International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 14 (2018) pp. 11600-11605
© Research India Publications. http://www.ripublication.com

Linear (*t*, *n*) Secret Sharing Scheme based on Single Polynomial

Kenan Kingsley Phiri[1], Patrick Ali[1], Levis Eneya[1], Hyunsung Kim[1,2,*]

[1] *Department of Mathematical Sciences, Chancellor College, University of Malawi, P.O.Box 280, Zomba, Malawi.*
[2]*Department of Cyber Security, Kyungil University, Kyungsan, Kyungbuk 38428, Korea.*

(**Corresponding Author*)

Abstract

Secret sharing scheme is a cryptographic solution that shares a secret to n players. Any number of authorized subset of players can reconstruct the secret. After Shamir developed the first scheme, many researchers have contributed a lot especially focused on detecting and/or identifying cheaters. Linear secret sharing scheme is one of the categories of secret sharing. Liu et al.'s linear scheme detects cheating with share size $|v_i|\geq|s|/\varepsilon$ based on Shamir's scheme with two polynomials. It has many advantages including cheating detection feasibility but is not efficient in computational concern. To remedy the overhead of Liu et al.'s scheme we propose a new linear (t, n) secret sharing scheme based on single polynomial, denoted by NSSP, which is also based on Shamir's scheme. NSSP reduces the computation overhead by half as compared to Liu et al.'s scheme while the other required properties of secret sharing are maintained. Furthermore, in NSSP share size almost reaches the theoretic lower bound.

Keywords: Linear, Secret Sharing, Cheating Detection, Polynomial.

INTRODUCTION

Secret sharing scheme divides a secret into n shares, and distributes to users - one to each user and the shares do not reveal any information about the secret. The secret can be reconstructed easily if any authorized number of users pools their shares together, but an unauthorized subset of users has no information about the secret. Motivation of secret sharing was to safeguard cryptographic keys and secure storage of information [1-3]. A key/information can be securely stored in multiple servers as shares and be recovered when it is needed. It is also used in military, i.e. in launching a missile. A missile launch code is divided into shares and is distributed to colonels. The authorized subset of the colonels can launch the missile by entering their share. Secret sharing is also used in banking system where a customer's information is divided into shares. The customer keeps his/her share and the others are kept in the bank's database. The customer has to present the share during every transaction which is combined with the bank's shares for authentication. Secret sharing is also applied in multi-party computation and many other applications [4-7].

In 1979, Shamir introduced a (t, n) threshold scheme to provide a secure way of sharing a secret [8]. It starts with a secret information s, which is divided into pieces of information called shares v_i, and distributes to n individual players U_i. Any authorized subset of players is able to reconstruct the secret using Lagrange interpolation. The scheme is perfect - s can be recovered by any t; $(0 < t \leq n)$ players but not less than t and is ideal - size of the share $|v_i|$ is equal to the size $|s|$ of secret.

In 1989, Tompa et al. attacked Shamir's scheme using cheating concept and proposed a scheme which solves the problem [9]. Cheating players present fake shares during reconstruction so that honest players get an invalid secret, while they exclusively obtain a valid secret. Even if there is only one cheater, Shamir's scheme cannot resist this attack. The scheme proposed by Tompa et al. is able to detect this using redundant information so that an honest player is safeguarded from constructing and believing an invalid secret. Though the scheme is based on Shamir's scheme, the redundant information makes it not to be linear and increases share size. Other schemes in [10-13] also solve the problem of cheating detection which helps to prevent cheating during secret reconstruction. Each scheme uses its own method of cheating detection as a way of protecting honest players. However, the share size to achieve this is longer than secret size. This makes these schemes to be inefficient. The other categories of cheating prevention are under identification of cheater [14-17] and robust secret sharing [18-21]. This paper will focus on cheating detection with less computational overhead. The problem of reducing share size in secret sharing with cheating detection has been studied in [12] who derived the lower bound to be $|v_i|\geq(|s|-1)/\varepsilon+1$, which is better bound as compared to [11] who derived the lower bound to be $|v_i|\geq|s|/\varepsilon$. As some schemes are able to achieve the above stated lower bound, most of them are not linear [9-11, 22].

Linear secret sharing schemes have been studied especially by [7, 11, 19, 23-25] because of their application in multi-party computation and function sharing. The schemes are able to detect cheating behavior of malicious players during reconstruction of the secret hence an honest player cannot be fooled. Schemes in [7, 19] could be applied if system needs to share more than one secrets (Multi-secrets). Scheme in [11] depends its security on the universal hash function which means that it is not unconditionally secure, where scheme in [24] has been proved to be easily broken by a simple attack as pointed out by [25]. Liu et al.'s scheme in [23] used two polynomials to detect cheating during secret reconstruction and reduce the share size given to a player to $|v_i|\geq|s|/\varepsilon$, though its share size is longer than [12]. Use of two polynomials increases number of computations the scheme undergoes. As a result there is an increased computation overhead.

The purpose of this paper is to devise a new secret sharing scheme, named NSSP, by adopting the required features but removing the problems from the previous schemes. We set our goals to propose a new linear secret sharing scheme to satisfy the following required aspects drawn from various previous schemes

- To reduce number of polynomials used in the scheme. This will reduce number of computations as compared to other schemes
- To reduce share size given to each player so that it almost becomes equal to secret size. This will reduce communication overhead
- To provide cheating detection function so that any malicious behavior is detected. This helps to prevent any successful cheating among players
- To satisfy the requirements of a secret sharing including recoverability and privacy. This enhances the strength of the scheme.

RELATED WORKS

This section reviews Shamir's scheme [8] and Liu et al.'s scheme [23] to provide background knowledge of our research.

Shamir's Scheme

Shamir's (t, n) threshold scheme is based on polynomial interpolation. Given any t points, (x_1, y_1), …, (x_t, y_t), it is possible to evaluate a unique polynomial $f(x)$ with degree t - 1, thus any t players can reveal the secret and not t - 1 or less. The scheme has two algorithms; share generation algorithm and secret reconstruction algorithm.

Share generation: A dealer D shares a secret s to n players using a random polynomial of degree t - 1 based on equation 1.

$$f(x) = s + \sum_{i=1}^{t-1} a_i x^i \tag{1}$$

Each player receives a share $v_i = f(i)$.

Secret reconstruction: Any t, ($0 < t \leq n$) players pool their shares together and use Lagrange interpolation to evaluate the polynomial $f'(x)$, hence

$$f'(x) = \sum_{i=1}^{t} y_i \prod_{j=1, j\neq i}^{t} \frac{x-x_j}{x_i-x_j} \tag{2}$$

$$\Rightarrow f'(0) = \sum_{i=1}^{k} y_i \prod_{j=1, j\neq i}^{t} \frac{x_i}{x_i-x_j} = a_0,$$

where $f'(0)$ is the secret and the polynomial in equation 2 is exactly the same as equation 1, which D used to share the secret.

We note that in this scheme the following properties are fulfilled:

- Share size is exactly equal to secret size
- If a new player joins or leaves, it is easy to add shares or delete them without affecting the other shares
- It is easy to change the shares of the same secret just by changing the polynomial without breaching any security

However, the scheme's weakness is failure to detect any cheating behavior during reconstruction of the secret as pointed out by [9]. Even one cheater can fool all the other players without being detected.

Liu et al.'s Scheme

Liu et al.'s scheme in [23] is a linear threshold secret sharing scheme which is just a combination of two Shamir's schemes. The share size of this scheme almost reaches the theoretic lower bound. Only one player can detect cheating from t - 1 cheaters which achieves stronger detection.

Share Generation: The dealer D shares a secret s to n players using a polynomial in equation 1 and b_0 using another random polynomial in equation 3

$$g(x) = b_0 + b_1 x + \ldots + b_{t-1} x^{t-1}, \tag{3}$$

such that a random value $r \in Z_p$ and that the following are satisfied; $ra_0 + b_0 = 0$ and $ra_1 + b_1 = 0$. Then D computes $v_i = \{m_i, d_i\}$, $i = 1, 2, \ldots, n$ where $m_i = f(i)$ and $d_i = g(i)$, and distributes v_i to U_i privately.

Secret Reconstruction: Any t shares are required to reconstruct the two polynomials in equations 4 and 5 using Lagrange interpolation.

$$f'(x) = a_0' + a_1'x + \ldots + a_{t-1}'x^{t-1} \tag{4}$$

$$g'(x) = b_0' + b_1'x + \ldots + b_{t-1}'x^{t-1} \tag{5}$$

If there exists a common number $r' \in Z_p$ which satisfies $r'a_0 + b_0 = 0$ and $r'a_1 + b_1 = 0$ then $s = f'(0)$ is valid. Otherwise $f'(0)$ is an invalid secret. The use of two polynomials in the scheme doubles the computation overhead as compared to Shamir's scheme.

NEW SCHEME-NSSP

In this section, we present a new scheme with single polynomial, named as NSSP which detects any cheating behavior by dishonest players during reconstruction of the secret. NSSP is based on Shamir's scheme, where share generation and secret reconstruction are done the same way as Shamir's scheme. However, in NSSP, there is a small anomaly from Shamir's scheme which is the restriction of the first three coefficients. These coefficients are very important for cheating detection and reducing number of polynomials. Therefore, these features make NSSP to be an important scheme. To devise a secure and efficient NSSP, we set the required properties in any secret sharing scheme and introduce the idea of hiding a secret. These are discussed in subsections 3.1 and 3.2.

Fundamental Properties

Any secret sharing scheme must be secure from any adversary by denying them the opportunity to know the secret when the required number of players is not reached. At the same time, the secret should be able to be reconstructed after sharing. Two fundamental properties of designing NSSP are

- Privacy: Unauthorized subsets of participants should be prevented from learning the secret
- Recoverability: Authorized subsets of participants should be able to recover the secret by pooling their shares

where any scheme that achieves privacy and recoverability is called a perfect secret sharing scheme [28]. Tompa et al.'s attack caused most secret sharing schemes to include another property called cheater prevention which has four categories; detection, identification, robustness and verifiability. There are two assumptions that are made under cheating detection, these are Ogata, Kurosawa and Stinson (OKS) assumption and Carpentieri, De Santis and Vaccaro (CDV) assumption [11-12]. In OKS assumption, no t - 1 malicious players are aware of the secret or know part of the secret while in CDV assumption, t - 1 cheating players already know the secret but want to prevent an honest player from knowing the secret. The idea in secret sharing is that no player or any other adversary should know the secret before reconstruction takes place. Therefore CDV assumption is not proper to use in constructing a secret sharing scheme.

Thereby, the purpose of this paper is to devise a perfect new secret sharing scheme under the OKS assumption.

Hiding of Secret

This subsection gives basic idea how NSSP hide secret information. A random element $r > 1$, chosen from the finite field F_p, is multiplied by the secret s and the result is also an element of the same field. Lemma 1 and propositions 1 and 2 give the basic idea on how a hidden element can be revealed using elements of the same field.

Lemma 1: If $a \in Z_p$ possesses a multiplicative inverse, then this inverse is unique [3].

A multiplicative inverse of an element a is denote as a^{-1}.

Proposition 1: [28] Let p be prime. Then every non zero element $a \in Z_p$ has a multiplicative inverse, that is there is a number b satisfying

$$ab \equiv 1 \bmod p.$$

Proposition 2: Let p be a prime and $a, b, c \in Z_p$. Then $a \cdot b \cdot c = a$ if and only if $b = c^{-1}$.

Proof : If b is the multiplicative inverse of c, then

$$\begin{aligned} a \cdot b \cdot c &= a \cdot c^{-1} \cdot c \\ &= a \cdot 1 \\ &= a. \end{aligned}$$

Let $a \cdot b \cdot c = a$. Then by proposition 1, a has a multiplicative inverse in Z_p denoted as a^{-1} hence

$$a \cdot a^{-1} \cdot b \cdot c = a \cdot a^{-1},$$

which implies that

$$b \cdot c = 1.$$

By proposition 1, we conclude that $b = c^{-1}$.□

Using the principle of [29] to hide any other information, s can be hidden as equation 6

$$F = s \cdot r \tag{6}$$

where $F \in Z_p$ is different from r and s. Therefore it is impossible to know them just by knowing F.

When revealing the secret, a multiplicative inverse of r is computed, which is also an element of the field. By proposition 1 the multiplicative inverse, $y = r^{-1}$ exists. Revealing the secret is done simply by finding the product of F by y which gives the result as a secret as proved in proposition 2.

$$\begin{aligned} F \cdot y &= s \cdot r \cdot y \\ &= s \cdot r \cdot r^{-1} \\ &= s \end{aligned}$$

The value s is obtained back from F.

This adoption is to hide the information from cheaters so that the case of CDV assumption is avoided. For this to be achieved, elements s and r should be not known to any player. Otherwise, the purpose will be defeated. Thereby, the purpose of this paper is to devise a perfect new secret sharing scheme under the OKS assumption.

New Scheme

NSSP has two algorithms, share generation algorithm and secret reconstruction algorithm. Share generation is done by a trusted dealer D who starts by choosing a random element r in a finite field, F_p where p is prime which is multiplied by a secret s to obtain another field element F as described in subsection

The aim is to hide the secret. D also computes the multiplicative inverse of $r = y$ which is used for cheating detection. The elements s, y and F are used to be the coefficients of a random polynomial used to generate shares where $a_0 = s$ is the coefficient of x^0, $a_1 = F$ is the coefficient of x and $a_2 = y$ is the coefficient of x^2. D uses the polynomial of degree t - 1 to divide the secret into n shares and distributes them to n players using Shamir's scheme. Algorithm 1 describes in detail how share generation is done.

To reconstruct the secret, any t players are required to submit their shares and identities $(i, f(i))$ to a trusted third party called a combiner C who uses Lagrange interpolation to compute a polynomial of degree t - 1. C does not reveal the shares once collected from the players, but reveals the secret once it is proved valid. To prove the validity of the reconstructed secret, *C* multiplies the coefficient of x^2 by the coefficient of x to obtain the coefficient of x^0. Algorithm 2 shows how secret reconstruction is done by *C*.

```
Algorithm 1: Generation of Secret Share
Input: Secret $s$.
Output: Secret shares $v_i$ where $i = 1, 2, \ldots, n$.
Initialization:
  -  $D$ generates a large prime $p$ and a threshold $t$.
  -  $D$ chooses a random integer $r \in Z_p$.
  -  $D$ computes $Y = r^{-1}$ and $F = s \cdot r$.
Share Generation:
  -  $D$ chooses a random polynomial $f(x) = a_0 + a_1x + \ldots + a_{t-1}x^{t-1}$.
  -  $D$ computes $v_i = f(i)$ and distributes $v_i$ to $U_i$ secretly.
```

```
Algorithm 2: Reconstruction of Secret
Input: Any list of $t$ shares.
Output: Secret $s' = s$ if no cheating occurred or $\perp$ if there is cheating.
  -  $C$ reconstructs $f'(x)$ from $(i, v_i)$ using Lagrange interpolation
       $f'(x) = \sum_{i=1}^{t} s_i \prod_{j=1,j\neq i}^{t} \frac{x_i - x}{x_i - x_j}$
          $= a_0' + a_1'x + \ldots + a_{t-1}'x^{t-1}$.
  -  $C$ computes $a_1' \cdot a_2' = a_0'$.
  -  $C$ outputs $f'(0) = s' = s$ if the computation holds or $\perp$ if it does not hold.
```

Cheating Detection

In this subsection, we show how cheating detection works using the coefficients of the reconstructed polynomial $f'(x)$. The coefficient of x^0 is the secret if there is no cheating. Since the coefficient of x^1 is $F = s \cdot r$, the secret is contained in it together with the random element r, and the coefficient of x^2 is the multiplicative inverse of an element r which is y. Using the knowledge of subsection 3.2, s is obtained from F as in equation 7

$$\begin{aligned} F \cdot y &= s \cdot r \cdot y \\ &= s \cdot r \cdot r^{-1} \\ &= s \end{aligned} \tag{7}$$

, which will only work if all shares are valid. Once fake shares are submitted to *C*, the interpolated polynomial $f'(x)$ of degree t - 1 which is unique will be different from $f(x)$ i.e. $f'(x) \neq f(x)$. Since *C* does not know the polynomial $f(x)$, then he/she computes $a_1' \cdot a_2' = a_0'$. Because the points will not be the same, the coefficients of the polynomial $f'(x)$ will not be the same to $f(x)$'s coefficients. Hence, equation 8 applies

$$a_1' \cdot a_2' \neq a_0'. \tag{8}$$

Therefore, *C* could detect that the secret is invalid. It means that even if cheating is occurred in NSSP, it is not successful.

ANALYSIS

This section provides the proof for privacy and recoverability, of NSSP and discusses the computational overhead of NSSP by comparing it to Shamir's and Liu et al.'s schemes.

Security Analysis

In this subsection we will prove the recoverability and privacy of NSSP, that is only t shares could get the secret but t - 1 or less shares do not reveal any information on the secret. We also show how cheaters can be detected in NSSP.

Lemma 2: Any secret share given to a participant does not reveal any information about the secret s in NSSP.

Proof : In share generation, NSSP uses Shamir's scheme of sharing the secret which uses the polynomial $f(x)$ of degree t - 1 to share the secret where a_0 is the secret. Since each share in Shamir's scheme is a result of substituting i in the polynomial to get $f(i)$ which does not reveal about a_0. Then security of NSSP is the same as Shamir's scheme because it just adopts the method. No information can be revealed by individual shares on their own in NSSP.

Proposition 3: Given any t shares, the secret can be reconstructed and any t - 1 shares or less do not reveal any information about the secret in NSSP.

Proof : In share reconstruction, NSSP uses Shamir's scheme of recovering the secret. To reconstruct the secret, Lagrange interpolation is used to come up with the secret. By this method, using any t points, a unique polynomial of degree t - 1 is obtained. Thus

$$f'(x) = \sum_{i=1}^{t} s_i \prod_{i=1,j\neq i}^{t} \frac{x_i - x}{x_i - x_j}.$$

The term a_0 is the secret. Since the polynomial is unique, the two polynomials $f(x)$ and $f'(x)$ are equal. The terms a_0 from $f(x)$ and a_0' from $f'(x)$ are also equal. This proves the correctness of the NSSP.

Now we need to show that t - 1 shares do not reveal any information about the secret. If we assume t - 1 participants collude to recover the secret, they have to solve a system of t - 1 equations with t unknowns.

$$\begin{pmatrix} 1 & ID_1 & \ldots & ID_1^{t-1} \\ 1 & ID_2 & \cdots & ID_2^{t-1} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & ID_{t-1} & \ldots & ID_{t-1}^{t-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{t-1} \end{pmatrix} = \begin{pmatrix} \mathrm{f}(ID_1) \\ \mathrm{f}(ID_2) \\ \vdots \\ \mathrm{f}(ID_{t-1}) \end{pmatrix}$$

It is impossible to solve this system of equations unless a t-th term is guessed. Thus we need at least t points to interpolate the polynomial which might be not correct since the shares are secretly transmitted to participants. By lemma 2 the t - 1 participants do not reveal the information about the secret and by subsection 3.4, cheating will be detected in NSSP. This proves our second claim.□

Computational Overhead Analysis

In this subsection, we provide the required computational overhead for NSSP and later compare these computations with those in Shamir's scheme [3] and Liu et al.'s scheme [23]. The computation analysis of NSSP is only focused on share generation and secret reconstruction because the computation in initialization phase is done once. For simplicity, we will use Mul, Add and Inv for multiplication, addition and inverse modulo, respectively.

During share generation, NSSP needs to compute n shares from a polynomial $f(x)$ of degree t - 1. To generate n shares, the following number of operations occurs

- $n(t - 1)$ Mul
- nt Add.

During secret reconstruction, Lagrange interpolation uses t points which are the shares and the identities of the player i.e. (i, v_i) to obtain a polynomial $f'(x)$. To reconstruct the polynomial the following computations are done

- $t^3 + t + 1$ Mul
- t Inv
- t Add.

Table 1 shows the comparison of the computational overhead of related schemes among Shamir's, Liu et al.'s and NSSP.

Table 1. Computational Overhead in Secret Sharing Schemes

Share Generation				
Scheme	**Mul**	**Add**	**Inv**	**Total Operations**
Shamir's	$n(t$-1)	nt	-	$n(t$-1) Mul + nt Add
Liu et al.'s	2$n(t$-1)	2nt	-	2$n(t$-1) Mul + 2nt Add
NSSP	$n(t$-1)	nt	-	$n(t$-1) Mul + nt Add
Share Reconstruction				
Shamir's	t^3+t+1	t	t	t^3+t+1 Mul + t Add + t Inv
Liu et al.'s	$2(t^3+t+1)$	2t	2t	$2(t^3+t+1)$ Mul + 2t Add +2t Inv
NSSP	t^3+t+1	t	t	t^3+t+1 Mul + t Add + t Inv

The analysis shows that NSSP has the same computation overhead to Shamir's scheme and half from Liu et al.'s scheme. Liu et al.'s scheme and NSSP have cheating detection capability. This makes NSSP more secure and efficient than Shamir's and Liu et al.'s schemes.

CONCLUSION

Secret sharing scheme should satisfy privacy and recoverability and provide cheating prevention which has been discussed in this paper. This paper has proposed a new secret sharing scheme named NSSP which also satisfies these properties where t shares can be used in reconstructing the secret while any t - 1 should not be able to know the secret. NSSP has the ability to use the coefficients of the polynomial to achieve cheating detection which is done by the combiner. The coefficients of the x^0, x and x^2 of the polynomial reconstructed by Lagrange interpolation are used for cheating detection. Cheating detection occurs even if there are t - 1 cheating players. Use of one polynomial to share the secret in NSSP helps it to reduce computation overhead by half as compared to Liu et al.'s scheme. Share size of NSSP almost reaches the theoretic lower bound which increases the efficiency of the scheme.

ACKNOWLEDGEMENTS

Corresponding author is Hyunsung Kim. This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2017R1D1A1B04032598).

REFERENCES

[1] A. Beimel, “Secret Sharing Schemes: A Survey,” Lecture Notes in Computer Science, vol. 6639, **(2011)**, pp. 11-46.

[2] A. J. Meneze, P. C. van Oorschot, S.A. Vanston, Handbook of Applied Cryptography, CRC press, **(1997)**.

[3] A. Slinko, Algebra for Applications: Cryptography, Secret Sharing, Error-Correcting, Fingerprinting, Compression, Springer, **(2015)**.

[4] Chandrasekhara, R. Roopalakshmi, “A Novel Approach of Secure Banking Application Using Visual Cryptography against FakeWebsite Authenticity Theft,” International Journal of Engineering, Research and Technology, vol. 2, no. 4, **(2013)**, pp. 2208-2211.

[5] B. Srikanth, G. Padmaja, S. Khasim, P. V. S. Likshmi, A. Haritha, “Secure Bank Authentication Using Image Processing and Visual Cryptography,” International Journal of Computer Science and Information Technologies, vol. 5, no. 2, **(2014)**, pp. 2432-2437.

[6] M. Liu, L. Xiao, Z. Zhang, “Linear Multi-secret Sharing Scheme Based on Multi-party Computation,” Finite Fields and Their Applications, vol. 12, **(2006)**, pp. 704-713.

[7] R. Cramer, I. Damgard, U. Maurer, “General Secure Multi-party from any Linear Secret Sharing Scheme,” Lecture Notes in Computer Science, vol. 1807, **(2000)**, pp. 316- 334.

[8] A. Shamir, “How to Share a Secret,” Communication of the ACM, vol. 22, no. 11, **(1979)**, pp. 612-613.

[9] M. Tompa, H. Woll, “How to share a secret with cheaters,” Journal of Cryptology, vol. 1, **(1989)**, pp. 133-138.

[10] P. Y. Lin, Y. H. Chen, M. C. Hsu, F. M. Juang, "Secret Sharing Mechanism with Cheater Detection," Proc. of Signal and Information Processing Association Annual Summit and Conference, **(2013)**, pp. 1-4.

[11] M. Carpentieri, A. De Santis, U. Vaccaro, "Size of Share and Probability of Cheating in Threshold Schemes," Lecture Notes in Computer Science, vol. 765, **(1993)**, pp. 118-125.

[12] W. Ogata, K. Kurosawa, "Optimum Secret Sharing Scheme Secure against Cheating," Lecture Notes in Computer Science, vol. 1070, **(1996)**, pp. 200-211.

[13] J. Pieprzyk, X. Zhang, "Cheating Prevention in Linear Secret Sharing," Information Security and Privacy, vol. 2384, **(2002)**, pp. 121-135.

[14] D. Pasaila, V. Alexa, S. Iftene, "Cheating Detection and Cheater Identication in CRT-Based Secret Sharing Schemes," International Journal of Computing, vol. 9, no. 2, **(2010)**, pp. 107-117.

[15] I. Lin, C. Chang, "A (t, n) Threshold Secret Sharing System with Efficient Identification of Cheaters," Computing and Informatics, vol. 24, **(2005)**, pp. 529-541.

[16] R. Kalombe, M. Kamble, "Cheater detection and identification based on Shamir Scheme," International Research Journal of Computer Science Engineering and Application, vol. 2, no. 2, **(2013)**, pp. 255-259.

[17] S. Obana, "Almost Optimum t-Cheater identifiable Secret Sharing Scheme," Lecture Notes in Computer Science, vol. 6632, **(2011)**, pp. 284-302.

[18] M. P. Jhanwar, R. Safavi-Naini, "On the share efficiency of Robust Secret Sharing and Secret Sharing with Cheater Detection," Lecture Notes in Computer Science, vol. 8250, **(2013)**, pp. 179-194.

[19] A. Bishop, V Pastro, "Robust Secret Sharing Schemes Against Local Adversaries," Lecture Notes in Computer Science, vol. 9615, **(2016)**, pp. 327-356.

[20] M. P. Jhanwar, R. Safavi-Naina, "Unconditionally-Secure Robust Secret Sharing with Minimum Share Size," Lecture Notes in Computer Science, vol. 7859, **(2013)**, pp. 96-110.

[21] K. Kaya, A. A. Selcuk, "Robust Threshold Scheme Based on the Chinese Remainder Theorem," Lecture Notes in Computer Science, vol. 5023, **(2008)**, pp. 94-108.

[22] R. Gennaro, S.A. Jarecki, H. Krawczyk, T. Rabin, "Robust Threshold DSS Signatures," Lecture Notes in Computer Science, vol. 164, **(1996)**, pp. 54-84.

[23] Y. Liu, Z. Wang, W. Yan, "Linear (k,n) Secret Sharing Scheme with Cheating Detection," Proc. of IEEE International Conference on Computing and communications; Dependable Autonomic and Secure Computing; Pervasive intelligence and computing, **(2015)**, pp. 1942-1947.

[24] L. Harn, C. Lin, "Detection and Identification of Cheaters in (t, n) Secret Sharing Scheme," Designs, Codes and Cryptography, vol. 52, **(2009)**, pp. 15-24.

[25] H. Ghodosi, "Comments on Harn-Lin's Cheating Detection Scheme," Designs, Codes and Cryptography, vol. 60, **(2011)**, pp. 63-66.

[26] E. F. Brickell, "Some Ideal Secret Sharing Schemes," Lecture Notes in Computer Science, vol. 434, **(2001)**, pp. 468-475

[27] K. M. Martin, "Challenging the Adversary Model in Secret Sharing Schemes, Coding and Cryptography II," Proc. of Royal Flemish Academy of Belgium for Science and Art, **(2008)**, pp. 45-63.

[28] J. Ho_stein, J. Pipher, J. H. Silverman, An Introduction to Mathematical Cryptography, Springer, **(2008)**.

[29] T. El Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," Lecture Notes in Computer Science, vol. 196, **(1985)**, pp. 10-18.