:** Reconstruct the masked secret using shares broadcasted in the present iteration. Reconstruct the mask using shares broadcasted in the previous iteration. Now construct the possible secret by subtracting the mask from the masked secret. Quit and output the possible secret.

Table 1: Player $i$'s reconstruction protocol


pages 623–632, New York, NY, USA, 2004. ACM.

[2] Gillat Kol and Moni Naor. Cryptography and game theory: Designing protocols for exchanging information. In *TCC '08*, pages 320–339, 2008.

[3] Gillat Kol and Moni Naor. Games for exchanging information. In *STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 423–432, New York, NY, USA, 2008. ACM.

[4] A. Shamir. How to share a secret. *Commun. ACM*, 22:612–613, 1979.