

HYBRID APPROACH FOR VERIFIABLE SECRET SHARING USING CHINESE REMAINDER THEOREM

MAJOR PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE AWARD OF DEGREE OF

Master of Technology

In

Information Systems

Submitted By:

BHARTI MANJWANI

(2K13/ISY/05)

Under the Guidance

Of

Dr. O. P. Verma

(Prof. and Head, Department of CSE)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(2013-2015)

CERTIFICATE

This is to certify that **Bharti Manjwani (2K13/ISY/05)** has carried out the major project titled “**Hybrid Approach For Verifiable Secret Sharing Using Chinese Remainder Theorem**” in partial fulfilment of the requirements for the award of Master of Technology degree in Information Systems by **Delhi Technological University**.

The major project is bonafide piece of work carried out and completed under my supervision and guidance during the academic session 2013-2015. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University/Institute for the award of any degree or diploma.

Dr. O. P. Verma

Professor and Head

Department of Computer Science and Engineering

Delhi Technological University

Delhi-110042

ACKNOWLEDGEMENT

I take the opportunity to express my sincere gratitude to my project mentor Dr. O. P. Verma, Prof. and Head of Department, Department of Computer Science and Engineering, Delhi Technological University, Delhi, for providing valuable guidance and constant encouragement throughout the project. It is my pleasure to record my sincere thanks to him for his constructive criticism and insight without which the project would not have shaped as it has.

I humbly extend my words of gratitude to other faculty members of this department for providing their valuable help and time whenever it was required.

Bharti Manjwani

Roll No. 2K13/ISY/05

M.Tech (Information Systems)

E-mail: manjwanibharti@gmail.com

ABSTRACT

It is not always in the best interests to have a single person in control of the data or any sensitive information. This has led to the need for secret sharing schemes. A secret sharing scheme is a method of distributing a secret among a group of users, requiring a cooperative effort to determine the key. This thesis presents a hybrid approach for verifiable secret sharing. Our algorithm shares secret among shareholders where shareholders are also divided into different levels. Thus proposed work includes multi-level secret sharing. Secrets can be recovered within the level or inter level secret sharing is also possible. Shareholders of higher level can contribute their shares in lower levels for secret recovery. To make the scheme verifiable we have used hashing because other schemes depends on hard to solve number theoretic problems or uses any additional information which in turn increases overhead of the protocol. It stands against dishonest dealer and dishonest shareholders. Here we have used one way hash function so that shareholders can verify the shares obtained from the dealer and dealer can also verify shares before accepting it from shareholders before secret reconstruction. There exist many dishonest strategies. VSS is a way to deal with one of the existing dishonest strategy. In VSS schemes neither dealer nor any participant will be able to distribute/submit invalid shares.

Table of Contents

Title	Page no.
CERTIFICATE	ii
ACKNOWLEDGEMENT	iii
ABSTRACT	iv
Figures and Tables	vii
1. INTRODUCTION	1
1.1 Motivation	1
1.2 Goal of mater thesis	2
1.3 Organization of the thesis	2
2. BACKGROUND WORK	3
2.1 Simple Secret Sharing	4
2.2 Threshold Secret Sharing	4
2.2.1 Scheme of Threshold Secret Sharing	5
2.3 Weighted Secret Sharing	8
2.4 Multi-level Secret Sharing	9
2.5 Verifiable Secret Sharing	9
2.5.1 Feldman's Verifiable Secret Sharing	10
2.5.2 Pedersen's Verifiable Secret Sharing	11
2.5.3 VSS based on hashing	11
2.6 Publically Verifiable Secret Sharing	12
2.7 Multi Secret Sharing	12
3. ADDITIONAL CAPABILITIES AND APPLICATIONS	13
3.1 Dealer Leakage Resilient VSS	13
3.2 Hashing	13
3.3 Salted Hashing	14
3.4 Applications of Secret Sharing Schemes	14
3.4.1 Securing Cryptographic keys	15
3.4.2 E-Voting	15

3.4.3 E-Auction	15
3.4.4 Distributed signatures	16
3.4.5 Threshold Scheme for Multiple Servers	16
4. PROPOSED WORK	17
4.1 Notations Used in Proposed Algorithm	17
4.2 Proposed Algorithm	18
4.3 Toy Example	22
5. SECURITY ANALYSIS AND RESULTS	25
5.1 Results	25
5.2 Security Analysis	36
5.3 Comparison	37
6. CONCLUSION AND FUTURE WORK	40
REFERENCES	

Figures and Tables

Fig/Table	Title	Page no.
Figure 2.1	Types of secret sharing schemes	3
Figure 2.2	Threshold secret sharing	5
Figure 3.2	Flowchart of proposed algorithm	21
Figure 5.1	Result 1	26
Figure 5.2	Result 2	27
Figure 5.3	Result 3	28
Figure 5.4	Result 4	29
Figure 5.5	Result 5	30
Figure 5.6	Result 6	31
Figure 5.7	Result 7	32
Figure 5.8	Result 8	33
Figure 5.9	Result 9	34
Figure 5.10	Result 10	35
Table 4.1	Notations used in proposed algorithm	17
Table 4.2	$\delta_{x,i}$ values selected by the dealer	23
Table 4.3	Shares of the shareholders	23
Table 5.1	Comparison on the basis of security property	27
Table 5.2	Comparison on the basis of various parameters	28

In secret sharing (SS) schemes, dealer splits his secret into n parts where n is the number of participants involved in the scheme. After splitting, each shareholder (participant) receives his share of the secret. These participants when combine their shares can recover the secret. This concept was first proposed by Shamir [1] and Blakely [2] in 1979. SS Scheme is referred as thresholding scheme if, t out of n shares are combined, then secret can be recovered. Fewer than t shares give no information about the secret. Verifiable secret sharing (VSS) is an extension of traditional secret sharing. It stands against dishonest dealer and dishonest shareholders. It allows verifying and validating shares. Among many dishonest strategies, VSS is a way to deal with one of the existing dishonest strategy. In VSS schemes, neither dealer nor any participant will be able to distribute/submit invalid shares. To reveal another dishonest strategy of leaking secret information in valid shares, concept of dealer leakage resilience (DLR) was introduced.

1.1 MOTIVATION

Existing approaches for secret sharing using Chinese Remainder Theorem are unconditionally secure but are not verifiable, which is utmost problem in case of dishonest dealer or dishonest shareholder. Our work includes verifiability in multi-level secret sharing. Instead of using traditional approach for verifiability we have used hashing which makes the scheme more secure as well as efficient. Traditional schemes uses modulo exponentiation (where security depends on hardness of solving discrete logarithmic problem) which is costly operation as compare to hashing.

1.2 GOAL OF MASTER THESIS

This thesis presents a hybrid approach for verifiable secret sharing using Chinese Remainder Theorem which includes multi- level secret sharing scheme that is verifiable. Participants are

divided into different levels and secret is splitted into n parts where n is equal to the total number of participants in all the levels. All the n participants will get a piece of every secret. Share distributed among the participants is verified using hashing. Hashing is used to make scheme more secure and reliable. All the participants from authorised set submit their shares for reconstruction of secrets. Shares submitted by the shareholders are also verified using hashing. This ensures neither dealer nor the participants will be able to submit invalid shares.

1.3 ORGANISATION OF THE THESIS

Chapter 2 includes literature review of secret sharing schemes. Basic definition of secret sharing is presented and is followed by threshold secret sharing. This chapter also includes different schemes of threshold SS. After this different types of SS schemes are discussed, like weighted secret sharing, multi-level SS, VSS and at the end multiple SS is also reviewed.

Chapter 3 focuses on additional capabilities of SS schemes like dealer leakage resilience VSS, Salted hashing and some real world applications of SS schemes.

Chapter 4 contains notations used in the scheme and describes proposed approach for Verifiable secret sharing. A toy example and diagrammatic representation of the scheme is also included to make it more explanative.

In Chapter 5 Security of the proposed algorithm is analysed and experimental results are also presented. It also includes a section that compares our scheme with other schemes on the basis of various parameters

.

Conclusion and future directions are presented in Chapter 6.

BACKGROUND WORK

Secret sharing schemes play an important role in cryptography. In secret sharing schemes, there are two phases, one is share generation and other one is secret reconstruction. The secret may be recovered only by certain predetermined groups or authorized groups. Usually such schemes are executed by a dealer and there are N participants or the shareholders who receive shares of the secret by the dealer. N shares (S_1, S_2, \dots, S_N) of a secret S are created by the dealer and are distributed among shareholders. Different types of secret sharing schemes are shown in the Figure 2.1.

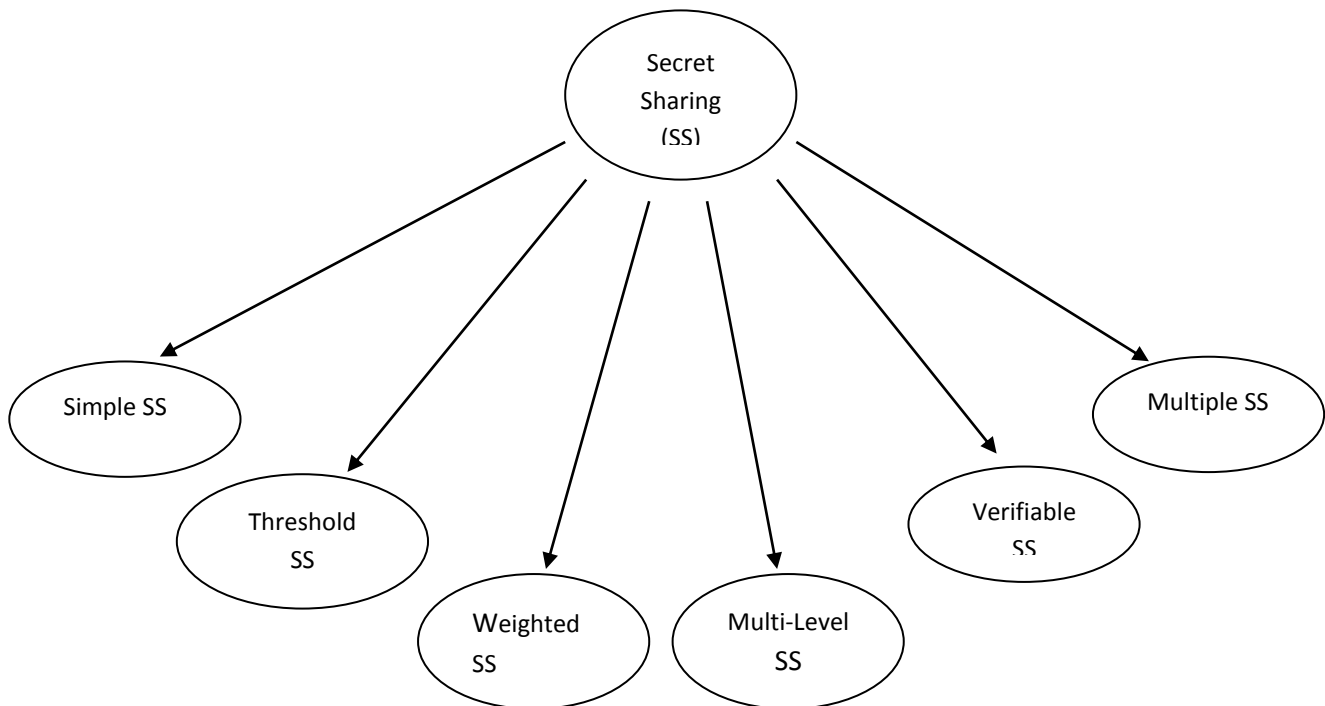


FIGURE 2.1 *Types of Secret Sharing*

2.1 SIMPLE SECRET SHARING

In this scheme there is a dealer and N participants. Dealer possesses a secret which is to be distributed among N participants. Dealer creates N shares of his secret and distributes them among the participants. Now secret can be reconstructed only when all the N participants combine their piece of share. If one or more participants are not present, then no information about the secret is leaked and secret cannot be recovered. Digital data is very sensitive and is more prone to attacks, eavesdrop and manipulation. To prevent such attacks or threats we must keep several copies of the data. But if we keep several copies of the data, there are more chances of secret leakage. Many cryptographic schemes are designed to handle such contradictory situation, secret sharing is one of them. Secret sharing schemes are very powerful tool in the domain of cryptography.

2.2 THRESHOLD SECRET SHARING

This term is an extension of secret sharing. In this instead of all the shares only t shares belonging to an authorised subset are required for secret recovery. This is termed as (t, n) threshold secret sharing scheme (shown in Figure 2.2) where t shares are mandatory for secret reconstruction. Less than t shares will not reveal the secret and also individual share will not leak any information about the secret to any player.

DEFINITION: A (t, n) -threshold SS scheme is a secret sharing scheme that can split a secret $s \in F$ into shares $\{S_1, S_2, \dots, S_n\} \in F$ such that $t \leq n$ and:

1. Given any set of t or more shares, secret can be recovered.
2. Even $t-1$ shares gives no information about the secret.

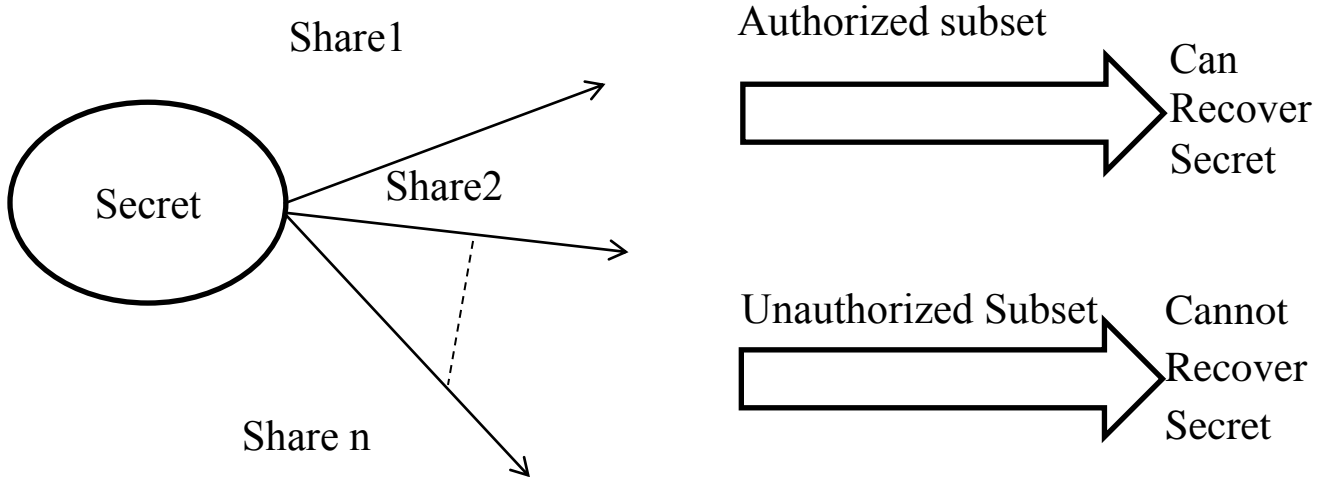


Figure 2.2 Threshold Secret Sharing

2.2.1 Schemes of Threshold Secret Sharing

There are two assumptions in every SS scheme. First it assumes that the dealer is honest and always distributes valid share to every participant of the protocol and the second is, every shareholder submits valid share for secret reconstruction. There is no procedure to deal with fake shares and there is no way to find the validity of the share. Some of the threshold secret sharing schemes are summarised as follows:

I. Shamir's (t, n) Threshold Secret Sharing Scheme

This scheme was proposed by Adi Shamir [1] in 1979. Shamir's scheme is based on Lagrange's polynomial interpolation and his scheme satisfies the basic requirements of secret sharing. Shareholders can unlock the secret if t or more shares are known. Shareholders cannot unlock the secret if less than t shares are known. Shamir's scheme is divided into two algorithms

- **Share Generation**

In this dealer selects a polynomial $f(x)$ of degree $t-1$ whose coefficients are randomly chosen from a finite field by the dealer

$$f(x) = a_0 + a_1 * x + a_2 x^2 + \dots + a_{m-1} x^{m-1}$$

Dealer computes a set of n shares $\{f(1), f(2), \dots, f(n)\}$ and distributes them among the participants through the private channels.

- **Secret Reconstruction**

The secret cannot be reconstructed till t parties are involved. Just as minimum two points are required for finding the equation of a line, three points are required for formulating a quadratic equation, four points for finding the equation of a curve similarly t shares are required to reconstruct equation of degree $t-1$. Shamir's scheme uses Lagrange's polynomial interpolation for polynomial reconstruction

$$f(x) = \sum_{i=1}^t f(i) \prod_{j=1, j \neq i}^t (x-j)/(i-j) \quad (2.1)$$

II. Blakley's (t, n) Threshold Secret Sharing Scheme

As two non-parallel lines lying in the same plane intersect at exactly one point. Similarly two non-parallel planes in a space intersect at one point. This concept can be generalised to n dimensions. n non parallel hyper planes of $n-1$ dimensions will also intersect at a point whose coordinates can be the secrets. This is the main concept behind Blakley's scheme [2] and rest is same as Shamir's polynomial system.

III. Secret Sharing Schemes Based On Chinese Remainder Theorem

These techniques are based on Chinese remainder theorem [17] and not on any interpolation method.

A) Mignotte's Scheme:

This scheme [8] is different from the schemes described above. Its uses Chinese Remainder Theorem (CRT) and a sequence of pairwise co-prime integers (I_1, I_2, \dots, I_n) such that

$$I_1 < I_2 < I_3 \dots < I_n$$

This sequence is termed as mignotte's sequence. This Sequence has a special property

$$I_{n-t+2} * \dots * I_n < I_1 * I_2 * \dots * I_t$$

Following steps are involved in this algorithm:

1. Secret S is chosen between

$$I_{n-t+2} * \dots * I_n < S < I_1 * I_2 * \dots * I_t$$

2. All the participants calculates their piece of share (S_1, S_2, \dots, S_n) as $S \bmod I_i$ (i varies from 1 to n).
3. Secret is recovered using CRT. If t is the threshold then system of congruence is formed.

$$S = S_1 \bmod I_1 \tag{2.1}$$

$$S = S_2 \bmod I_2 \tag{2.2}$$

$$S = S_i \bmod I_i \tag{2.3}$$

Value of S is obtained using equation 2.1, 2.2 and 2.3

$$S = \sum_{i=1}^t (N / I_i) * x_i * S_i \bmod N$$

$$N = I_1 * I_2 * \dots * I_t$$

$$\text{And } (N / I_i) * x_i \bmod I_i = 1$$

B) Asmuth and Bloom's Scheme

Another algorithm [4] that uses CRT for secret sharing. This also uses a special sequence of integers $I_0, I_1 < I_2 \dots < I_n$ which has the special property such that

$$I_0 * I_{n-t+2} * \dots * I_n < I_1 * I_2 * \dots * I_t$$

In this secret S belongs to the set Z_{I_0} . Dealer selects a random value δ and forms, $S + \delta * I_0$ this value should lie between product of smallest t integers and product of any $t-1$ integers. This

condition brings consistency in the system that is it ensures minimum t shares are required to recover or unlock the secret. Thus shares of the secret in this scheme are calculated as:

$$S_i = S + \delta * I_0 \text{ mod } I_i$$

here i varies from 1 to n . This system of congruence is solved using CRT.

$$S' = S_1 \text{ mod } I_1 \tag{2.5}$$

$$S' = S_2 \text{ mod } I_2 \tag{2.6}$$

$$S' = S_i \text{ mod } I_i \tag{2.7}$$

Value of S' is obtained using Equation 2.5, 2.6 and 2.7. Secret is obtained by applying mod operation

$$S = S' \text{ mod } I_0$$

2.3 WEIGHTED SECRET SHARING

In such secret sharing schemes [16], a positive weight is assigned to each participant. Secret reconstruction takes place when sum of weights of authorised subset is equal to the threshold. One of the applications of weighted secret sharing can be an organisation where there is a CEO, Manager and 3 software developers. CEO is given 3 shares of secret, Manager is given 2 and each developer is given a single share. Consider 3 as threshold for secret recovery. Now if CEO wants to recover secret, he alone will be able to unlock secret or Manager along with one of the software developers or 3 software developers will recover the secret. The main Concept of such schemes is to give more shares to important members or participants. This scheme was first proposed by Shamir.

2.4 MULTI LEVEL SECRET SHARING

In hierarchical or multilevel secret sharing schemes [21], participants are divided into m different levels (L_1, L_2, \dots, L_m). A threshold t_i is associated with each level L_i . Number of participants can differ from level to level. N_i denotes number of participants at each level. Assume L_1 is the highest level and L_m is lowest level. Secret recovery at any level is possible if number of participants or shareholders at this level or at higher level in an authorised set is equal to or greater than threshold.

2.5 VERIFIABLE SECRET SHARING AND ITS SCHEMES

In traditional SS schemes, it is assumed that the dealer and shareholders are honest and reliable but in reality dealer doesn't trust the players completely, and therefore it is reasonable to expect that the players do not trust the dealer either. A dishonest dealer may distribute inconsistent shares to the participants and dishonest players may submit fake shares during secret reconstruction. To prevent such cheatings we need a SS scheme which is verifiable. VSS is an extension of secret sharing. A cryptographic SS scheme is verifiable if some auxiliary information is added that helps the shareholders to verify their respective shares. Shareholders will not accept the share if it is inconsistent or invalid. With the help of these schemes it is possible for shareholders to verify their shares without having access to the secrets and even without revealing their shares.

VSS schemes can be interactive and non-interactive. Initially interactive schemes were proposed in which dealer and players communicate in order to check the validity of the shares which increases overhead of the dealer because it has to communicate with N players. Later non-interactive schemes were introduced which reduced dealer's overhead. No communication is required between dealer and players in such schemes.

A (n, t) - VSS (where n is total number of players and t is threshold of secret reconstruction) is a 2-phase protocol (sharing and reconstruction), N be the set of players $N : \{N_1, N_2, N_3, \dots, N_n\}$,

D be the Dealer and S be the secret to be shared and $S \in F$, where F denotes finite field must satisfy the following properties:

- **Secrecy:** if D is honest, no player should get information before reconstruction phase.
- **Commitment:** if D is dishonest, joint view of honest parties output $S^* \in F$ at the end of reconstruction phase.
- **Correctness:** Every party should get access to the Secret only at the end of reconstruction phase.

2.5.1 Feldman's VSS

Feldman's scheme [7] is non-interactive VSS which is based on Shamir's SS scheme and uses some commitment coefficients in order to verify the shares.

- **Secret Verification**

Dealer chooses 2 prime numbers P and Q such that $P-1$ divides Q . A cyclic group G of order P is chosen. g be the generator of the group. Dealers forms a random polynomial in Z_q^* . Here he used shamir's scheme to create shares of a secret. To make scheme verifiable dealer publishes commitments to the coefficients of polynomial. The commitment coefficients in this scheme are given as

$$C_0 = g^s \tag{2.8}$$

$$C_1 = g^{a^1} \tag{2.9}$$

$$C_2 = g^{a^2} \tag{2.10}$$

$$C_{r-1} = g^{a^{r-1}} \tag{2.11}$$

After these values are published, every shareholder will verify their shares using from Equation 2.8 to 2.11

$$g^{S_i} = \prod_{j=0}^{t-1} C_j^{i^j}$$

where S_i is the share of i^{th} shareholder. Secret reconstruction is same as shamir's scheme. Feldman's scheme is not secure as C_0 may leak some information about S .

2.5.2 Pedersen's (n,t,n) VSS

To make scheme more secure, Pedersen changed the commitment function in his scheme. Here 2 (say g and h) elements from the group G are chosen. Dealer selects 2 polynomials $f(x)$ and $g(x)$. Coefficients of $f(x)$ are $S, a_1, a_2, \dots, a_{t-1}$. All the coefficients except the secret are randomly chosen by dealer. Similarly $g(x)$ is a polynomial with coefficients $b_0, b_1, b_2, \dots, b_{t-1}$ randomly chosen by dealer. r is another variable chosen by him. Here committed values are

$$C_0 = g^s * h^r \tag{2.11}$$

$$C_1 = g^{a_1} * h^{b_1} \tag{2.12}$$

$$C_2 = g^{a_2} * h^{b_2} \tag{2.13}$$

$$C_{t-1} = g^{a_{t-1}} * h^{b_{t-1}} \tag{2.14}$$

Each participant verifies his share using Equation 2.11 to 2.14

$$g^{f(i)} * h^{g(x)} = \prod_{j=0}^{t-1} C_j^{i^j}$$

2.5.3 VSS Based on Hashing

Instead of using costly operations to achieve verifiability in the scheme, we can also use hashing which is computationally light as compare to other ways of achieving verifiability In such schemes [3], dealer before distributing shares publishes the hash code of the shares so that shareholders before receiving their shares verifies whether the share is valid or not through the

hash code. If hash code of the share received does not match with the hash code published by the dealer, shareholder will not accept the share and similarly during reconstruction dealer matches the hash code of share with the one he published. If both the hash code does not matches share is considered as invalid. Thus verifiability is achieved in the scheme.

2.6 PUBLICALLY VERIFIABLE SECRET SHARING

Some VSS schemes possesses a special property, that anyone can verify that the distributed shares are valid or not. This special property is referred as Public verifiability and the scheme is Publically VSS scheme. In these schemes validity of the shares can be verified by everyone, not only by the shareholders.

2.7 MULTI SECRET SHARING

There are many applications where sharing a single secret is not sufficient. For such applications notion of multi secret sharing was proposed. Dealer can share multiple secrets using these schemes. Each secret may have different access structure. Access structure is referred to the authorised subset that can unlock the secret.

ADDITIONAL CAPABILITIES AND APPLICATIONS

In this chapter, we will discuss additional capabilities of SS schemes and we will also review possible applications of such schemes.

3.1 DEALER LEAKAGE RESILIENT VSS

VSS captures the one type of dishonest behaviour of the dealer. There exist many other dishonest strategies that can be adopted by the dealer for cheating. Consider the case when dealer tries to subliminally leak information in valid shares. Thus this leads to genuine behaviour of the dealer to every player but gives information about the secret to the attacker. Dealer's malicious behaviour does not get revealed. To overcome such threats DLR-VSS was introduced. DLR-VSS holds the property of verifiability and DLR as well. One of the ways to achieve this property is not to allow the dealer to employ randomness, due to which dealer will not be able to leak any information through valid shares. Communication between Dealer and shareholders outside setting of the protocol is not allowed. If dealer tries to do so, he will be discarded and assumed as faulty dealer.

VSS ensures security in presence of trusted dealer whereas DLR-VSS ensures secrecy even in presence of faulty dealer. Our algorithm holds this property by allowing the shareholders to choose coefficients of polynomial instead of giving this power to dealer. Due to which dealer will not be able to hide secret information in his messages.

3.2 HASHING

In order to ensure verifiability in the algorithm, dealer uses extra information such as check vectors or some sort of encryption mechanism. This increases overhead on the dealer, he needs to compute extra information and distribute the same among shareholders. Other possible ways to achieve verifiability can be modular exponentiation or depends on solving discrete logarithmic

problem. This increases complexity of the algorithm. To overcome such issues, hashing is used to achieve verifiability.

One way hash functions (HF) are used that converts variable length input into fixed length output. Fixed length output obtained by hashing are called hash values or message digest.

A hash function is said to be secure if it holds the following properties:

- **One wayness:** For any input k , its hash values will be

$$k' = HF(k)$$

For any k' it is hard to find k .

- **Hash value collision:** No two inputs yield same message digest. For any inputs k_1 , it is difficult to find another input k_2 which has

$$HF(k_1) = HF(k_2)$$

Using hash functions for the sake of verifiability makes the algorithm computationally light as compare to other methods that are used in VSS schemes.

3.3 SALTED HASHING

In hashing each input is hashed the same way, if two inputs are same they will have same hash codes. This property makes hashing prone to attacks like dictionary attack, brute force attack, look ups and reverse lookups. We can overcome such drawbacks by randomising the hash function. Randomization is introduced in hashing by adding a random number called salt to the input, so that even same input yields different hash codes. Thus adding salt to the hashing makes it more secure.

3.4 APPLICATIONS OF SECRET SHARING SCHEMES

SS schemes have wide application scope. It can be adopted for many real world applications, Some of them are listed below:

3.4.1 Securing Cryptographic Keys

Cryptographic keys play an important role in any cryptosystem. Securing or handling keys are of major concern. One person cannot be trusted for managing keys. Secret sharing schemes are used to handle such situations. Here key is considered as a secret. In such cases, key is splitted into different parts. Each part is termed as share of the key and these shares are distributed to all the participants who will pool their shares for key construction. Individual share does not give information about the key. They are of no use on their own. Threshold of the protocol is set for key reconstruction. Minimum numbers of participants equal to the threshold are required for key reconstruction. Thus securing cryptographic keys is one of the application areas of secret sharing schemes.

3.4.2 E-Voting

Electronic voting also termed as E-voting uses electronic systems for casting and counting votes. In E-voting votes are digitized Confidentiality of the voter is threatened if his vote is decrypted, by the election authorities who are counting the votes. No single authority can be trusted to hand over entire vote of a candidate completely because it can be possible that authority is corrupted and can manipulate the vote. To overcome this issue, Secret sharing schemes can be adopted. Each vote can be treated as a secret and shares of the vote are distributed among the authorities who are counting the votes. Now only t authority can access the vote and it cannot be manipulated by any $t-1$ authorities. Using SS scheme adds security and reliability to the E-Voting system.

3.4.3 E-Auction

E-Auction is a mechanism in which participants bid for the items and item allocation is done based on their bidding prices. E-Auction protocol consists of an auction server, auctioneer, bidders and a bulletin board. In sealed bid auction, each bidder hand over his bid to the auctioneer and after getting all the bids, auctioneer decides the winner based on the bids submitted. It is assumed that auctioneer is an honest member of the system. To make system transparent and reliable an improved E-Auction system can be used where any secret sharing scheme can be adopted. In this, there will be multiple auctioneers and each bid acts as a secret

whose shares are distributed among n auctioneers. t out of n auctioneers will decide winning bid.

3.4.4 Distributed Signatures

A signature is a mathematical way to authenticate a message. It is generally hash code of the message encrypted with a secret key. Sender puts his signature in order to authenticate that message. If there are multiple co-signers, each of them will sign the message one by one according to the priority. But this is not an efficient way because any co-signer can repudiate. Secret sharing schemes can be adopted in such scenario. Signing key will act as secret which will be shares among all the co-signers. Each share is given to each co-signer. No one will have complete control over the secret. No single co-signer knows the signing key. Minimum t co-signers need to pool their shares for signing key construction. Thus the scheme is secure and repudiation is not possible now.

3.4.5 Threshold Scheme for Multiple Servers

In secret sharing schemes a secret is divided into multiple shares. Consider the case when all the multiple shares are stored on different servers. Each server possesses a secret share but a individual share is of no use. Individual share does not leak any information about the secret. Even $t-1$ share does not give any information about the secret where t is threshold. Minimum t shares are required to unlock the secret. This scheme is helpful even if one or two servers meet any kind of failure in that case also secret can be recovered. Another possible advantage of using secret sharing scheme is if any adversary cracks or breaks down one or two server then also adversary will not have access to the secret. Minimum t servers need to be broken for accessing secret.

PROPOSED WORK

4.1 NOTATIONS USED IN PROPOSED ALGORITHM

Notations which are used in our proposed algorithm are represented in Table 4.1 below

TABLE 4.1 *Notations used in proposed algorithm*

Notations	Meaning
D	Dealer
z	Number of levels
L_z	z^{th} level
N_z	Number of shareholders at z^{th} level
t	Threshold
p	A big prime
$I_1^i, I_2^i \dots I_{N_i}^i$	Sequence of N_i pairwise co prime numbers at i^{th} level
δ^i	Random number selected by dealer at i^{th} level
$S_{N_i}^i$	Share of N_i^{th} shareholder at i^{th} level

4.2 PROPOSED ALGORITHM

In our proposed algorithm, we have divided the shareholders into z levels (L_1, L_2, \dots, L_z) where L_1 is highest level of subset and L_z is lowest level. Number of shareholders may vary from level to level. Each Level will have N_i shareholders. For example if $N_3 = 4$ it implies that there are 4

shareholders in level 3. There is a dealer D who wants to share secret among all the shareholders and let t be the threshold of the protocol. Whole protocol is divided into 2 phases, share generation and secret reconstruction. There are two important conditions which are necessary to meet for successful secret reconstruction:

- (i) Secret can be reconstructed if there are t or more valid shares available.
- (ii) Even combination of $t-1$ shareholders will not be able to recover secret.

Each shareholder will keep a share which will be used to reconstruct secret. The whole algorithm is summarised below:

- **Share Generation:** Assume there is a secret S which is shared among shareholders and $S \in Z_p^*$ where p is a big prime.

Case 1: Intra Level Secret Sharing

- (i) D Selects an integer I_0 . For each level, D will select a sequence of pairwise co-prime positive integers which are made public. Integers in each level will be equal to number of shareholders in that level $(I_1^i, I_2^i, \dots, I_{N_i}^i)$ and $I_1^i < I_2^i < \dots < I_{N_i}^i$ where $i = 1, 2, \dots, z$ and gcd of I_0 with every other selected integer should be 1.
- (ii) Dealer forms $S + \delta^i * I_0$ where δ^i is a value selected by the dealer for every level i
 $S + \delta^i * I_0$ should lie between

$$I_{N_i-t+2}^i * I_{N_i-t+3}^i \dots I_{N_i}^i < S + \delta^i * I_0 < I_1^i * I_2^i * \dots * I_t^i$$

This is the threshold range for every level and secrets should lie in this range otherwise algorithm would be inconsistent, i.e. reconstruction can be possible by combing less than t shares. The value which will be shared is:

$$S_{N_i}^i = S + \delta^i * I_0 \pmod{I_{N_i}^i} \tag{4.1}$$

Where i denotes level and N_i denotes shareholder of i^{th} level.

- (iii) Before distributing $S^i_{N_i}$, D computes its hash values and these values are made public so that everyone can access it. Shareholders accept the share if and only if its hash value matches with the hash value published by the dealer otherwise discards it. This mechanism checks the dishonesty of the dealer and makes the scheme verifiable. Thus dealer will not be able to distribute invalid shares.

Case2: Inter Level Secret Sharing

For inter level secret sharing, D needs to select another parameter $I^i_{N_i}$ (In N_i , N is the number of shareholder in i^{th} level). N_i Contributing his share in j^{th} level for secret reconstruction) such that

$$I^j_t < I^i_{N_i, j} < I^j_{N_j-t+2}$$

Then the dealer computes $\Delta S^i_{N_i, j}$. In N_i , N is the number of shareholder at i^{th} level contributing share in j^{th} level.

$$\Delta S^i_{N_i, j} = S + \delta^i * I_0 - S^i_{N_i}$$

In inter level secret sharing, share of the shareholder will be $S^i_{N_i} + \Delta S^i_{N_i, j}$

- **Secret Reconstruction:** A system of equations is formed based on distributed shares. D Accepts shares only if share is valid which is verified using the hash value published by the D before. Equation which is formed is:

Case 1: Intra Level Secret Sharing

$$S^i \text{ mod } I^i_{N_i}$$

Case2: Inter Level Secret Sharing

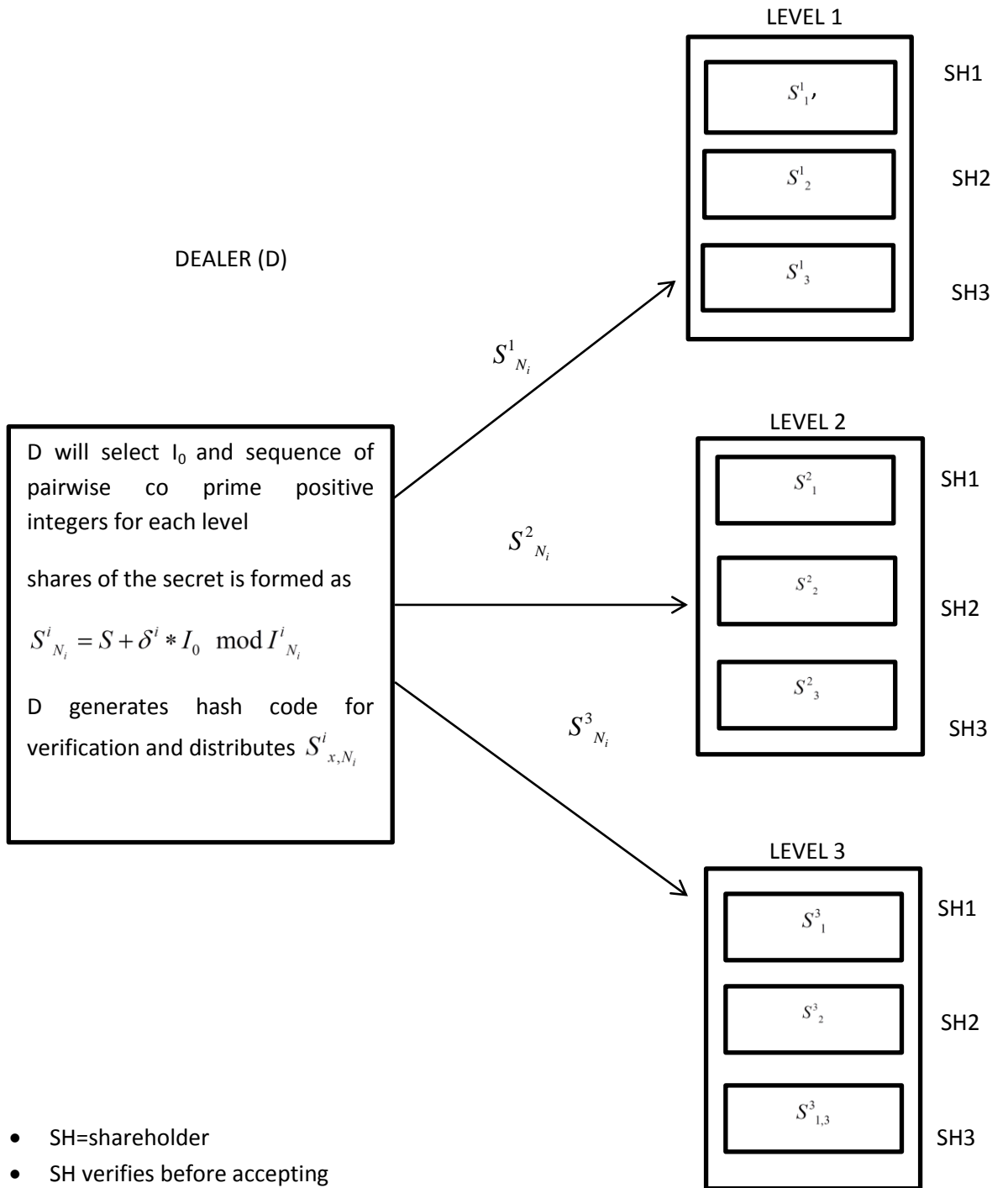
$$S^i_{N_i} + \Delta S^i_{N_i, j} \text{ mod } I^i_{N_i, j}$$

Using the standard Chinese Remainder Theorem (CRT) [17] a unique solution for

$x = S + \delta^i * I_0$. S Can be reconstructed by

$$S = x \text{ mod } I_0$$

Thus the authorized set of shareholders will reconstruct the secret. The whole scheme is demonstrated through Figure 4.1 which shows there are 3 levels and 3 shareholders at each level.



- SH=shareholder
- SH verifies before accepting

- δ random number selected by D $I^i_{N_i}$ is N_i co-prime numbers for each level where N_i is total shareholders at level i.

Figure 4.1 Proposed Algorithm

4.3 TOY EXAMPLE

To explain the proposed scheme, we are using a toy example:

Example: we divided the players into 3 levels ($z = 3$). L_1, L_2 and L_3 are the 3 levels and number of players in these levels be 3, 4 and 7 that is $N_1 = 3, N_2 = 4, N_3 = 7$. Threshold of the protocol is $t = 3$ (Authorized set used for secret reconstruction comprises 3 players). and prime P used is 563. S secret which is to be shared is 128.

- (i) Dealer selects $I_0 = 863$ and sequence of pairwise co-prime integers selected for each level are

- For level 1:

$$I_1 = 137, I_2 = 139, I_3 = 250$$

And threshold range for this level is (34750,4760750)

- For level 2:

$$I_1 = 293, I_2 = 307, I_3 = 313, I_4 = 319$$

And threshold range for this level is (99847,28154663)

- For level 3:

$$I_1 = 229, I_2 = 233, I_3 = 239, I_4 = 241, I_5 = 277, I_6 = 281, I_7 = 283$$

And threshold range for this level is (79523,12752323)

- δ^i value for each level which are shown in Table 4.2.

TABLE 4.2 $\delta_{x,i}$ values selected by the dealer

Levels	δ^i
Level 1	550
Level 2	9864
Level 3	10946

Therefore shares of the shareholder at level 1 will be calculated as

$$128 + 550 * 863 \text{ mod } 137 = 73$$

$$128 + 550 * 863 \text{ mod } 139 = 93$$

$$128 + 550 * 863 \text{ mod } 250 = 28$$

Similarly, all the shares of all the shareholders are calculated which are shown in Table 4.3.

TABLE 4.3 shares of the shareholders

Shareholders	Shares
1 st shareholder of level 1	73
2 nd shareholder of level 1	93
3 rd shareholder of level 1	28
1 st shareholder of level 2	231
2 nd shareholder of level 2	264
3 rd shareholder of level 2	99
4 th shareholder of level 2	245
1 st shareholder of level 3	86
2 nd shareholder of level 3	22
3 rd shareholder of level 3	86
4 th shareholder of level 3	86
5 th shareholder of level 3	218

6 th shareholder of level 3	31
7 th shareholder of level 3	94

Case 1: Intra Level Secret Sharing

When we want to recover secret from level 2, 3 out of 4 will need to contribute their shares to reconstruct the secret, say first 3 are taking part in the protocol for reconstruction. Following system of equation needs to be solved using CRT:

$$X = 231 \text{ mod } 293$$

$$X = 264 \text{ mod } 307$$

$$X = 99 \text{ mod } 313$$

This will give $S = X \text{ mod } 863 = 128$.

Case 2: Inter Level Secret Sharing

Consider 1st shareholder of 1st level and 1st shareholder of 2nd level and 1rd shareholder of 3rd level are contributing their shares in level 3 for reconstruction.

Dealer selects two values (because 2 shares belong to other levels) between 241 and 277 which are co-prime to one another. Say the values are 253 and 263. following system of equations is formed for secret recovery:

$$X = 73 + 192 \text{ mod } 253$$

$$X = 231 + 124 \text{ mod } 263$$

$$X = 86 \text{ mod } 229$$

Solving these equations using CRT we get, $S = X \text{ mod } 863 = 128$.

RESULTS AND SECURITY ANALYSIS

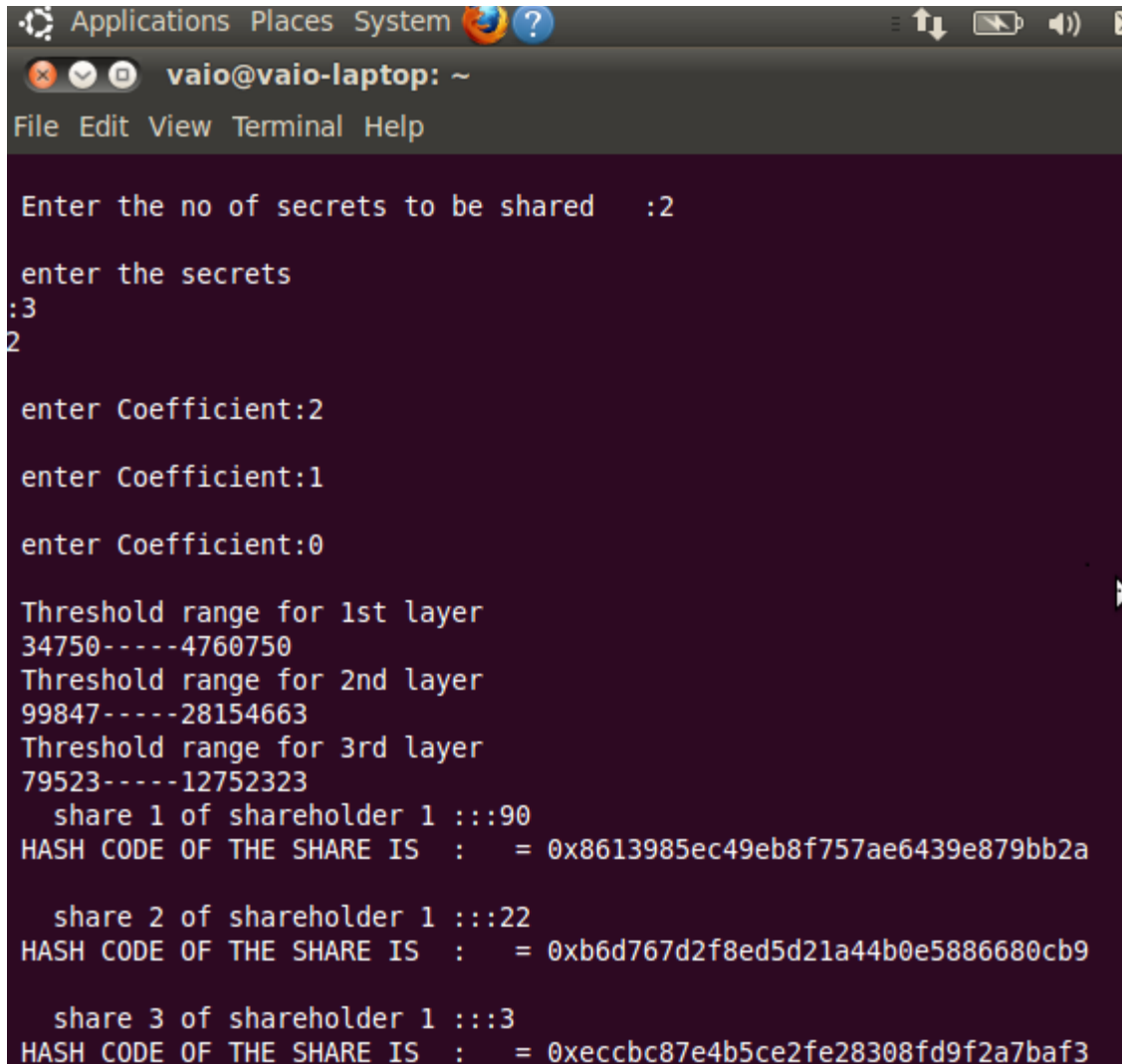
5.1 RESULTS

The algorithm proposed in chapter 4 has been implemented in C++ using NTL and GMP libraries and tested on a 3-GHz, third generation system. Implementation results are presented here with the help of snapshots here with help of a toy example in chapter 4. We have also analysed various parameters used in the scheme and also made comparisons and with some existing schemes in this chapter.

In this implementation results, we have taken 3 levels and there are 3, 4, and 7 shareholders at their respective levels. Threshold of the scheme is 3. The results are as follows:

.

FIGURE 5.1 *Result 1*



```
Applications Places System ?
vaio@vaio-laptop: ~
File Edit View Terminal Help

Enter the no of secrets to be shared :2

enter the secrets
:3
2

enter Coefficient:2
enter Coefficient:1
enter Coefficient:0

Threshold range for 1st layer
34750-----4760750
Threshold range for 2nd layer
99847-----28154663
Threshold range for 3rd layer
79523-----12752323
  share 1 of shareholder 1 :::90
HASH CODE OF THE SHARE IS : = 0x8613985ec49eb8f757ae6439e879bb2a

  share 2 of shareholder 1 :::22
HASH CODE OF THE SHARE IS : = 0xb6d767d2f8ed5d21a44b0e5886680cb9

  share 3 of shareholder 1 :::3
HASH CODE OF THE SHARE IS : = 0xecbc87e4b5ce2fe28308fd9f2a7baf3
```

```
vaio@vaio-laptop: ~
File Edit View Terminal Help

share 4 of shareholder 1 :::45
HASH CODE OF THE SHARE IS : = 0x6c8349cc7260ae62e3b1396831a8398f

share 5 of shareholder 1 :::115
HASH CODE OF THE SHARE IS : = 0x2b44928ae11fb9384c4cf38708677c48

share 1 of shareholder 2 :::112
HASH CODE OF THE SHARE IS : = 0x7f6ffaa6bb0b408017b62254211691b5

share 2 of shareholder 2 :::81
HASH CODE OF THE SHARE IS : = 0x43ec517d68b6edd3015b3edc9a11367b

share 3 of shareholder 2 :::110
HASH CODE OF THE SHARE IS : = 0x5f93f983524def3dca464469d2cf9f3e

share 4 of shareholder 2 :::17
HASH CODE OF THE SHARE IS : = 0x70efdf2ec9b086079795c442636b55fb

share 5 of shareholder 2 :::68
HASH CODE OF THE SHARE IS : = 0xa3f390d88e4c41f2747bfa2f1b5f87db

share 1 of shareholder 3 :::158
HASH CODE OF THE SHARE IS : = 0x06409663226af2f3114485aa4e0a23b4

share 2 of shareholder 3 :::77
HASH CODE OF THE SHARE IS : = 0x28dd2c7955ce926456240b2ff0100bde

share 3 of shareholder 3 :::184
```

FIGURE 5.2 *Result 2*

```
vaio@vaio-laptop: ~
File Edit View Terminal Help

share 3 of shareholder 3 :::184
HASH CODE OF THE SHARE IS : = 0x6cdd60ea0045eb7a6ec44c54d29ed402

share 4 of shareholder 3 :::167
HASH CODE OF THE SHARE IS : = 0x5878a7ab84fb43402106c575658472fa

share 5 of shareholder 3 :::216
HASH CODE OF THE SHARE IS : = 0x45fbc6d3e05ebd93369ce542e8f2322d

share 1 of shareholder 4 :::111
HASH CODE OF THE SHARE IS : = 0x698d51a19d8a121ce581499d7b701668

share 2 of shareholder 4 :::65
HASH CODE OF THE SHARE IS : = 0xfc490ca45c00b1249bbe3554a4fdf6fb

share 3 of shareholder 4 :::226
HASH CODE OF THE SHARE IS : = 0x9cfd10e8fc047a44b08ed031e1f0ed1

share 4 of shareholder 4 :::17
HASH CODE OF THE SHARE IS : = 0x70efdf2ec9b086079795c442636b55fb

share 5 of shareholder 4 :::255
HASH CODE OF THE SHARE IS : = 0xfe131d7f5a6b38b23cc967316c13dae2

share 1 of shareholder 5 :::144
HASH CODE OF THE SHARE IS : = 0x0a09c8844ba8f0936c20bd791130d6b6

share 2 of shareholder 5 :::0
```

FIGURE 5.3 *Result 3*

```
vaio@vaio-laptop: ~
File Edit View Terminal Help

share 2 of shareholder 5 :::0
HASH CODE OF THE SHARE IS : = 0xcgcd208495d565ef66e7dff9f98764da

share 3 of shareholder 5 :::12
HASH CODE OF THE SHARE IS : = 0xc20ad4d76fe97759aa27a0c99bff6710

share 4 of shareholder 5 :::259
HASH CODE OF THE SHARE IS : = 0xcfa0860e83a4c3a763a7e62d825349f7

share 5 of shareholder 5 :::253
HASH CODE OF THE SHARE IS : = 0xc24cd76e1ce41366a4bbe8a49b02a028

share 1 of shareholder 6 :::292
HASH CODE OF THE SHARE IS : = 0x1700002963a49da13542e0726b7bb758

share 2 of shareholder 6 :::158
HASH CODE OF THE SHARE IS : = 0x06409663226af2f3114485aa4e0a23b4

share 3 of shareholder 6 :::35
HASH CODE OF THE SHARE IS : = 0x1c383cd30b7c298ab50293adfecb7b18

share 4 of shareholder 6 :::84
HASH CODE OF THE SHARE IS : = 0x68d30a9594728bc39aa24be94b319d21

share 5 of shareholder 6 :::279
HASH CODE OF THE SHARE IS : = 0xd395771085aab05244a4fb8fd91bf4ee

share 1 of shareholder 7 :::125
```

FIGURE 5.4 *Result 4*

```
vaio@vaio-laptop: ~
File Edit View Terminal Help

share 1 of shareholder 7 :::125
HASH CODE OF THE SHARE IS : = 0x3def184ad8f4755ff269862ea77393dd

share 2 of shareholder 7 :::286
HASH CODE OF THE SHARE IS : = 0x16a5cdae362b8d27a1d8f8c7b78b4330

share 3 of shareholder 7 :::136
HASH CODE OF THE SHARE IS : = 0x42a0e188f5033bc65bf8d78622277c4e

share 4 of shareholder 7 :::203
HASH CODE OF THE SHARE IS : = 0xe2c0be24560d78c5e599c2a9c9d0bbd2

share 5 of shareholder 7 :::69
HASH CODE OF THE SHARE IS : = 0x14bfa6bb14875e45bba028a21ed38046

share 1 of shareholder 8 :::156
HASH CODE OF THE SHARE IS : = 0x1c9ac0159c94d8d0cbec973445af2da

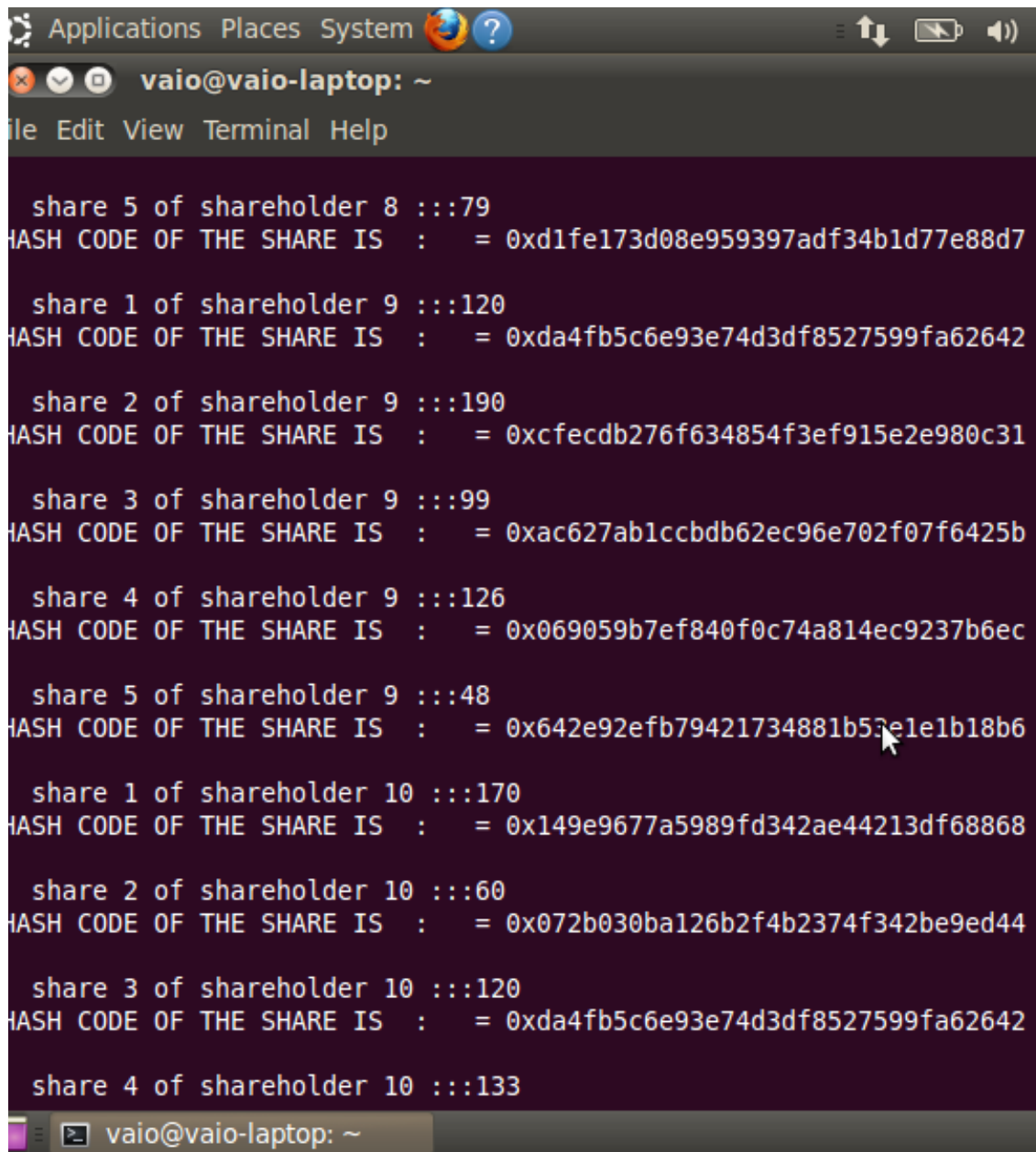
share 2 of shareholder 8 :::106
HASH CODE OF THE SHARE IS : = 0xf0935e4cd5920aa6c7c996a5ee53a70f

share 3 of shareholder 8 :::35
HASH CODE OF THE SHARE IS : = 0x1c383cd30b7c298ab50293adfecb7b18

share 4 of shareholder 8 :::1
HASH CODE OF THE SHARE IS : = 0xc4ca4238a0b923820dcc509a6f75849b

share 5 of shareholder 8 :::79
```

FIGURE 5.5 *Result 5*



The image shows a terminal window on a Linux system. The window title is "vaio@vaio-laptop: ~". The terminal output displays the following information:

```
share 5 of shareholder 8 :::79
HASH CODE OF THE SHARE IS : = 0xd1fe173d08e959397adf34b1d77e88d7

share 1 of shareholder 9 :::120
HASH CODE OF THE SHARE IS : = 0xda4fb5c6e93e74d3df8527599fa62642

share 2 of shareholder 9 :::190
HASH CODE OF THE SHARE IS : = 0xcfecdb276f634854f3ef915e2e980c31

share 3 of shareholder 9 :::99
HASH CODE OF THE SHARE IS : = 0xac627ab1ccbdb62ec96e702f07f6425b

share 4 of shareholder 9 :::126
HASH CODE OF THE SHARE IS : = 0x069059b7ef840f0c74a814ec9237b6ec

share 5 of shareholder 9 :::48
HASH CODE OF THE SHARE IS : = 0x642e92efb79421734881b55e1e1b18b6

share 1 of shareholder 10 :::170
HASH CODE OF THE SHARE IS : = 0x149e9677a5989fd342ae44213df68868

share 2 of shareholder 10 :::60
HASH CODE OF THE SHARE IS : = 0x072b030ba126b2f4b2374f342be9ed44

share 3 of shareholder 10 :::120
HASH CODE OF THE SHARE IS : = 0xda4fb5c6e93e74d3df8527599fa62642

share 4 of shareholder 10 :::133
```

FIGURE 5.6 *Result 6*

```
vaio@vaio-laptop: ~
File Edit View Terminal Help

share 4 of shareholder 10 :::133
HASH CODE OF THE SHARE IS : = 0x9fc3d7152ba9336a670e36d0ed79bc43

share 5 of shareholder 10 :::180
HASH CODE OF THE SHARE IS : = 0x045117b0e0a11a242b9765e79cbf113f

share 1 of shareholder 11 :::170
HASH CODE OF THE SHARE IS : = 0x149e9677a5989fd342ae44213df68868

share 2 of shareholder 11 :::145
HASH CODE OF THE SHARE IS : = 0x2b24d495052a8ce66358eb576b8912c8

share 3 of shareholder 11 :::204
HASH CODE OF THE SHARE IS : = 0x274ad4786c3abca69fa097b85867d9a4

share 4 of shareholder 11 :::234
HASH CODE OF THE SHARE IS : = 0x289dff07669d7a23de0ef88d2f7129e7

share 5 of shareholder 11 :::36
HASH CODE OF THE SHARE IS : = 0x19ca14e7ea6328a42e0eb13d585e4c22

share 1 of shareholder 12 :::152
HASH CODE OF THE SHARE IS : = 0x37a749d808e46495a8da1e5352d03cae

share 2 of shareholder 12 :::227
HASH CODE OF THE SHARE IS : = 0x705f2172834666788607efbfca35afb3

share 3 of shareholder 12 :::19
```

FIGURE 5.7 *Result 7*

```
vaio@vaio-laptop: ~
File Edit View Terminal Help
share 3 of shareholder 12 :::19
HASH CODE OF THE SHARE IS : = 0x1f0e3dad99908345f7439f8ffabdfc4

share 4 of shareholder 12 :::276
HASH CODE OF THE SHARE IS : = 0xdb8e1af0cb3aca1ae2d0018624204529

share 5 of shareholder 12 :::213
HASH CODE OF THE SHARE IS : = 0x979d472a84804b9f647bc185a877a8b5

share 1 of shareholder 13 :::29
HASH CODE OF THE SHARE IS : = 0x6ea9ab1baa0efb9e19094440c317e21b

share 2 of shareholder 13 :::51
HASH CODE OF THE SHARE IS : = 0x2838023a778dfaecdcd212708f721b788

share 3 of shareholder 13 :::51
HASH CODE OF THE SHARE IS : = 0x2838023a778dfaecdcd212708f721b788

share 4 of shareholder 13 :::161
HASH CODE OF THE SHARE IS : = 0xbd4c9ab730f5513206b999ec0d90d1fb

share 5 of shareholder 13 :::112
HASH CODE OF THE SHARE IS : = 0x7f6ffaa6bb0b408017b62254211691b5

share 1 of shareholder 14 :::149
HASH CODE OF THE SHARE IS : = 0xf2217062e9a397a1dca429e7d70bc6ca

share 2 of shareholder 14 :::235
HASH CODE OF THE SHARE IS : = 0x577ef1154f3240ad5b9b413aa7346a1e
```

FIGURE 5.8 *Result 8*


```
Applications Places System vaio@vaio-laptop: ~
File Edit View Terminal Help
HASH CODE OF THE SHARE IS : = 0xf2217062e9a397a1dca429e7d70bc6ca

share 2 of shareholder 14 :::235
HASH CODE OF THE SHARE IS : = 0x577ef1154f3240ad5b9b413aa7346a1e

share 3 of shareholder 14 :::116
HASH CODE OF THE SHARE IS : = 0xc45147dee729311ef5b5c3003946c48f

share 4 of shareholder 14 :::261
HASH CODE OF THE SHARE IS : = 0xb1a59b315fc9a3002ce38bbe070ec3f5

share 5 of shareholder 14 :::237
HASH CODE OF THE SHARE IS : = 0x539fd53b59e3bb12d203f45a912eeaf2

HASH CODE OF THE SHARE IS : = 0xcfcfd208495d565ef66e7dff9f98764da

press
1 to recover secret from level 1
2 to recover secret from level 2
3 to recover secret from level 3
Enter your choice :1

secret is : 3.000000
secret is : 2.000000
secret is : 2.000000
secret is : 1.000000
secret is : 0.000000
INTERLEVEL SECRET SHARING
using shares of
```

FIGURE 5.9 *Result 9*

```
vaio@vaio-laptop: ~  
File Edit View Terminal Help  
secret is : 2.000000  
secret is : 2.000000  
secret is : 1.000000  
secret is : 0.000000  
INTERLEVEL SECRET SHARING  
using shares of  
1st shareholder of level 1  
1st shareholder of level 2  
1st shareholder of level 3  
  
Enter a number between 239 and 281  
253  
263  
  
recovered share ::8  
recovered share ::23  
recovered share ::54  
recovered share ::107  
recovered share ::188  
  
secret is : 3.000000  
secret is : 2.000000  
secret is : 2.000000  
secret is : 1.000000  
vaio@vaio-laptop:~$
```

FIGURE 5.10 *Result 10*

5.2 SECURITY ANALYSIS

Traceability: Algorithm is said to be traceable when it is possible to find out whether any participant during reconstruction phase has submitted any invalid or fake share or not.

Proof: let $f(i)$ be the original valid share and $f'(i)$ is the fake or invalid share. If any participant sends $f'(i)$ to the dealer instead of $f(i)$, then the dealer will not accept the share because

$$H(f(i)) \neq H(f'(i))$$

Where H is any one way hash function and it's very difficult to find 2 values that results the same hash value. Thus the algorithm is traceable.

Robustness: Scheme is said to be robust if all the secrets can be recovered by pooling t or more shares.

$$I_{N_i-t+2}^i * I_{N_i-t+3}^i \dots I_{N_i}^i < S + \delta^i * I_0 < I_1^i * I_2^i * \dots I_t^i$$

Use of above condition has ensured robustness in our scheme. Any t honest players can unlock the shared secret.

Confidentiality: Scheme holds confidentiality if even t-1 players are not able to reveal the secret. Assume $t-1$ participants are available for secret recovery and product of their moduli is X' . These $t-1$ shareholders use CRT to recover a secret. Suppose they obtained a value S' . The relation between original secret and recovered secret will be

$$S = S' + \delta * X'$$

Here S is the original secret. Guessing the correct value of δ to reach original secret is very difficult. Thus we can say that even with t-1 shares, scheme will not leak any information about the secret.

Salted Hashing:

- A. Salted hashing can be used in place of simple hashing. In salted hashing a random number, referred as salt, is added to the share before using one way hash functions. Salted hashing ensures that no two similar secrets yield similar hash codes. But in that case only dealer will be able to verify shares submitted by shareholders. We need to make an assumption that dealer is honest and he will not distribute invalid shares. Just by randomizing the hashes, lookup tables, reverse lookup tables, and rainbow tables become ineffective. An attacker won't know in advance what the salt will be, so they can't pre-compute a lookup table or rainbow table.
- B. Another possible method to use salted hashing and still verification is possible from both ends i.e. shareholders can verify shares before accepting it from dealer and dealer also can verify share before accepting it from shareholders prior to reconstruction phase this can be achieved by treating salt (or random number) as one of the secrets. Constant term of the polynomial will be the salt and degree of polynomial will become $t+k$.

5.3 COMPARISION

Table 5.1 shows the Comparison of the schemes [18, 22, 10, 23,25,5 and 21] on the basis of security property the algorithm holds. We have used acronyms R for robustness, C for confidentiality, Ct for consistency, T for traceability and V for verifiability.

TABLE 5.1 *Comparisons on the Basis of Security Property*

Scheme No.	Robustness(R)	Confidentiality(C)	Consistency (Ct)	Traceability(T)	Verifiability(V)
[18]	Yes	Yes	No	Yes	Yes
[22]	Yes	Yes	No	Yes	Yes
[10]	Yes	Yes	Yes	Yes	Yes
[23]	Yes	Yes	No	Yes	Yes
[25]	Yes	Yes	No	Yes	No
[5]	Yes	Yes	Yes	Yes	Yes
[21]	Yes	Yes	No	Yes	No

Proposed	Yes	Yes	No	Yes	Yes
----------	-----	-----	----	-----	-----

Table 5.1 shows that our proposed approach satisfies all the properties(R, C, Ct, V, T) of VSS, therefore we can say that scheme is verifiable.

Table in 5.2 shows the comparison of our scheme with other schemes [21, 10, 5, 25 and 27] w.r.t various parameters mentioned in the table. Some of them are whether dealer publishes the shares or uses secure channel for distribution, how verifiability is achieved in the scheme using hashing or modular exponentiation. Shareholder before accepting shares verifies whether provided shares are valid or not, this way honesty of the dealer is checked and dealer before accepting shares from shareholders for secret reconstruction verifies whether pooled shares are valid or fake, this way honesty of the shareholders is checked.

Some of the properties are described below:

- Scheme is multi-use if shares of participants are different for different secrets.
- Algorithm can resist conspiracy attack if $t-1$ corrupt shareholders cannot unlock the secret. A Conspiracy resistant scheme ensures that reconstruction of recovered secret does not give information about uncovered secrets.
- SETUP (Secretly Embedded Trapdoor with Universal Protection) is a technique where attacker breaks the security of the system, secret information is leaked but other parties of the protocol are not able to detect this malicious behaviour. All VSS schemes are not SETUP resilient.
- Scheme is unconditionally secure if its security does not depend on any mathematical construct. It is said to be secure even if adversary has unbounded computational power.

TABLE 5.2 *Comparison on the Basis of various parameters*

Property	[21]	[10]	[5]	[25]	[27]	ours
Dealer Publishes the Share	No	No	Yes	Yes	No	No
Use of Hash For Verifiability	No	Yes	No	Yes	---	Yes
Use of Modular Exponentiation Or DLP	No	No	Yes	No	N0	No
Can verify Dealer's honesty	No	Yes	Yes	No	No	Yes
Can verify Shareholder's honesty	No	Yes	Yes	Yes	No	Yes
Has unconditional security	Yes	No	No	No	Yes	No
Conspiracy Attack Resistance	Yes	Yes	Yes	Yes	Yes	Yes

CONCLUSION AND FUTURE WORK

Secret sharing is an important domain of cryptography and is attracting many of the researchers these days. They are taking keen interest in developing efficient schemes which are secured that can be deployed practically. In this thesis we have proposed approach for VSS which focuses on verifiable multi-level secret sharing. Our algorithm holds all properties of VSS except Consistency. In our proposed algorithm, we have divided shareholders into multiple levels, each level may have different shareholders and shares of secret are distributed to the shareholders present in all the levels. Any piece of secret of higher level can be used in lower levels for secret recovery. Secret can only be recovered if authorised subset of participants combining their piece of secret is equal to or greater than predefined threshold value. Verifiability in the scheme is achieved using one way hash functions. Due to which dealer will not be able to distribute invalid shares and shareholders will not be able to submit any fake share during reconstruction phase. Thus the scheme is verifiable and secure.

Some promising future work directions are as follows:

- Finding other method in which each level or layer can have different thresholds instead of a global threshold.
- Find a way to extend single secret sharing scheme to multiple secrets sharing scheme.
- Leaking information in valid shares by the dealer is always a challenge so this should be handled.

REFERENCES

- [1] Adi Shamir, “*How to share a secret*”, Communications of the ACM, vol 22(11),pp 612–613, 1979.
- [2] George Robert Blakley, “*Safeguarding cryptographic keys*”. In Managing Requirements Knowledge, International Workshop , pp 313, IEEE Computer Society, 1899.
- [3] Keyur Parmar & Devesh Jinwala, “*A novel approach for verifiable secret sharing by using a one way hash function*”, 10th National Workshop on Cryptology Department of Mathematics and Computer Applications, September 2 – 4, 2010.
- [4] Charles Asmuth and John Bloom, “*A Modular Approach to Key Safeguarding*”, IEEE Transactions on Information Theory, Vol. IT-29, no. 2, March 1983.
- [5] Ruxandra F. Olimid, “*Dealer-Leakage Resilient Verifiable Secret Sharing*”, Department of Computer Science, University of Bucharest, Romania, September 19, 2014.
- [6] Gustavus J. Simmons, “*The subliminal channel and digital signature*”, In EUROCRYPT, pp 364-378, 1984.
- [7] Feldman, “*A practical scheme for non-interactive verifiable secret sharing*”, Proceedings of the 28th Annual Symposium on Foundations of Computer Science, 1987
- [8] M. Mignotte, “*How to share a secret*”, Workshop on Cryptography, Springer, Heidelberg, pp 371–375, 1983.
- [9] Kamer Kaya and Ali Aydın Selçuk, “*A Verifiable Secret Sharing Scheme Based on the Chinese Remainder Theorem*”, INDOCRYPT, Lecture Notes Computer Science, vol 5365, pp 414–425, 2008.
- [10] Jun Shao, “*Efficient verifiable multi-secret sharing scheme based on hash function*”, Information Sciences, pp 104–109, 2014.
- [11] Alexandre Ruiz and Jorge L. Villar, “*Publicly Verifiable Secret Sharing from Paillier’s Cryptosystem*”, 2005.
- [12] Qassim Al Mahmoud, “*Polynomial differential-based strong (n, t, n) -verifiable secret sharing*”, IET Information Security, January 2013.

- [13] Divya G Nair, Binu V P, G. Santhosh Kumar, “*An Improved E-voting scheme using Secret Sharing based Secure Multi-party Computation*”, Cochin University of Science and Technology, February 2015.
- [14] Shashank Agrawal, “*Verifiable secret sharing in a total of three rounds*”, in Elsevier, Information Processing Letters, 2012.
- [15] J. Benaloh, “*Verifiable Secret-Ballot Elections*”, PhD thesis, Department of Computer Science, Yale University, September 1987.
- [16] S. Iftene and I. Boureanu, “*Weighted threshold secret sharing based on the Chinese remainder theorem*”, Scientific Annals of the “Al. I. Cuza” University of Iasi, Computer Science Section, XV:161–172, 2005.
- [17] S. Iftene and F. Chelaru, “*The general Chinese remainder theorem*”, 2006.
- [18] J. Shao, Z. Cao, “*A new efficient (t, n) verifiable multi-secret sharing (VMSS) based on YCH scheme*”, Applied Mathematics and Computation vol 168 (1), pp 135–140, 2005.
- [19] Lein Harn a and Changlu Lin, “*Strong (n, t, n) verifiable secret sharing scheme*”, in ELSEVIER, Information Sciences 180 ,pp 3059–3064, 2010.
- [20] Youliang Tian, Jianfeng Ma, Changgen Peng, and Qi Jiang, “*Fair (t, n) threshold secret sharing scheme*”, IET Information Security”, 2013.
- [21] Lein Harn and Miao Fuyou, “*Multilevel threshold secret sharing based on the Chinese Remainder Theorem*”, in ELSEVIER, Information Processing Letters 114 ,pp 504–509, 2014.
- [22] M.H. Dehkordi, S. Mashhadi, “*An efficient threshold verifiable multi-secret sharing*”, Computer Standards and Interfaces vol 30 (3) pp 187–190, 2008.
- [23] J. Zhao, J. Zhang, R. Zhao, “*A practical verifiable multi-secret sharing scheme*”, Computer Standard Interfaces vol 29 (1) pp 138–141, 2007.
- [24] Changlu Lin, Lein Harn and Dingfeng Ye, “*Information-Theoretically secure strong verifiable secret sharing*”, SECryp- International Conference on Security and Cryptography, 2009.
- [25] A. Das, A. Adhikari, “*An efficient multi-use multi-secret sharing scheme based on hash function*”, Applied Mathematics Letters vol 23 (9) pp 993–996, 2010.
- [26] T.-Y. Chang, M.-S. Hwang, W.-P. Yang, “*An improvement on the Lin–Wu (t, n) threshold verifiable multi-secret sharing scheme*”, Applied Mathematics and Computation vol 163(1) pages 169–178, 2005.

[27] Lein Harn, “*Secure secret reconstruction and multi-secret sharing Schemes with unconditional security*”, Security and Communication Networks, pp 567-573, 2014.