Des. Codes Cryptogr. (2009) 52:15–24
DOI 10.1007/s10623-008-9265-8

Detection and identification of cheaters in (t, n) secret sharing scheme

Lein Harn · Changlu Lin

Received: 15 October 2008 / Revised: 4 December 2008 / Accepted: 22 December 2008 /
Published online: 11 January 2009

Abstract In a (t, n) secret sharing scheme, a secret s is divided into n shares and shared among a set of n shareholders by a mutually trusted dealer in such a way that any t or more than t shares will be able to reconstruct this secret; but fewer than t shares cannot know any information about the secret. When shareholders present their shares in the secret reconstruction phase, dishonest shareholder(s) (i.e. cheater(s)) can always exclusively derive the secret by presenting faked share(s) and thus the other honest shareholders get nothing but a faked secret. Cheater detection and identification are very important to achieve fair reconstruction of a secret. In this paper, we consider the situation that there are more than t shareholders participated in the secret reconstruction. Since there are more than t shares (i.e. it only requires t shares) for reconstructing the secret, the redundant shares can be used for cheater detection and identification. Our proposed scheme uses the shares generated by the dealer to reconstruct the secret and, at the same time, to detect and identify cheaters. We have included discussion on three attacks of cheaters and bounds of detectability and identifiability of our proposed scheme under these three attacks. Our proposed scheme is an extension of Shamir's secret sharing scheme.

Communicated by P. Wild.

L. Harn
Department of Computer Science and Electrical Engineering, University of Missouri, Kansas City, MO 64110-2499, USA
e-mail: harnl@umkc.edu

C. Lin (✉)
State Key Laboratory of Information Security, Graduate University of Chinese Academy of Sciences, Beijing 100049, People's Republic of China
e-mail: lincl@is.ac.cn

C. Lin
Key Laboratory of Network Security and Cryptology, Fujian Normal University, Fujian 350007, People's Republic of China

Keywords Secret sharing scheme · Detection · Identification · Consistency · Majority voting

Mathematics Subject Classification (2000) 94A62

1 Introduction

Secret sharing schemes were originally introduced by both Blakley [3] and Shamir [15] independently in 1979 as a solution for safeguarding cryptographic keys and have been studied extensively in the literatures. In a secret sharing scheme, a secret s is divided into n *shares* and shared among a set of n shareholders by a mutually trusted *dealer* in such a way that any t or more than t shares will be able to reconstruct this secret; but fewer than t shares cannot know any information about s. Such a scheme is called a (t, n) secret sharing, denoted as (t, n)-SS.

Shamir's (t, n)-SS scheme is very simple and efficient to share a secret among n shareholders. However, when the shareholders present their shares in the secret reconstruction phase, dishonest shareholder(s) (i.e. cheater(s)) can always exclusively derive the secret by presenting faked share(s) and thus the other honest shareholders get nothing but a faked secret. It is easy to see that the Shamir's original scheme does not prevent any malicious behavior of dishonest shareholders during secret reconstruction. Cheater detection and identification are very important to achieve fair reconstruction of a secret.

There are many research papers in the literatures to investigate the problem of cheater detection and/or identification for secret sharing schemes. Some of them [1,4–6,8,9,12–14,18] consider that there are exactly t shareholders participated in the secret reconstruction. In order to enable each shareholder the ability of cheater detection and identification, the dealer needs to generate and distribute additional information, such as using check vectors and certificate vectors for each shareholder. Some other papers [2,11] proposed to design a secret sharing scheme based on an error-correcting code in which faked shares can be treated as error codes to be detected and corrected based on coding technique. For example, McEliece and Sarwate [11] described to construct a secret sharing scheme based on Reed-Solomons code. The performance of their scheme can guarantee that the secret is correctly calculated by honest participants with any group of $t + 2e$ participants including at most e cheaters. There are some papers [7,10,17] to propose secret sharing schemes based on well-known computational assumptions. Since these schemes are conditionally secure, the ability to detect and identify cheaters are much stronger than those schemes that are unconditionally secure. For example, the scheme based on RSA assumption [10] enables any honest participant to detect and identify cheaters even when all of the other participants corrupt together.

In this paper, we use a different approach to prevent cheaters. We consider the situation that there are more than t shareholders participated in the secret reconstruction. Since there are more than t shares (i.e. it only requires t shares) for reconstructing the secret, the redundant shares can be used for cheater detection and identification. Our proposed scheme uses the shares generated by the dealer to reconstruct the secret and, at the same time, to detect and identify cheaters. Simmons [16] has suggested to use the same method to detect cheaters. In this paper, we have included discussion on possible attacks of cheaters and bounds of detectability and identifiability of our proposed scheme under these attacks. One example is included to illustrate our scheme.

The rest of this paper is organized as follows. In the next section, we provide some preliminaries. In Sect. 3, we introduce cheater detection and identification scheme. In Sect. 4,

we describe attacks of cheaters. In Sect. 5, we analyze our scheme under three attacks and calculate bounds of detectability and identifiability of our proposed scheme. One example is included in Sect. 6. We conclude in Sect. 7.

2 Preliminaries

In this section, we introduce some fundamental backgrounds.

2.1 Shamir's (t, n)-SS

In Shamir's (t, n)-SS based on Lagrange interpolating polynomial, there are n shareholders $\mathscr{P} = \{P_1, \ldots, P_n\}$ and a mutually trusted dealer D, and the scheme consists of two algorithms:

1. *Share generation algorithm* the dealer D first picks a polynomial $f(x)$ of degree $t-1$ randomly: $f(x) = a_0 + a_1 x + \cdots + a_{t-1}x^{t-1}$, in which the secret $s = a_0$ and all coefficients $a_0, a_1, \ldots, a_{t-1}$ are in a finite field $\mathbb{F}$, and D computes:

$$s_1 = f(1), s_2 = f(2), \ldots, s_n = f(n).$$

   Then, the algorithm outputs a list of n shares $(s_1, s_2, \ldots, s_n)$ and distributes each share s_i to corresponding shareholder P_i secretly.
2. *Secret reconstruction algorithm* this algorithm takes any t shares $(s_{i_1}, \ldots, s_{i_t})$ where $\{i_1, \ldots, i_t\} \subset \{1, 2, \ldots, n\}$ as inputs, and outputs the secret s.

We note that the above scheme satisfies the basic requirements of secret sharing scheme as follows: (1) With knowledge of any t or more than t shares, it can reconstruct the secret s easily; (2) With knowledge of fewer than t shares, it cannot get *any* information about the secret s. Shamir's scheme is *information-theoretically secure* since the scheme satisfies these two requirements without making any computational assumption. For more information on this scheme, readers can refer to the original paper [15].

2.2 Consistency

Let S be the domain of a secret and T be the domain of shares corresponding to the secret. We say that the function $F_I\colon T^t \to S$ is an *induced* function of the (t, n)-SS for each subset $I \subset \{1, 2, \ldots, n\}$ with $|I| = t$. This function defines the secret s as follows with any set of t shares $s_{i_1}, \ldots, s_{i_t}$.

$$s = F(I) = F_I(s_{i_1}, \ldots, s_{i_t}), \quad \text{where} \quad I = \{i_1, \ldots, i_t\}.$$

Actually, the sharing secret s is computed from the polynomial $f_I(x)$ which is constructed by the interpolation of the points $(i_1, s_{i_1}), \ldots, (i_t, s_{i_t})$.

Definition 1 (*Consistency*) In a (t, n)-SS scheme, let $m \geq t$, a set of m shares $s_1, s_2, \ldots, s_m$ is said to be consistent if any subset containing t shares of the set reconstructs the same secret. Formally, let $\mathscr{T} = \{T_1, \ldots, T_u\}$ be the set of u elements where each element contains t shares of the set of m shares, where $u = \binom{m}{t}$ denotes the total number of these subsets, then we have

$$s^i = F(T_i) = F_{T_i}(s_{i_1}, \ldots, s_{i_t}), \quad \text{where } i = 1, \ldots, u.$$

$s_1, s_2, \ldots, s_m$ are consistent which means that

$$s^1 = \cdots = s^u.$$

Moreover, if $s_1, s_2, \ldots, s_m$ are consistent, then the reconstructed secrets s^i for $i = 1, \ldots, u$ are all identical.

Remark 1 In fact, since all shares are generated by a polynomial in Shamir's (t, n)-SS scheme, to check whether m, where $m \geq t$, shares are consistent or not, we only need to check whether the interpolation of m points $(1, s_1), \ldots, (m, s_m)$ yields a polynomial with degree $t - 1$ or not. If this condition is satisfied, we can conclude that all secrets s^i for $i = 1, \ldots, u$ are identical and all shares are consistent. This approach to check shares' consistency only requires *one* computation instead of u combinations of t out of m shares.

2.3 Majority of secrets

If the shares $s_1, \ldots, s_m$ are *inconsistent*, it is easy to see that secrets s^i for $i = 1, \ldots, u$ reconstructed by combinations of t out of m shares are not identical. Then, we can divide the set $U = \{s^1, \ldots, s^u\}$ containing all reconstructed secrets into several mutually disjoint subsets U_i, for $i = 1, \ldots, v$. Each subset contains same secret. These subsets satisfy following conditions.

- $U = U_1 \cup \cdots \cup U_v$, where $U_i = \{s^{i_1}, \ldots, s^{i_{w_i}}\}$ and $s^{w_i} = s^{i_1} = \cdots = s^{i_{w_i}}$;
- $U_k \cap U_l = \emptyset$ for $1 \leq k, l \leq v$ and $k \neq l$.

Definition 2 (*Majority of secrets*) For all subsets U_i for $i = 1, \ldots, v$ as defined previously, set $w_i = |U_i|$ and $w_z = \max_i\{w_i\}$, then the secret s^{w_z} is said to be the majority of secrets.

3 Our algorithms

In this section, we first describe our approach to detect and identify cheaters. Then, we propose our scheme which is based on Shamir's (t, n)-SS scheme. One unique feature of our proposed scheme is that we use the same share for secret reconstruction to detect and identify cheaters. Our scheme is an extension of Shamir's (t, n)-SS scheme.

- *Method for detecting cheaters* In Shamir's (t, n)-SS scheme, a $t - 1$ degree interpolating polynomial can be uniquely reconstructed based on t shares. Thus, if there are more than t shares and there is no faked share, according to Def. 1, a consistent polynomial should be reconstructed for all combinations of t shares. Cheater detection is determined by detecting inconsistent polynomials (or secrets) among all reconstructed secrets. However, cheaters can collaborate to determine their faked shares to fool honest shareholders to believe that a faked secret is a real secret. In Sec. 5, we will discuss bounds of detectability of our proposed detecting scheme under three attacks as presented in next section.
- *Method for identifying cheaters* When cheaters have been detected, there are inconsistent reconstructed polynomials (or secrets) for all combinations of t shares. Among all reconstructed secrets, if the legitimate secret is the majority of secrets as we have defined in Def. 2, we can use the *majority voting mechanism* to identify each faked share. We need to investigate conditions that the legitimate secret is the majority of secrets. In addition, we will discuss bounds of identifiability of our proposed identifying scheme under three attacks as presented in next section.

We use c to denote the number of faked shares and j $(n \geq j \geq t)$ to denote the number of participants in a secret reconstruction. There are $j - c$ legitimate shares in a secret reconstruction. We use $J = \{i_1, \ldots, i_j\} \subseteq \{1, \ldots, n\}$ to denote all participants, $\mathscr{T} = \{T_1, \ldots, T_u\}$

to denote all subsets with t participants of J where $u = \binom{j}{t}$, H to denote the set of the honest participants, and C to denote the set of the cheaters. Our proposed scheme consists of the following algorithms.

1. *Share generation algorithm* this algorithm is same as Shamir's scheme.
2. *Secret reconstruction algorithm* this algorithm consists of following two sub-algorithms for cheater detection and cheater identification respectively.

Algorithm 1 (*Cheater detection*)

Input: $t, n, J, s_{i_1}, \ldots, s_{i_j}$

1. Compute an interpolated polynomial $f(x)$ of j points $(i_i, s_{i_1}), \ldots, (i_j, s_{i_j})$. Set the degree of $f(x)$ to be d.
2. If $d = t - 1$, then $s = f(0)$, and

Output: `There is no cheater` and `Secret is` s; otherwise

Output: `There are cheaters`.

Algorithm 2 (*Cheater identification*)

Input: $t, n, s, J, \mathscr{T}, s_{i_1}, \ldots, s_{i_j}$

1. For all $T_i \in \mathscr{T}$, compute $s^i = F(T_i)$ where $i = 1, \ldots, u$.
2. Divide $U = \{s^1, \ldots, s^u\}$ into v subsets U_i such that $U = U_1 \cup \cdots \cup U_v$ where $U_k \cap U_l = \emptyset$ for $1 \leq k, l \leq v$ and $k \neq l$, and $U_i = \{s^{i_1}, \ldots, s^{i_{w_i}}\}$ where $s^{w_i} = s^{i_1} = \cdots = s^{i_{w_i}}$.
3. Set $w_z = \max_i\{w_i\}$, and set $s = s^{w_z}$.
4. Pick $T_k \in \mathscr{T}$ such that $s = F(T_k) = F_{T_k}(s_{i_{k_1}}, \ldots, s_{i_{k_t}})$, and set $R = J - \{i_{k_1}, \ldots, i_{k_t}\}$.
5. Pick $i_r \in R$ orderly and remove it from R, and compute $s^r = F(s_{i_r}, s_{i_{k_2}}, \ldots, s_{i_{k_t}})$.
6. If $s^r = s$, then put i_r into H; otherwise put i_r into C.
7. Return Step 5 until $R = \emptyset$.

Output: `The cheater set is` C.

Remark 2 The computational complexity of algorithm 1 is $O(1)$ and the complexity of algorithm 2 is $O(j!)$, where $j \leq n$. We want to point out that n is the total number of shares in a secret sharing scheme and n is independent with the security of secret sharing scheme. In most secret sharing applications, n can be a small positive integer.

4 Attacks of cheaters

In this section, we consider three attacks of cheaters that are against our proposed detection and the identification scheme.

– *Type* 1 *attack* the cheaters of this type attack can be either honest shareholders who present their shares in error *accidentally* or dishonest shareholders who present their faked shares *without* any collaboration. Each faked share of this attack is just a random integer and is completely independent with other shares.

– *Type* 2 *attack* the cheaters of this type attack are dishonest shareholders who modify their shares on purpose to fool honest shareholders. In this type attack, we assume that all shareholders release their shares *synchronously*. Thus, cheaters can only collaborate among themselves to figure out their faked shares before secret reconstruction; but cannot modify their shares after knowing honest shareholders' shares (i.e. we assume that all shares must be revealed simultaneously). Under this assumption, only when the number of cheaters is larger than or equal to the threshold value t, the cheaters can implement an attack successfully to fool honest shareholders.
– *Type* 3 *attack* the cheaters of this type attack are dishonest shareholders who modify their shares on purpose to fool honest shareholders. In this type attack, we assume that all shareholders release their shares *asynchronously*. Since shareholders release their shares one at a time, the optimum choice for cheaters is to release their shares after all honest shareholders releasing their shares. The cheaters can modify their shares accordingly. We consider the worst-case analysis to determine the bounds of detectability and identifiability of our proposed scheme.

5 Algorithms analysis

In this section, we analyze our scheme proposed in Sect. 3 to detect and identify cheaters using the property of consistency and the notion of majority of secrets respectively. We also investigate the bounds of detectability and the identifiability of our proposed scheme under Type 1, Type 2 and Type 3 attacks.

Theorem 1 *Under type 1 attack, our proposed scheme can detect cheaters if* $j \geq t+1$*, and identify cheaters if* $j - c > t$.

Proof In our proposed scheme, the detection of cheaters is determined by detecting inconsistent secrets (or polynomials) among all reconstructed secrets. For j participating shareholders, there are $u = \binom{j}{t}$ secret reconstruction cases. In order to detect the cheaters in type 1 attack, it requires $j \geq t+1$. It is easy to see that if there exists any cheater, the reconstructed secrets s^i for $i = 1, \ldots, u$ are inconsistent. Thus, we can detect cheaters using algorithm 1.

The identification of cheaters is determined by the majority of secrets among all reconstructed secrets. When $j - c > t$, there are $\binom{j-c}{t}$ cases that will construct the legitimate secret. This legitimate secret is the *majority of secrets* and can be used in majority voting to identify all faked shares using algorithm 2. □

Theorem 2 *Under type 2 attack, our proposed scheme can detect cheaters if* $\{(c < t) \cap (j \geq t+1)\} \cup \{(c \geq t) \cap (j - c \geq t)\}$*, and identify cheaters if* $\{(c < t) \cap (j - c \geq t+1)\} \cup \{(c \geq t) \cap (j - c > c + t - 1)\}$.

Proof For type 2 attack, if $c \geq t$, cheaters can determine the secret before secret reconstruction. Cheaters can modify their shares accordingly without being detected by honest shareholders. This attack succeeds only when $j - c < t$. We give detail description of this attack. For example, we suppose that $c = t$, $j = 2t - 1$, and $s_1, \ldots, s_j$ are shares for the j participants $P_1, \ldots, P_j$ respectively. Then, there are $t - 1$ honest participants. Without loss

of generality, we assume $P_1, \ldots, P_{t-1}$ are honest and other participants are dishonest. These t cheaters first generate the interpolating polynomial $f(x)$ using their shares, and they can compute shares s_i, for $i = 1, \ldots, t-1$, using $f(x)$. Thus, these cheaters through these $t-1$ legitimate shares and a faked share s'_t. Then cheaters generate another $t-1$ faked shares $s'_{t+1}, \ldots, s'_j$ using $g(x)$. When all shareholders want to reconstruct the secret, $P_1, \ldots, P_{t-1}$ present legitimate shares $s_1, \ldots, s_{t-1}$ respectively, while the other shareholders present faked shares. The reconstructed secret is

$$s' = F(s_1, \ldots, s_{t-1}, s'_t)s = F(s_1, \ldots, s_{t-1}, s_t).$$

This implies that these cheaters successfully fool $P_1, \ldots, P_{t-1}$ who obtain a faked secret s'; however cheaters obtain the correct secret s. To avoid this attack, it requires $j - c \geq t$ when $c \geq t$.

The identification of the cheater is determined by the majority of secrets among all reconstructed secrets. When $j - c > t$, there are $\binom{j-c}{t}$ cases that can reconstruct the legitimate secret s since all involving shares are legitimate. When $c \geq t$, all cheaters can determine the secret before secret reconstruction. They can utilize up to $t-1$ legitimate shares and modify their shares accordingly. Cheaters first generate the interpolating polynomial $f(x)$ using their shares, and they can compute shares s_i, for $i = 1, \ldots, t-1$, using $f(x)$. Thus, these cheaters can choose a new $(t-1)$th interpolating polynomial $g(x)$ passing through these $t-1$ legitimate shares and a faked share s'_1. Then, cheaters generate additional $c-1$ faked shares $s'_2, \ldots, s'_c$ using $g(x)$. There are $\binom{c+t-1}{t}$ cases that reconstruct the illegitimate secret s' since there are $t-1$ legitimate shares and c illegitimate shares passing through $g(x)$. If the number $\binom{j-c}{t}$ is smaller than the number $\binom{c+t-1}{t}$, our scheme fails since the majority of secrets is a faked secret s'. In other words, when $c+t-1 > j-c$, cheaters can fail our proposed identifying scheme. Thus, it requires $j - c > c + t - 1$ to identify cheaters when $c \geq t$.

It is easy to see that if $c < t$, this attack is the same as type 1 attack. □

Theorem 3 *Under type* 3 *attack, our proposed scheme can detect cheaters if* $j - c \geq t$, *and identify cheaters if* $\{j \geq t+1\} \cap \{j - c > c + t - 1\}$.

Proof For type 3 attack, since shareholders release their shares asynchronously, the optimum choice for cheaters is to release their shares after knowing all honest shareholders' shares. Cheaters can modify their shares accordingly. This attack succeeds even if $c < t$. The detection of cheaters is determined by detecting inconsistent secrets among all reconstructed secrets. Since cheaters can modify their shares after knowing all honest shareholders' shares, it requires $j - c \geq t$ to ensure that reconstructed secrets are inconsistent if there is any faked share.

The identification of the cheater is determined by the majority of secrets among all reconstructed secrets. When $j - c > t$, there are $\binom{j-c}{t}$ cases that can reconstruct the legitimate secret s since all involving shares are legitimate. The cheaters can utilize up to $t-1$ legitimate shares and modify their shares accordingly. Cheaters first choose a new interpolating polynomial $g(x)$ passing through these $t-1$ legitimate shares and one faked share. Then, generate $c-1$ faked shares using $g(x)$. There are $\binom{c+t-1}{t}$ cases that can reconstruct the illegitimate secret since there are $t-1$ legitimate shares and c illegitimate shares passing

through $g(x)$. If the number $\binom{j-c}{t}$ is smaller than the number of cases $\binom{c+t-1}{t}$ of the illegitimate secret, our scheme fails since the majority of secrets is a faked secret. In other words, when $c+t-1 > j-c$, cheaters can fail our proposed identifying scheme. Thus, it requires $j-c > c+t-1$ to identify cheaters. □

Table 1 Detectability for $(4, n)$-SS Scheme for $j = 5$

	$c_{\max}$	Conditions used to determine $c_{\max}$
Type 1	**5**	$j \geq t+1$
Type 2	**3**	$\{(c < t) \cap (j \geq t+1)\} \cup \{(c \geq t) \cap (j-c \geq t)\}$
Type 3	**1**	$j-c \geq t$

Table 2 Identifiability for $(4, n)$-SS Scheme for $j = 9$

	$c_{\max}$	Conditions used to determine $c_{\max}$
Type 1	**4**	$j-c \geq t+1$
Type 2	**3**	$\{(c < t) \cap (j-c \geq t+1)\} \cup \{(c \geq t) \cap (j-c > c+t-1)\}$
Type 3	**2**	$\{j \geq t+1\} \cap \{j-c > c+t-1\}$

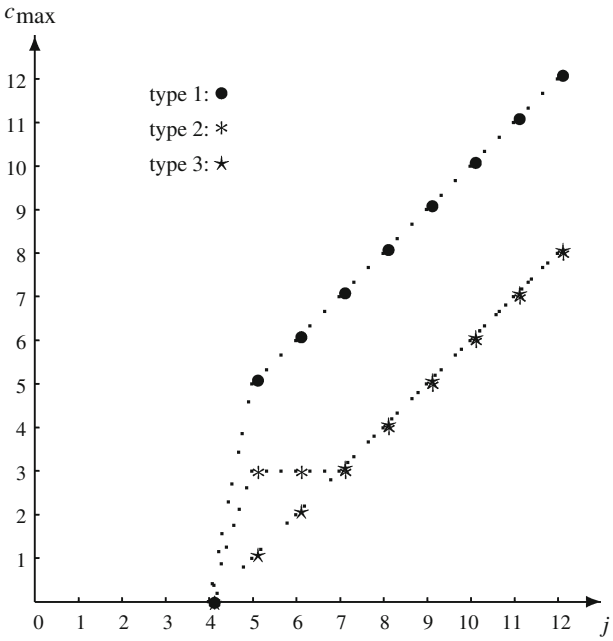

Fig. 1 Detectability for $(4, n)$-SS scheme

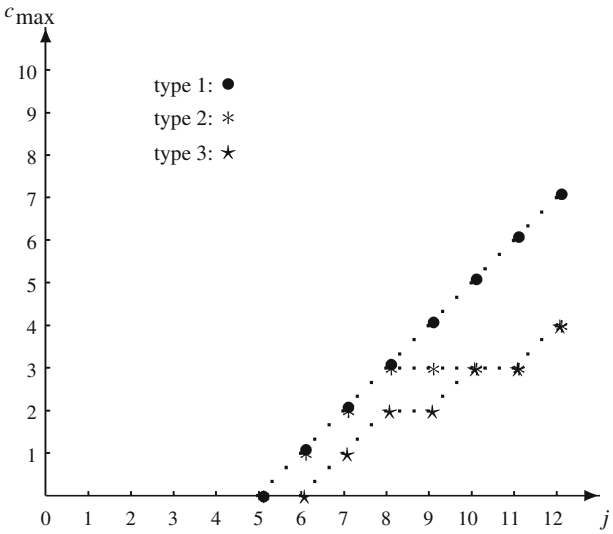

Fig. 2 Identifiability for $(4, n)$-SS scheme

6 Example

In this section, we illustrate our proposed scheme with a $(4, n)$-SS scheme. We denote $c_{\max}$ as the maximum number of cheaters when the number of participants j is fixed in our proposed scheme.

We summarize bounds for detectability and identifiability of our proposed scheme under three attacks in Tables 1 and 2. In Table 1, it shows that detectability of our proposed scheme decreases gradually from type 1 to type 3 for participants $j = 5$. Similarly, in Table 2, it shows that identifiability of our proposed scheme decreases gradually from type 1 to type 3 for participants $j = 9$. The decrement of detectability and identifiability from type 1 attack to type 3 attack is caused by the increment of attackers ability from type 1 attack to type 3 attack.

In Fig. 1, it illustrates that detectability of our proposed scheme is in proportion to the number of participants. Similarly, in Fig. 2, it illustrates that identifiability of our proposed scheme is in proportion to the number of participants.

7 Conclusions

In this paper, we consider the cases when there are more than t shareholders participated in secret reconstruction. Since there are more than t shares for reconstructing the secret, the redundant shares of a (t, n) secret sharing scheme can be used to detect and identify cheaters. We introduce the property of consistency and the notion of the majority of secrets to detect and identity cheaters. The bounds of detectability and identifiability under three attacks are presented. We utilizes shares for secret reconstruction to detect and identify cheaters. Our scheme is an extension of Shamir's secret sharing scheme.

References

1. Araki T.: Efficient (k, n) threshold secret sharing schemes secure against cheating from $n - 1$ cheaters. In: Proceedings of ACISP'07, LNCS, vol. 4586, pp. 13–142. Springer-Verlag (2007).
2. Bhndo C., De Santis A., Gargano L., Vaccaro U.: Secret sharing schemes with veto capabilities. In: Proceedings of the First French-Israeli Workshop on Algebraic Coding, LNCS, vol. 781, pp. 82–89. Springer-Verlag (1993).
3. Blakley G.R.: Safeguarding cryptographic keys. In: Proceedings of AFIPS'79, vol. 48, pp. 313–317 (1979).
4. Brickell E.F., Stinson D.R.: The detection of cheaters in threshold schemes. In: Proceedings of Crypto'88, LNCS, vol. 403, pp. 564–577. Springer-Verlag (1990).
5. Carpentieri M.: A perfect threshold secret sharing scheme to identify cheaters. Des. Codes Cryptogr. **5**(3), 183–187 (1995).
6. Carpentieri M., De Santis A., Vaccaro U.: Size of shares and probability of cheating in threshold schemes. In: Proceedings of Eurocrypt'93, LNCS, vol. 765, pp. 118–125. Springer-Verlag (1994).
7. Charnes C., Pieprzyk J., Safavi-Naini R.: Conditionally secure secret sharing scheme with disenrolment capability. In: Proceedings of CCS'94, pp. 89–95. ACM (1994).
8. He J., Dawson E.: Shared secret reconstruction. Des. Codes Cryptogr. **14**(3), 221–237 (1998).
9. Kurosawa K., Obana S., Ogata W.: t-cheater identifiable (k, n) secret sharing schemes. In: Proceedings of Crypto'95, LNCS, vol. 963, pp. 410–423. Springer-Verlag (1995).
10. Lin H.Y., Harn L.: A generalized secret sharing scheme with cheater detection. In: Proceedings of Asiacrypt'91, LNCS, vol. 739, pp. 149–158. Springer-Verlag (1991).
11. McEliece R.J., Sarwate D.V.: On sharing secrets and Reed-Solomon codes. Comm. ACM **24**, 583–584 (1981).
12. Ogata W., Kurosawa K., Stinson D.R.: Optimum secret sharing scheme secure against cheating. SIAM J. Discrete Math. **20**(1), 79–95 (2006).
13. Pieprzyk J., Zhang X.M.: Cheating prevention in linear secret sharing. In: Proceedings of ACISP'02, LNCS, vol. 2384, pp. 121–135. Springer-Verlag (2002).
14. Rabin T., Ben-Or M.: Verifiable secret sharing and multiparty protocols with honest majority. In: Proceedings of the 21st Annual ACM Symposium on the Theory of Computing, pp. 73–85 (1989).
15. Shamir A.: How to share a secret. Comm. ACM **22**(11), 612–613 (1979).
16. Simmons G.: An introduction to shared secret schemes and their applications. Sandia Report SAND 88-2298 (1988).
17. Tartary C., Wang H.: Dynamic threshold and cheater resistance for shamir secret sharing scheme. In: Proceedings of Inscrypt'06, LNCS, vol. 4318, pp. 103–117. Springer-Verlag (2006).
18. Tompa M., Woll H.: How to share a secret with cheaters. J. Cryptol. **1**(3), 133–138 (1989).