

VERIFIABLE SECRET SHARING SCHEME BASED ON INTEGER REPRESENTATION

Qassim Al Mahmoud¹

ABSTRACT

In Shamir's scheme that the security based on the numbers of the field of a prime number P which the coefficients' polynomial reduced to modulo P (takes a value from some field $\mathbb{Z}_P$, where P is a large prime number). Thus, the adversary must know only the free coefficient of the polynomial in order to break the scheme. Our scheme which based on representation integer using the so-called g -adic expansion we can see any integer a can build such polynomial so that the polynomial has degree of k , k being the length digits for that integer^a. Where the coefficients of that polynomial are taken from the set of $\{0, 1, \dots, g-1\}^k$, we will introduce a small participation improving Shamir's scheme to share the secret S that will be seen constructed from the whole of the coefficients in the polynomial used, which is based on the representation integer. Then apply Pederson's VSS scheme in order to improve our scheme to be Verifiable Secret Sharing Scheme Based on Integer Representation.

Keywords: Shamir scheme, information dispersal, integer representation, verifiable secret sharing (VSS), g -adic

1. INTRODUCTION

The secret sharing is the main topic in the information security field. In the early evolution of cryptography the key exchange between users was one of those for which scientists tried to find an ideal solution. Day after day they have seen the need to save the information from the people themselves, who are responsible for saving the key. Security of crucial information, such as credit evidences and confidential documents, has become one of the most important issues nowadays. Usually, we encrypt these pieces of critical information to protect data. But how to protect the encryption key?.

Shamir mentioned the question in 1979 [1]. In practical, to improve encryptions can not solve this problem. Traditional methods for encryptions are ill-suited for simultaneously achieving high levels of confidentiality and reliability as they usually keep keys in a single and well-guarded location where single-point-failure exists. Moreover, it is also very significant that keys not be lost or exposed.

Secret sharing schemes have been independently introduced by Shamir [1] Blakley [2] and, as a solution for safeguarding cryptographic keys. Secret sharing schemes can be used in any way in which the access to an important resource has to be restricted.

¹ Faculty of Mathematics and Computer Science, the University of Bucharest, Romania

Basic secret sharing schemes assume that the dealer who divides the secret and distributes shares to participants is a mutually trusted party. In 1985, Chor et al. [3] extended the original secret sharing and presented a notion of verifiable secret sharing (VSS). The property of verifiability enables participants to verify that their shares are consistent according to the security property. In Feldman's VSS scheme [4] is that the committed values are publicly known and the privacy of secret S depends on the difficulty of solving the discrete logarithm problem. In other words, Feldman's scheme is computationally secure.

Later, Pedersen [5] (Pedersen, 1992) used a commitment scheme to remove the assumption in Feldman's VSS scheme to propose a VSS scheme which is information-theoretically secure. However, in Pedersen's VSS scheme the dealer can succeed in distributing incorrect shares if the dealer can solve the discrete logarithm problem (see previous study).

Consideration to the concept of secret must be taken into account by the group of people selected to be the group authorized to build the concept of secret sharing, dividing this group into subsets where each subset can retrieve private confidence. In fact this is the definition of the access structure. We have previously mentioned that there is a subset of group that is authorized to possess the key for retrieving the information which is stored, when we need it, and that is group is called the Access Structure. *Let $X = \{1, 2, \dots, n\}$ be the set of users, the access structure $\Gamma \subseteq P(X)$ we give bounds on the amount of information (shares) for each participant. Then we apply this to construct computational schemes for general access structures.*

Let us consider a set of groups $\Gamma \subseteq P(X)$. The (authorized) access structure of a secret sharing scheme is the set of all groups which are designed to reconstruct the secret. The elements of the access structure A will be referred to as the authorized groups/ sets and the rest are called unauthorized groups/ sets. In other words, the unauthorized access structure $\bar{\Gamma}$ is well specified by the set of the maximal unauthorized groups.

In secret sharing schemes, the number of the participants in the reconstruction phase was important for recovering the secret. Such schemes have been referred to as "threshold secret sharing schemes."

In the Shamir threshold secret sharing scheme, the secret is chosen as the free coefficient of a polynomial, so that only a fixed number (k) of people (or more) may reconstruct the polynomial, and then restore that secret. Less than the satisfying number of people ($k-1$) should not be able to know anything about that polynomial then secret, this being called the threshold secret sharing scheme.

In way that secret S is generated from some fragments of pieces in that scheme, the secret S refer to information and the shares refer to fragments such that the scheme $(S, (F_1, F_2, \dots, F_n))$ the problem is finding the secret S given the fragments $(S, (F_1, F_2, \dots, F_n)) \cdot (k, n)$ - Threshold secret sharing scheme is based on information dispersal. Threshold information dispersal schemes have been introduced by Rabin [6]. For

any authorized subsets A of set of participants so that $|A| = k$ the problem to find a secret S with k of fragment F . Krawczyk [7] has proposed combining perfect threshold secret sharing schemes with encryption and information dispersal in order to decrease the size of shares with the cost of decreasing the level of security.

In [8], the representation integer using the so-called g –adic expansion we can see any integer α can build such polynomial that polynomial has degree of k , so that k is the length digits for that integer α . Where the coefficients of that polynomial are taken from the set of $\{0, 1, \dots, g - 1\}^k$, we will improve the Shamir's scheme to share the secrets S that will be seen constructed from the whole of the coefficients in the polynomial used, which is based on the representation integer, then apply Pederson VSS on our scheme in order to be construct Verifiable secret sharing scheme based on g –adic integer representation.

This paper is organized as follows: within the rest of this section, we will give the definition of the g –adic representation integer; section 2 will be the previous study, *which is divided into three studies: the first study will be explained on a threshold Shamir secret sharing scheme. The second study will be a (k, n) - Threshold information and an important example (Krawczyk scheme). And the third study will be about Pedersen VSS*; in section 3, we will give an explanation for our scheme, which is the Verifiable secret sharing scheme based on g –adic integer representation with a small artificially example; in Section 4, we will discuss about the security of our scheme; and finally the conclusion will be included in section 6.

Integer Representation [8]:

Our scheme will be seen in section 3, and it is based on integer representation. Within theorem 1 and definition 1, so that it is very important to view and understand this section well.

In [8], the integer can be represented using the so-called g –adic expansion, for an integer $g > 1$ and a positive real number α , denote by $\log_g \alpha$ the logarithm for base g of α . For a set M , let M^k be the set of all sequences of length k with entries from M

Example 1: We have $\log_2 8 = 3$ because $2^3 = 8$. Also $\log_8 8 = 1$ because $8^1 = 8$.

Example 2: The sequence $(0, 1, 1, 1, 0)$ is an element of $\{1, 0\}^5$. Also $\{1, 2\}^2 = \{(1, 1), (1, 2), (2, 1), (2, 2)\}$.

Theorem 1[8]: Let g be an integer, $g \geq 2$. For each positive integer α , there is a uniquely determined positive integer k and a uniquely determined sequence $(a_1, a_2, \dots, a_k) \in \{0, 1, \dots, g - 1\}$ with $a_1 \neq 0$ and

$$\alpha = \sum_{i=1}^k a_i g^{k-i} \quad (1)$$

In addition, $k = \lfloor \log_g \alpha \rfloor + 1$ and $a_i = \left\lfloor (\alpha - \sum_{i=1}^k a_i g^{k-i} / g^{k-i}) \right\rfloor$, $1 \leq i \leq k$.

Definition 1[8]: The sequence $(a_1, a_2, \dots, a_n)$ from theorem is called g -adic expansion of a . Its elements are called digits. Its length is $k = \lfloor \log_g a \rfloor + 1$. If $g = 2$ the sequence is called binary expansion of a , If $g = 16$ then the sequence is called the hexadecimal expansion of a . For more details the reader has to read in [8], [9].

In definition 1 we had seen that any integer can be a representation as k -tuple of elements from the set $M = \{0, 1, \dots, g-1\}^k$ as in eq. (1). In fact this is definition of integer representation will be useful in section 3.

2. PREVIOUS STUDIES

In this section we have to mention, in general, the important threshold schemes in secret sharing: the first part in this section talks about Shamir's secret sharing scheme, the second part will be about information dispersal and an important example (Krawczyk schemes), And the third study will be about Pedersen VSS.

Threshold Shamir Secret Sharing Scheme

In 1979, Shamir [1] has introduced the threshold secret sharing as a solution for safeguarding cryptographic keys. His scheme has generated the thinking of how to share a secret with multi participants, and it has been used until now as the root for wide research papers in computer security. Thus, we should give an overview of his important scheme.

In Shamir's (k, n) -scheme based on Lagrange interpolating polynomial, there are n shareholders, $P = \{P_1, P_2, \dots, P_n\}$ and a dealer D . The scheme consists of two algorithms:

1-Generation Shares Algorithm: Dealer D does the following:

- Picks a polynomial $f(x)$ of degree $(k-1)$ randomly $f(x) = a_0 + a_1x + \dots + a_{k-1}x^{k-1}$, in which the secret $S = a_0$ and all coefficients $a_0, a_1, \dots, a_{k-1}$ are in a finite field $F_p = GF(P)$
- Computes: $s_1 = f(1), s_2 = f(2), \dots, s_n = f(n)$
- Outputs a list of n shares $(s_1, s_2, \dots, s_n)$, and distributes each share to corresponding participants privately.

2-Secret Reconstruction Algorithm:

With any t shares, $(s_{i1}, s_{i2}, \dots, s_{it})$, we can reconstruct the secret S as t participants work together and then, pooling their shares, then apply Lagrange's interpolation to find the coefficients in polynomial used $f(x)$ as follow:

$$f(x) = \sum_{i=0}^k L_i(x) f(x_i) \quad (2)$$

Where k in $f(x)$ is order polynomial that approximates the function and the points as,

$$L_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^k \frac{x - x_j}{x_i - x_j} \quad (3)$$

Then the secret is:

$$S = f(0) = \sum_{i \in A} S_i \left(\prod_{j \in A / \{i\}} \frac{x_j}{x_j - x_i} \right)$$

We note that the above scheme satisfies the basic requirements of the secret sharing scheme as follows:

- With knowledge of any k or more than k shares, it can reconstruct the secret S .
- With knowledge of any fewer than k shares, it cannot reconstruct 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 [1].

In Shamir's (t, n) SS, a dealer is a third trusted party who generates and distributes shares to n participants by using such polynomial, k participants and (more) can reconstruct the secret S , and less than k know nothing about S .

Threshold Information Dispersal

In Shamir secret sharing in section (2.1) above has restriction on unauthorized groups, which is difference in information dispersal, in addition, the secret S has been divided into n shares with the same size of secret S , in other hand, if the secret has size m , each shares have the size m too, Thus his method leads to n -fold increase in total storage. Threshold information dispersal schemes have been introduced by Rabin [6]. He has proposed the Information Dispersal Algorithm (IDA) to develop that breaks a file (secret S) of length $L = |S|$ into n pieces $|S_i|$, $1 \leq i \leq n$, the secret S can be reconstructed by any k shares S_i each of length $|S_i| = L/k$, so that every k pieces suffice for reconstructing S .

In Iftene [10] has been defined information dispersal scheme more clearly, so that it mentions the scheme from original paper.

Definition 2: Let n be an integer. Where $n \geq 2$, and $2 \leq k \leq n$. Informally, an $(k-1)$ -threshold information dispersal scheme is a method of generating $(S, (F_1, F_2, \dots, F_n))$ such that for any set A with $|A| = k$, the problem of finding the element S . Given the set $\{I_i / i \in A\}$, is "easy". Such that the secret S refers to information and the shares $(F_1, F_2, \dots, F_n)$ refer to fragments.

Krawczyk [7] has presented a threshold information dispersal scheme very similar to Shamir's threshold secret sharing scheme:

- - The information S is chosen as the vector of the coefficients of a random polynomial $f(x)$ of degree $k-1$ over some field;

- - The fragments $(F_1, F_2, \dots, F_n)$ are chosen as follows: $F_i = f(x_i)$, $1 \leq i \leq n$, where $(x_1, x_2, \dots, x_n)$ are pairwise distinct public values.
- - Having the fragments $\{F_i / i \in A\}$, for some group A with $|A| = k$, the polynomial $f(x)$ and, thus, the information S , can be obtained using Lagrange's interpolation formula as in equations (2, 3).

The information is chosen as the whole polynomial as opposed to the perfect secret sharing scheme in which the secret is chosen only as the free coefficient. Our scheme will be seen in section 3, being based on the Krawczyk scheme, which is different in the way of how the polynomial $f(x)$ is chosen, and the coefficients are chosen based on integer representation.

Pederson's VSS Scheme

In Feldman's VSS scheme [4] is that the committed values are publicly known and the privacy of secret s depends on the difficulty of solving the discrete logarithm problem. In other words, Feldman's scheme is computationally secure. In 1992 Pedersen [5] proposed a non-interactive and information-theoretically secure VSS scheme based on Feldman's VSS scheme.

Let p and q be two large primes such that $q | (p-1)$, and $g, h \in Z_p$ are two elements of order q . There are n participants $P = \{P_1, P_2, \dots, P_n\}$ and a dealer D who will divide a secret $s \in Z_p$ we describe Pedersen's scheme below.

1-Generation Shares Algorithm: Dealer D does as follows:

- Picks a polynomial $f(x) = a_0 + a_1x + \dots + a_{t-1}x^{k-1}$ of degree at most $(k-1)$ randomly, in which the secret $S = a_0 = f(0)$ and all coefficients $a_0, \dots, a_{k-1}$ are in Z_p
- Picks $b_0, \dots, b_{k-1} \in Z_p$ at random. Let $k(x) = b_0 + b_1x + \dots + b_{t-1}x^{k-1}$.
- Computes shares (s_i, t_i) for $i = 1, \dots, n$ and each coefficient's commitment of added sum of polynomials of $f(x)$ and $k(x)$ as follows: $(s_i, t_i) = (f(i), (k(i)))$
- Computes $c_j = g^{a_j} h^{b_j} \mod p$ for $j = 0, 1, \dots, k-1$.
- Outputs a list of n shares (s_i, t_i) and distributes each share to corresponding participants P_i privately. D also broadcasts c_j

2. Share verification: each participant P_i , who has received the share (s_i, t_i) and all broadcasted information, can verify that share defines a secret by testing:

$$g^{s_i} h^{t_i} \mod p = \prod_{j=0}^{k-1} c_j^{i^j} \mod p \quad (4)$$

2 Secret Reconstruction Algorithm: It is same as Shamir's scheme.

In Pedersen's scheme, the value g^{a_0} is not made publicly known, that is, the secret S is embedded in the commitment $c_0 = g^{a_0} h^{b_0}$. Thus, no information directly about the secret S is revealed even if an attacker with unlimited computing power can solve $\log_g h$, the attacker still gets no information about the secret S .

3 Verifiable Secret Sharing Scheme Based On Integer Representation

Now, after we overviewed the mathematical integer representation we can start to view our scheme as based on polynomials, which use the integer representation to construct the coefficients of polynomial used, then apply Pederson's shares verification algorithm and then generate the shares and verification shares, with the difference that the secret will be the value $f(g)$, where g is the base in g -adic, the shares verification will be as in Pederson's VSS. The reconstruction algorithm is any subset authorized A so that $|A| = k$ pooling their shares and use interpolation polynomial to reconstruct the secret the polynomial $f(x)$, then find $f(g) = S$ where S the secret.

Our scheme has three algorithms:

1- Construction of the Polynomial Used, with Share Generation Algorithm

- - Choose a secret S that has the length k (number of digits) which is defined in definition 1, and g the base in g -adic, g will be public number.
- - Represent S by using the definition 1 and theorem 1, as follow:

$$S = \sum_{i=1}^k a_i g^{k-i}$$

Where the coefficients $(a_1, a_2, \dots, a_k) \in \{0, 1, \dots, g-1\}$ with $a_1 \neq 0$

- - Construct the polynomial $f(x)$ of degree $(k-1)$ from equation (1) as:
$$f(x) = a_{k-0}x^0 + a_{k-1}x^1 + \dots + a_1x^{k-1}.$$
- - Picks $b_0, \dots, b_{k-1} \in \mathbb{Z}_p$ at random. Let $k(x) = b_0 + b_1x + \dots + b_{k-1}x^{k-1}$.
- - Computes shares (s_i, t_i) and each coefficient's commitment of added sum of polynomials of $f(x)$ and $k(x)$ as follows: $(s_i, t_i) = (f(i), (k(i)))$
- - Computes $c_j = g^{a_j} h^{b_j} \mod p$ for $j = 0, 1, \dots, k-1$.
- - Outputs a list of n shares (s_i, t_i) and distributes each share to corresponding participants P_i privately. D also broadcasts c_j

2- Share verification: It is same as Pederson share verification.

3- Secret Reconstruction Algorithm

In reconstruction algorithm any subset authorized A so that $|A| = k$ pool their shares and S can be obtained using Lagrange's interpolation formula as in equations (2,3) to reconstruct the polynomial $f(x)$, then find $f(g) = S$ where S the secret.

Note that g is public number for all participants (n), when we use g large number that has large length k , the scheme close to be more secure.

We note that when the secret and the base g are small numbers, the security for our schemes non-secure, so that when we are use large numbers (not necessarily prime numbers, different in Shamir's scheme) our scheme will seem more secure than Shamir's scheme.

In the next section will be discussing about the security in our scheme and will be compared to Shamir's scheme.

Example (with artificially parameters)

1- Shares generations Algorithm.

Let us will share the integer secret $S=5923$, $n=10$ participants.

We can represent the secret in g -adic representation definition as follow:

Now let us we will use to represent the secret in hexadecimal ($g=16$) representation then we can rewrite the secret 5923 in hexadecimal as follow:

$$5923 = 1 \times 16^3 + 7 \times 16^2 + 2 \times 16^1 + 3$$

Then the secret polynomial used is:

$$f(x) = x^3 + 7x^2 + 2x^1 + 3$$

The threshold will be $k=4$ such that any 4 participants and more than 4 can reconstruct the secret and less than 4 participants cannot reconstruct it.

Now we will use this polynomial in order to generate the shares in generation algorithm as in Shamir's scheme, then the shares will be:

$$\begin{aligned} \{s_1 = f(1) = 13, s_2 = f(2) = 43, s_3 = f(3) = 99, \\ s_4 = f(4) = 187, s_5 = f(5) = 313, s_6 = f(6) = 483, \\ s_7 = f(7) = 703, s_8 = f(8) = 979, s_9 = f(9) = 1317, \\ s_{10} = f(10) = 1723\} \end{aligned}$$

2- Shares Verification: It is same as Pederson's share verification.

3 Secret Reconstruction Algorithm: In the secret reconstruction algorithm, any authorized subset of n participants with order has 4, S can be obtained using Lagrange's interpolation formula as in equations (2, 3) to reconstruct the polynomial $f(x)$, and then find $f(16) = S$.

Suppose that the participants $\{2, 3, 6, \text{ and } 7\}$ want to reconstruct the secret S , they do the following:

They use the shares

$$\{s_2 = f(2) = 43, s_3 = f(3) = 99, s_6 = f(6) = 483, s_7 = f(7) = 703\}$$

In order to reconstruct the secret polynomial used $f(x) = x^3 + 7x^2 + 2x + 3$

Then they find the secret as $S = f(16) = 5923$

3. SECURITY ANALYSIS

Shamir's scheme is provable secure, that means: in a (k, n) scheme one can prove that it makes no difference whether an attacker has $k-1$ valid shares at his disposal or none at all; as long as he has less than k shares, there is no better option than guessing to find out the secret.

We have noted in Shamir's scheme that the security based on the numbers of elements in field of a prime number P which the coefficients' polynomial reduced to modulo P (take a values from some field Z_p , where P large prime number). Thus, the adversary must know only the free coefficient of polynomial in order to break the scheme, on the other hand the probability that adversary will know the secret S is $1/P$. Shamir's scheme is a uniform distribution and his scheme is perfect [1]. Our scheme is more secure than Shamir scheme, which means that the adversary must know all coefficients of polynomial which are used. In addition, he must exactly know the values of coefficients $(a_1, a_2, \dots, a_k)$, with the order of $(a_1, a_2, \dots, a_k)$ exactly. The probability for adversary to know the secret in our scheme is $1/(g^k)$. If k and g are two large numbers, our scheme seems to be secure than Shamir's scheme. This means that the probability is closer to zero. We have seen that if $k = 1$, and $g = p$ our scheme is similar to Shamir's security scheme. In addition our scheme is a uniform distribution as Shamir's scheme.

In Pedersen's scheme, the value g^{a_0} is not made publicly known, that is, the secret S is embedded in the commitment $c_0 = g^{a_0} h^{b_0}$. Thus, no information directly about the secret s is revealed even if an attacker with unlimited computing power can solve $\log_g h$, the attacker still gets no information about the secret S . In our scheme the secret S is embedded in all commitments $c_i = g^{a_i} h^{b_i}$, and then the attacker needs to compute all broadcasts information till break the scheme.

4. CONCLUSION

We introduced this scheme to be helpful in future works for anyone who is interested in this domain of information security. Our future work will study the domain which is related to the secret sharing scheme based on integer representation, such as those schemes that do not require a dealer (free dealer secret sharing scheme). In addition, we will develop our method, in order to build the verification scheme.

In this paper, we have used the Integer representation in order to construct the polynomial that assigns an integer which will be the secret (S). Our scheme, in section (3); was based on Shamir's scheme which generates the shares, then distributes the shares to the participants, and based on the Information dispersal scheme, in order to divide the secret (S) within whole terms in polynomial ($f(x)$) that is constructed by our scheme, the secret $f(g) = S$. We have seen that our scheme is more secure than Shamir's scheme when both were compared in section IV.

5. REFERENCES

1. Shamir." How to share a secret ". *Communications of the ACM*, 1979.
2. G. R. Blakley." Safeguarding cryptographic keys". In *National Computer Conference*, 1979, volume 48 of *American Federation of Information Processing Societies Proceedings*, pages, 1979.
3. B. Chor, S. Goldwasser, S. Micali, B. Awerbuch, *Verifiable secret sharing and achieving simultaneity in the presence of faults*, *Proceedings of the 26th IEEE Symposium on Foundations of Computer Science*, 21–23 October, Oregon, Portland, IEEE Computer Society, 1985, pp. 383–395.
4. Feldman, P., 1987. A practical scheme for non-interactive verifiable secret sharing. In: *Proceedings of the 28th IEEE Symposium on Foundations of Computer Science*, 27–29 October. IEEE Computer Society, Los Angeles, California, pp. 427–437.
5. Pedersen, T.P., 1992. Non-interactive and information-theoretic secure verifiable secret sharing. In: *Advances in Cryptology-CRYPTO'91*, LNCS, vol. 576. Springer- Verlag, Berlin, pp. 129–140.
6. M. O. Rabin." Efficient dispersal of information for security, load balancing, and fault tolerance". *Journal of ACM*, 36(2):335–348, 1989.
7. H. Krawczyk. , *Secret sharing made short*. In D. R. Stinson, editor, *Advances in Cryptology - CRYPTO '93*, volume 773 of *Lecture Notes in Computer Science*, pages 136–146. Springer-Verlag, 1994.
8. Johannes A. Buchmann. „*Introduction to Cryptography*". Department of Computer Science Technical University, Germany, ISBN O-387-95034-6, Springer-Verlag New Yourk, Heidelberg, pages 2-6, September, 2000.
9. Delfs, H. Knebl, H. "Introduction to Cryptography Principles and Applications". second edition, Georg-Simon-Ohm University of Applied Sciences N^urnberg Department of Computer Science, Germany, ISBN-13 978-3-540-49243-6 Springer Berlin Heidelberg New York, pages 35- 37, 2007
10. Sorin Iftene. "Secret Sharing Schemes with Applications in Security Protocols". *Sci. Ann. Cuza Univ.*17-18, 2007.