Multisecret-sharing scheme with two-level security and its applications in Blockchain

R. K. Sharma, R. Sarma, Neha Arora and Vidya Sagar

Department of Mathematics, Indian Institute of Technology Delhi, Hauz Khas, New Delhi-110016, India¹.

September 9, 2022

Abstract

A (t, m)-threshold secret sharing and multisecret-sharing scheme based on Shamir's SSS are introduced with two-level security using a one-way function. Besides we give its application in smart contract-enabled consortium blockchain network. The proposed scheme is thoroughly examined in terms of security and efficiency. Privacy, security, integrity, and scalability are also analyzed while applying it to the blockchain network.

MSC 2020: Primary 94A62; Secondary 94A60.

Keywords: Secret sharing scheme, Multisecret-sharing scheme, One-way function, Hash Function, Blockchain, Smart Contract.

1 INTRODUCTION

Let $m, t \in \mathbb{N}$ with $1 < t \le m$. In secret sharing scheme, we divide the given data \mathbf{s} , which is the *secret*, into m pieces, each of which is called as *share*. One can obtain \mathbf{s} with the help of any t or more shares but not with t-1 or less shares. Such scheme is called as a (t,m)-threshold secret sharing scheme. Various cryptosystems which are based on single key have many shortcomings. For example, if the key is maliciously or accidentally disclosed to the public, or if the key's owner is found to be untrustworthy [16], the entire system will be jeopardized, thus secret sharing schemes (SSS) are becoming essential nowadays; in fact, these are used heavily in electronic voting systems, cryptographic protocols, banking systems, etc.

In 1979, Blakley [3] and Shamir [13] introduced secret sharing scheme independently. To solve secret sharing problem, linear projective geometry was used in [3] whereas Lagrange interpolation polynomial was used in [13]. After that, this topic got a lot of attention and researchers investigated various types of SSS, including On-line Schemes, Rational Schemes, Quantum Schemes, Chinese Remainder Schemes, Visual Schemes, Multiple Schemes, Proactive Schemes, Verifiable Schemes, Ideal Schemes, Linear Schemes, etc. in recent past. Multisecret-sharing schemes (MSS) are well-known among the secret sharing

¹E-mail addresses: rksharmaiitd@gmail.com (Rajendra Kumar Sharma), ritumoni407@gmail.com (Ritumoni Sarma), nehaarora1907@gmail.com (Neha Arora), vsagariitd@gmail.com (Vidya Sagar).

scheme families. As the name suggests, in MSS [5, 19], we have multiple secrets to be shared instead of a single secret.

Blockchain is an open, decentralized, and distributed ledger that can record transactions among multiple parties in an efficient way. Each transaction is hashed (that is, digitally signed with cryptographically secured function) and then stored in blocks. Each block is linked with the previous block hash that makes it immutable. The concept of blockchain came into light in 2008, when a white paper [11] was published on virtual cryptocurrency Bitcoin by an anonymous person Satoshi Nakamoto. Later, it was made functional in 2009. To enhance the privacy and security of blockchain, smart contracts [10] were used. It was first introduced by Nick Szabo in [14].

In [12], the authors used Shamir's secret sharing scheme to distribute transaction data using private key encryption and distributed storage blockchain, keeping the data integrity intact. Similarly, in [8], the authors proposed local secret sharing and applied it on distributed storage blockchain with the aim of reducing the storage and communication cost. A few applications of secret sharing schemes on the blockchain network have also been proposed in various sectors like healthcare [15], smart city architecture [4], supply chain [18], etc. Their objective is to ensure the privacy and security of data from adversaries.

However, a secret sharing scheme has the limitation of dishonest dealers or participants, but none of them discussed about dishonest dealers and participants simultaneously. Also it is important to ensure that both the dealer's committee and participants (or miners) are honest. Since dealer plays the central role, it is assumed that the dealer must be honest. Also, in recent years, many blockchain based secret sharing schemes were introduced to outcome the limitation of dishonest participants. To protect the secret from attackers, many authors used the threshold t/m less than 1/2 [1, 6, 7, 9, 17].

In [2], the authors have discussed Dynamic Proactive Secret Sharing (DPSS) scheme, where dealers and participants keep on changing and it is based on honest majority. Then, they have discussed Evolving-Committee Proactive Secret Sharing (ECPSS) scheme, which is a combination of DPSS and committee-selection protocol. They assume that either PoW or Cryptographic sortition can be done to choose the desired committee. It decreases the probability of corrupt members in the committee and restricts the adversary to know about the committee. Then, they have defined Target-Anonymous Channels, which keep the receivers (participants who receive shares of the secret) anonymous.

In this manuscript, first we define SSS and MSS based on Shamir's SSS with two-level security, where initially we check the honesty of participants. Further, only honest participants will get their share for the computation of the secret s. Then we apply our scheme on blockchain network. For this, we replace dealer with a team (or committee) of dealers, who need to prove their honesty using non-interactive zero-knowledge proofs before involving in the process of secret generation and distribution. Also, for the generation of a new block, a fresh committee will be formed, depending on the nodes involved in the transaction process. However, committee will have a predetermined minimum and maximum number of nodes and committee keeps on changing with the change in block. Moreover, m participants will be chosen in an anonymous way. Once any t out of them are able to retrieve the secret and validate the transactions, a new block will be formed and added to the chain. In this case, if there are a few cheating participants and they try to find the secret by involving themselves in the secret recovery process, even then they will not be able to proceed to find the secret unless they retrieve the correct encrypted value of the secret, where encryption is done using a one-way function.

The rest of the manuscript is arranged as follows. Section 2 includes preliminaries. In

Section 3, we propose our scheme. Application of the scheme on blockchain network is discussed in section 4. In Section 5, we analyze our scheme on the basis of its efficiency and security. Also, we examine the privacy, security, integrity, and scalability of the scheme while applying it on blockchain in this section. Section 6 concludes the manuscript.

2 Definitions and Preliminaries

Definition 2.1. Secret sharing scheme is a way in which one (called dealer) distributes the secret to multiple people (called participants) in such a way that they can collectively recover the secret but individually they can't.

Let the secret be distributed to m participants. If the secret can only be recovered by any t or more participants then t is said to be the *threshold* of the scheme, where $1 < t \le m$. A (t, m)-threshold secret scheme is a scheme with threshold t and m participants.

Shamir's Secret sharing scheme [13]: In this scheme, Shamir has taken two entities: a dealer and a set of participants. Dealer is the one who knows the secret \mathbf{s} and distributes their shares (pieces of the secret) to all participants in such a way that it follows the basic properties of SSS and its threshold. For this, he has constructed a polynomial f(x) of degree t-1 such that constant term of the polynomial is \mathbf{s} and the remaining coefficients are randomly generated.

- Secret generation: Dealer chooses random variables, say $r_1, r_2, \ldots, r_{t-1}$ and generates the polynomial $f(x) = \mathbf{s} + r_1 x + r_2 x^2 + \cdots + r_{t-1} x^{t-1}$ and then enumerates the participants and computes $f(1), f(2), \ldots, f(m)$.
- Secret distribution: Distribute (i, f(i)) to the i^{th} participant, $1 \le i \le m$.
- Secret recovery: Any t or more participants come together to combine their shares (i, f(i)) to compute the polynomial f(x) with the help of Lagrange interpolation polynomial.

Definition 2.2. A (t, m)-threshold secret sharing scheme is said to be *perfect* if any \tilde{k} (where $\tilde{k} < t$) participants cannot recover the secret.

Definition 2.3. *Hash function* is a function which maps bit string of arbitrary length to a bit string of fixed length in random manner.

Cryptographically secure hash function is a hash function which satisfies the following properties: one-wayness, collision resistant, target collision resistant, non-malleability and pseudo-randomness.

Remark 2.4. Throughout this paper, by a hash function, we mean a Cryptographically secure hash function.

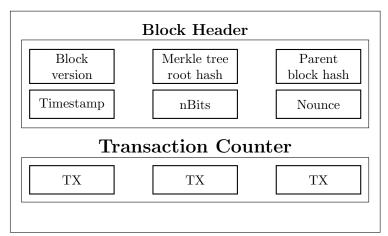
Definition 2.5. A function $T: A \to B$ is said to be *one-way function* if for any $b \in B$, it is computationally hard to find some $a \in A$ such that T(a) = b in polynomial time.

Definition 2.6. Distributed ledger is a type of database which records the data or information in such a way that it will be shared and replicated in its most updated form to all the members available on the decentralized network. Participants in the network agree on the consensus and update the database timely using cryptographic signature, which makes it auditable for the remaining members.

Definition 2.7. [20] A *block* is a container that contains a series of transactions. A block is divided into two components: block header and transaction counter. Block header contains the following.

- Block version: It indicates the validation rules for the block.
- Merkle tree root hash: The aggregate of hash value of all transactions.
- Timestamp: Current time in seconds/minutes since the starting of the network.
- nBits: It is related to the difficulty level for the computation of new hash and its size in bits.
- Nounce: A variable which keeps on increasing with every hash calculation and PoW done.
- Parent block hash: Hash value of the previous block.

Transaction counter contains the transactions. Maximum number of transactions that a block can contain depends on the size of the block, the size of each transaction, and the total number of transactions occurred in a fixed time interval.

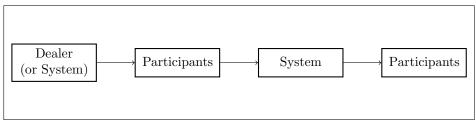


Block Structure

Definition 2.8. A smart contract is a digital contract which enforces the members to follow the procedure and instructions mentioned in the contract. It ensures smooth functioning of the system and verification of the transactions without any interruption from the outside network. They are immutable and distributed. It also provides security to the network.

Definition 2.9. A zero-knowledge proof is a protocol by which a prover can prove the knowledge of certain information (say, x) without revealing it and the verifier can verify it without getting any information about x. A zero knowledge proof must posses three properties: completeness, soundness, and zero-knowledge.

3 Proposed Schemes



Communication Channel

In previous secret sharing schemes, without testing the honesty of the participant, it is assumed that majority of them are honest and thus they become eligible to compute the secret. However the hypothesis of honest participants may fail. Here we have defined two steps secret sharing scheme where the first step is only for verifying if the active participants are honest or not. In this step, instead of sending the shares for the computation of the secret \mathbf{s} , dealer(or system) shares the information (on similar lines as we have done in section 3.1 and 3.2 or using any available SSS) for the computation of $H(\mathbf{s})$, where H is a one-way function. Once $H(\mathbf{s})$ is computed correctly, the system shares the information, for the computation of the secret \mathbf{s} , only to those participants, who participated in the computation of $H(\mathbf{s})$ and then they can collectively compute the secret \mathbf{s} .

We have proposed the secret sharing scheme by applying the above procedure on Shamir's SSS for the single secret \mathbf{s} and for multi secret $\mathbf{s} = (s_1, s_2, \dots, s_m)$.

3.1 Generalization of Shamir's Secret Sharing Scheme with two-level security

3.1.1 Set up Phase:

In this scheme, we have $\{P_1, P_2, P_3, \ldots, P_m\}$ as m participants and (t, m) is the threshold. Dealer D (can be replaced by the system S) chooses $a_i \in \mathbb{F}_p^*$, where p is a large prime such that $m \ll p$, to be the public key of P_i respectively such that $a_i \neq a_j$ for $i \neq j$. Let $\mathbf{s} \in \mathbb{F}_p$ be the secret.

Set up 1. $p \leftarrow$ Prime (large) 2. $(t, m), 1 < t \le m \leftarrow$ Threshold 3. m << p4. $P_1, P_2, \dots, P_m \leftarrow$ Participants 5. $D \leftarrow$ Dealer 6. $a_i \in \mathbb{F}_p^*, a_i \ne a_j \ \forall \ i \ne j \leftarrow$ Public key of P_i .

3.1.2 Computing and distributing shares:

Dealer chooses a one-way function H, and random elements $r_1, r_2, \ldots, r_{t-1} \in \mathbb{F}_p$, and then computes $H(r_1), H(r_2), \ldots, H(r_{t-1})$. He then generates the polynomials f(x) and h(x) as

follows:

$$f(x) = \mathbf{s} + r_1 x + r_2 x^2 + \dots + r_{t-1} x^{t-1}$$
(3.1)

$$h(x) = H(\mathbf{s}) + H(r_1)x + H(r_2)x^2 + \dots + H(r_{t-1})x^{t-1}$$
(3.2)

Dealer then computes $(h(a_i), f(a_i))$ and initially shares $h(a_i)$ to the participant P_i $(1 \le i \le m)$.

Dealer (for single secret)

- 1. $H \leftarrow$ One-way function
- 2. $r_1, r_2, \dots, r_{t-1} \in \mathbb{F}_p \longleftarrow$ Random elements
- 3. $\mathbf{s} \longleftarrow \text{Secret}$
- 4. Compute $H(r_1), H(r_2), \dots, H(r_{t-1})$
- 5. Generate f(x) and h(x) by:

$$f(x) = \mathbf{s} + r_1 x + r_2 x^2 + \dots + r_{t-1} x^{t-1}$$

$$h(x) = H(\mathbf{s}) + H(r_1) x + H(r_2) x^2 + \dots + H(r_{t-1}) x^{t-1}$$

Output : $(a_i, f(a_i), h(a_i)); 1 \le i \le m \longrightarrow \text{System}$

3.1.3 Recovering the secret:

Any t or more participants, upon receiving $h(a_i)$ corresponding to their public key a_i , can come forward to compute the polynomial h(x) by using Shamir's secret sharing scheme and then share its constant term $H(\mathbf{s})$ to the system.

System verifies $H(\mathbf{s})$ and after confirming the honesty of minimum of t-participants, it reveals $f(a_i)$ to only those t-participants P_i , who take part in computation of h(x) and passes the honesty test. Then, they can finally recover the secret \mathbf{s} .

Participants: $t \leftarrow$ minimum number of participants required to recover the secret

- 1. At least t participants interact with the system to find the secret.
- 2. Input: $(a_i, h(a_i))$ by at least t participants
- 3. Output: $h(x) \longrightarrow \text{System}$
- 4. System verifiers h(x)

less than t participants are honest

STOP

else:

send $f(a_i)$ to participating members

- 5. Input: $(a_i, f(a_i))$
- 6. Output: f(x)
- 7. Compute the secret **s**.

3.2 Generalization of Shamir's Secret Sharing Scheme for multi-secret with two-level security

3.2.1 Set up Phase:

In this scheme, we have assumed the same set up as we have done in 3.1.1 for single secret. Let $\mathbf{s} = (s_1, s_2, \ldots, s_m)$ be the secret such that $s_i \in \mathbb{F}_p$. Also, the first k $(1 < k \le t)$ bits of \mathbf{s} are message bits and remaining m - k bits are parity bits with $t \le m - 1$.

3.2.2 Computing and distributing shares:

Dealer chooses a one-way function H. He computes $\tilde{s} = \sum_{i=1}^{m} s_i$ and makes it public. He then computes α_i , and $H(\alpha_i)$ for each $i \in \{1, 2, ..., t\}$, where

$$\alpha_i = \sum_{\substack{j=1\\j\neq i}}^m s_j.$$

He then generates the polynomials f(x) and h(x) as follows:

$$f(x) = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \dots + \alpha_t x^{t-1},$$
 (3.3)

$$h(x) = H(\alpha_1) + H(\alpha_2)x + H(\alpha_3)x^2 + \dots + H(\alpha_t)x^{t-1}.$$
 (3.4)

Dealer then computes $(h(a_i), f(a_i))$, where a_i is the public key of i^{th} participant P_i and initially shares $h(a_i)$ to the participant P_i $(1 \le i \le m)$.

Dealer (for multi secret)

- 1. $H \leftarrow$ One-way function
- 2. $\mathbf{s} = (s_1, s_2, \dots, s_m) \leftarrow$ Secret such that the first k bits $(1 < k \le t)$ are message bits and the remaining n k bits are parity bits.
- 3. Compute $\tilde{s} = \sum_{i=1}^{m} s_i \leftarrow$ Public to all
- 4. Compute $\alpha_i = \sum_{\substack{j=1 \ j \neq i}}^m s_j$ where $1 \le i \le t$
- 5. Compute $H(\alpha_1), H(\alpha_2), \ldots, H(\alpha_t)$
- 6. Generate f(x) and h(x) by: $f(x) = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \dots + \alpha_t x^{t-1}$ $h(x) = H(\alpha_1) + H(\alpha_2) x + H(\alpha_3) x^2 + \dots + H(\alpha_t) x^{t-1}$

Output : $(a_i, f(a_i), h(a_i)) \ \forall \ 1 \le i \le m \longrightarrow \text{system}$

3.2.3 Recovering the secret:

Any t or more participants, upon receiving $h(a_i)$ corresponding to their public key a_i , can come forward to compute the polynomial h(x) by using Shamir's secret sharing scheme and then share it with the system.

System verifies h(x) and after confirming the honesty of minimum of t-participants, it reveals $f(a_i)$ to only those t-participants, who take part in the computation of h(x) and passes the honesty test. Then, they can recover the polynomial f(x).

Once, f(x) is recovered, participants can compute $s_i = \tilde{s} - \alpha_i \ \forall \ 1 \leq i \leq t$.

Example 3.1. Suppose p = 199. Let (5, 11) be the threshold and $P_1, P_2, P_3, \ldots, P_{11}$ be the 11 participants and $(a_1, a_2, \ldots, a_{11}) = (7, 5, 4, 3, 2, 9, 6, 8, 11, 10, 12)$, where a_i is the public key of P_i .

Let s = (7, 9, 2, 3, 7, 5, 4, 9, 3, 21, 27) be the secret, where only the first 5 bits are message bits and the remaining 6 bits are parity bits.

Suppose $H: \mathbb{F}_{199} \longrightarrow \mathbb{F}_{199}$ defined by $H(n) = g^n$, where $g \in \mathbb{F}_{199}^*$ is the one-way function. Take g = 3.

Then $\tilde{s} = \sum_{i=1}^{11} s_i = 97$ and it is made public.

Further, $\alpha_1 = 90, \alpha_2 = 88, \alpha_3 = 95, \alpha_4 = 94, \alpha_5 = 90$ and $H(\alpha_1) = 188, H(\alpha_2) = 43, H(\alpha_3) = 113, H(\alpha_4) = 104, H(\alpha_5) = 188$ and therefore

$$f(x) = 90 + 88x + 95x^2 + 94x^3 + 90x^4$$

and

$$h(x) = 188 + 43x + 113x^2 + 104x^3 + 188x^4.$$

Dealer then computes $f(a_i)$ and $h(a_i)$ for each i and share it with the system, which is displayed in the following table.

i	1	2	3	5	5	6	7	8	9	10	11
a_i	7	5	4	3	2	9	6	8	11	10	12
$f(a_i) \pmod{199}$	167	61	173	92	52	147	90	170	70	117	116
$h(a_i) \pmod{199}$	163	0	38	67	188	40	185	36	65	147	34

System initially shares $h(a_i)$ to P_i for each i and any 5 or more participants can decrypt the polynomial h(x). Without loss of generality, assume P_1, P_2, P_3, P_4, P_5 come together to recover the polynomial h(x), then they have the following.

i	1	2	3	4	5
a_i	7	5	4	3	2
$h(a_i)$	163	0	38	67	188

Since h(x) is a polynomial of degree 4 and the active participants have its value at 5 distinct points, they use Lagrange Interpolation to compute the polynomial h(x). Participants will now share it with the system and after verifying, the system will share the corresponding values of the polynomial f(x) to respective the participants as follows.

i	1	2	3	4	5
a_i	7	5	4	3	2
$f(a_i)$	167	61	173	92	52

Participants will now compute the polynomial f(x) by the same method. Then, they compute $s_j = \tilde{s} - \alpha_j = 97 - \alpha_j \quad \forall \ 1 \leq j \leq 5$.

j	1	2	3	4	5
α_j	90	88	95	94	90
s_{j}	7	9	2	3	7

Example 3.2. Suppose p = 113. Let (9, 17) be the threshold and $P_1, P_2, P_3, \ldots P_{17}$ be the 17 participants and $a_i = i$ be the public key of P_i .

Let s = (3, 5, 7, 9, 11, 3, 5, 6, 2, 1, 7, 8, 6, 2, 5, 1, 4) be the secret, where only the first 6 bits are message bits and remaining 11 bits are parity bits.

Note that message bits is less than the threshold.

Suppose $H: \mathbb{F}_{113} \longrightarrow \mathbb{F}_{113}$ defined by $H(n) = n^2 \pmod{113}$ is the one-way function.

Then
$$\tilde{s} = \sum_{i=1}^{17} s_i = 85 \pmod{113}$$
 and it is made public.

Further, $\alpha_1 = 82, \alpha_2 = 80, \alpha_3 = 78, \alpha_4 = 76, \alpha_5 = 74, \alpha_6 = 82, \alpha_7 = 80, \alpha_8 = 79, \alpha_9 = 83$ and $H(\alpha_1) = 57, H(\alpha_2) = 72, H(\alpha_3) = 95, H(\alpha_4) = 13, H(\alpha_5) = 52, H(\alpha_6) = 57, H(\alpha_7) = 72, H(\alpha_8) = 26, H(\alpha_9) = 109$ and therefore

$$f(x) = 82 + 80x + 78x^{2} + 76x^{3} + 74x^{4} + 82x^{5} + 80x^{6} + 79x^{7} + 83x^{8}$$

and

$$h(x) = 57 + 72x + 95x^{2} + 13x^{3} + 52x^{4} + 57x^{5} + 72x^{6} + 26x^{7} + 109x^{8}.$$

Dealer then computes $f(a_i)$ and $h(a_i)$ for each i and share it with the system, which is displayed in the following table.

a_i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
$f(a_i)$	36	92	92	63	16	9	28	96	68	104	52	106	84	47	73	3	41
$h(a_i)$	101	83	44	108	1	65	66	89	100	37	3	105	19	27	45	25	0

System initially shares $h(a_i)$ to P_i for each i and any 9 or more participants can decrypt the polynomial h(x). Without loss of generality, assume P_1, P_2, \ldots, P_9 come together to recover the polynomial h(x), then they have the following.

a_i	1	2	3	4	5	6	7	8	9
$h(a_i)$	101	83	44	108	1	65	66	89	100

Since h(x) is a polynomial of degree 8 and the active participants have its value at 9 distinct points, they use Lagrange Interpolation to compute the polynomial h(x). Participants will now share it with the system and after verifying, the system will share the corresponding values of the polynomial f(x) to the respective participants as follows.

a_i	1	2	3	4	5	6	7	8	9
$f(a_i)$	36	92	92	63	16	9	28	96	68

Participants will now compute the polynomial f(x) by the same method. Then, they compute $s_j = \tilde{s} - \alpha_j = 85 - \alpha_j \quad \forall \ 1 \leq j \leq 9$.

j	1	2	3	4	5	6	7	8	9
α_j	82	80	78	76	74	82	80	79	83
s_{j}	3	5	7	9	11	3	5	6	2

Since, only first 6 bits are message bits, thus required message is (3, 5, 7, 9, 11, 3).

4 Multisecret-sharing scheme on a Blockchain Network

4.1 Blockchain Architecture

Blockchain is a chain of virtual blocks in which each block contains certain information along with its hash and the hash of the previous block. In this subsection, we will demonstrate how the blocks of blockchain are formed with the help of the proposed scheme. We impose a few assumptions and then define the structure of a blockchain network to efficiently apply this scheme as follows.

- Type of blockchain: We assume that our platform is smart contract-enabled consortium blockchain network with limited number of members, referred to as *nodes*. Each member is bound to follow the procedure written in smart contract and any kind of violation will lead to heavy penalty or cancellation of their participation. We refer any kind of exchange as *transaction*.
- Structure of block: Each block is divided into two sections: Block Header and Transaction Counter. Block Header further contains block version, Merkle tree root hash, timestamp, nBits, nounce (secret), and parent block hash. Transaction Counter stores the transactions.
- Generation of a block: We assume that a new block is generated after a finite predetermined time once the secret is recovered corresponding to all the transactions done in a fixed time interval.

- Channel: We have assumed our scheme as Evolving-Committee Proactive Secret Sharing Scheme and Channel as Target-Anonymous Channel, discussed in [2].
- Dealer: There will be a team (or committee) of dealers, which is freshly formed for the secret generation and validation of each new block. It will be chosen on the bases of transactions occurred and PoW done, where nodes need to prove the validity of their transaction, using non-interactive zero-knowledge proofs (i.e. without revealing any information about the transactions) and then generate the shares of the secret to be distributed.
- Secret generation and distribution: To generate the secret s_{B_i} for *i*-th block, dealers require the number of transactions in that particular time interval, the number of people involved in the transactions, transaction id's, and the total amount debited and credited. The secret will be distributed among random active nodes.
- Participants: Since, our scheme follows Target-Anonymous Channel, the participants (who receive the secret share) are anonymous. Secret share will be distributed to few active nodes (miners) anonymously and they required to collectively participate, compute the secret, and verify and validate the transactions (called as mining process). If participants were not able to conclude (that the transaction is valid or not) within the predetermined time interval, it would be considered as validated and automatically be added to the block and no further questioning will be allowed.
- Secret recovery: A threshold of 50% is required to set, that is, at least 50% active nodes need to find the secret. Once they recover the secret, they need to verify and validate the transactions.
- Formation of block: Once, the transactions are validated, a new block will be formed and added to the longest available chain, containing all the validated transactions stored in it.

Example 4.1. We will show it with an example.

- 1. Assume that we have 100 participants in our blockchain network and each member is given the identity U_i , $1 \le i \le 100$.
- 2. A new block is generated after every τ_0 minutes, and the active nodes (miners) will be given τ_1 minutes to recover the secret that validates the transactions.
- 3. Let $T_{i,j}$ be the transaction ID of the j^{th} transaction for the i^{th} block.
- 4. Let T_i be the concatenation of all the transaction IDs of the i^{th} block.
- 5. Suppose there are 20 transactions that has taken place during the k^{th} interval $[(k-1)\tau_{\circ}, k\tau_{\circ}]$ and identities involved in these transactions are $U_1, U_2, U_3, \ldots, U_{14}$.
- 6. Then $U_1, U_2, U_3, \dots, U_{14}$ form the committee of dealers and they need to validate the transactions using zero-knowledge proof, before the generation of the secret.
- 7. Once, they all get convinced with all the transactions, they will reveal their transaction IDs and the amount credited or debited from their account. Finally, they generate the secret s_{B_k} for the k^{th} block, where $s_{B_k} = \mathcal{E}(N_{trans}, N_{peop}, T_k, A_{deb}, A_{cred})$ is a m-tuple such that first t bits are message bits and remaining m-t bits are parity bits,

 N_{trans} is number of transactions happened in that particular time interval, N_{peop} is number of people involved in the transactions, T_k is same as defined above, and A_{deb} and A_{cred} represents the total amount debited and credited respectively (note that $A_{deb} = A_{cred}$), and \mathcal{E} is the encryption function which maps 5-tuple to m-tuple.

- 8. Dealers will then submit s_{B_k} to the system and system will run our MSS and distribute the shares to random m participants (active nodes) using Target-Anonymous Channel. Any t nodes can compute the secret and then verify and validate the transactions within the given τ_1 minutes.
- 9. Once, the verification and validation is done, a new block will be added to the chain using all the parameters required for block header and block generation.

4.2 Applications on various sectors

We can effectively apply our scheme on different sectors such as national security, healthcare, supply chain management, decision making process of a company, elections, etc., where a few crucial and confidential information is required to be shared with a group of people in such a way that no adversary will get any information about it. A few of them are mentioned below:

- 1. National security is a serious concern. Even a small attack or information leakage can have major consequences. Thus, we can use this scheme to protect the data. Also, authorities from different departments can communicate and take the decisions accordingly. For example, Nuclear Command Authority (NCA) of India, which is responsible for command, control and operational decisions regarding India's nuclear weapons programme, can interact with the Political Council headed by the Prime Minister of India and an Executive Council headed by the National Security Advisor, to take a decision regarding a nuclear test, in such a way that no outsider will get the information prior to the completion of the test.
- 2. If the board of directors of a *company* takes a crucial decision which can affect the overall growth of the company, then shareholders can verify if the decision taken by the board will add to the future growth of the company or not and they can question it accordingly.
- 3. In healthcare sector, patients can share their medical history (that includes medications, health issues, lab results etc.) with the hospital, termed as health information exchange (HIE) and hospital can further forward it to specialists and other relevant departments within it. Also, it helps in storing the electronic health record (EHR) of the patient.
- 4. To apply our scheme on *supply chain*, three basic entities: suppliers, enterprises, and market dealers can be considered. Enterprise can send their requirement and ask for the quotations from the suppliers in an encrypted form through blockchain platform. Similarly, enterprises can share their product information and quotation with market dealers.

5 Analysis

5.1 Analysis of Multisecret-sharing scheme

Now we show here that our scheme is secure by proving that any t-1 or less participants can't retrieve the secret. If possible, we assume that t-1 participants come together to compute the secret s. For this they initially require to compute the polynomial h(x).

Without loss of generality, we assume $P_1, P_2, \ldots, P_{t-1}$ are t-1 participants and $(a_i, h(a_i))$ are their respective shares. Also, h(x) is a polynomial of degree t-1 with t coefficients. Thus we have a system of t-1 linear equations in t variables;

$$h(a_1) = H_0 + H_1 a_1 + \dots + H_{t-1} a_1^{t-1},$$

$$h(a_2) = H_0 + H_1 a_2 + \dots + H_{t-1} a_2^{t-1},$$

$$\vdots$$

$$h(a_{t-1}) = H_0 + H_1 a_{t-1} + \dots + H_{t-1} a_{t-1}^{t-1}.$$

Therefore,

$$\underbrace{\begin{bmatrix} 1 & a_1 & \dots & a_1^{t-1} \\ 1 & a_2 & \dots & a_2^{t-1} \\ 1 & a_3 & \dots & a_3^{t-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & a_{t-1} & \dots & a_{t-1}^{t-1} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} H_0 \\ H_1 \\ H_2 \\ \vdots \\ H_{t-1} \end{bmatrix}}_{\mathbf{X}} = \underbrace{\begin{bmatrix} h(a_1) \\ h(a_2) \\ h(a_3) \\ \vdots \\ h(a_{t-1}) \end{bmatrix}}_{\mathbf{B}},$$

where A is a $(t-1) \times t$ matrix. Since, all a'_i s are distinct and it is a sub matrix of a Vandermonde matrix of size $t \times t$, implies rank of A is t-1. For every $H_0 \in \mathbb{F}_p$ there exist unique $H_1, H_2, \ldots, H_{t-1}$, which implies there are at least p solutions. Thus, our scheme is secure against the attack made by any t-1 or less participants. Therefore, our scheme is perfect.

5.2 Analysis of the scheme on Blockchain Network

- **Privacy**: Dealers first need to convince each other regarding their valid transactions using zero-knowledge proof and then secret will be generated using the encryption of the transaction details. Moreover, the honesty of the participants will be tested. Thus this scheme maintains privacy.
- Integrity: Data is stored after two-level verification (using our scheme) in blockchain network and once it is recorded in a block, it can't be removed. Also, each block is linked with the previous block hash and any change in the transaction will lead to the change in the hash value of all preceding blocks. Thus, data stored is immutable and permanent.
- Security: In our blockchain network, all nodes will be treated equally and new nodes can join only after signing smart contract and a proper verification by active nodes. Further, they require to prove their honesty before getting any information (shares) of the secret. Also each block is added to the blockchain network only after verifying and validating it by at least t participants. Thus, it will provide security against double spending. Also, our scheme is secured against Finney attack, Race attack, 51% attack, and Sybil attack.

• Scalability: Scalability in blockchain refers to the ability of the platform to expand as per the requirement and support the increasing load of transactions and nodes in the network.

Performance of blockchain network is measured on the basis of average time taken by a transaction to validate. An increase in the number of nodes will lead to an increase in number of transactions, which will affect its performance. Each transaction require space to get stored in block. Moreover, blockchain is decentralized; thus each node is required enough space to store the data, which increases the storage and maintenance cost. Also every node must keep an updated record which will decrease the transmission speed.

To resolve these issues, we have designed our algorithm in such a way that there will be only limited number of nodes. Moreover, secret sharing data can be deleted after the secret gets recovered and block formation process is done.

Also to resolve the storage issue for every node, a few super computers can be installed which store the data in place of each node. It will also protect the network from single point of failure. However, nodes can be given access to that information. We can also use the method of sharding which involves splitting a blockchain into multiple pieces (called shards), and storing them at different places. It helps to manage the storage and cost problem with the increase in the number of transactions.

Moreover, limited number of nodes and limited transactions will enhance the speed of transmission and compacting multiple transactions into an m length secret will also reduce the storage requirement.

Since each transaction holder has already convinced other dealers regarding the validity of the transaction, we have assumed if participants were not able to conclude (that the transaction is valid or not) within a pre-determined time interval, it would be considered as validated and automatically be added to the block. In this way, we can some how reduce the scalability issue in our MSS based blockchain network.

6 Conclusion

In this manuscript, we introduce (t, m)-threshold secret sharing scheme and multisecretsharing scheme with two-level security based on Shamir's SSS using one-way function. Then we generalize the scheme to multi dealer (called as committee of dealers) to efficiently apply it to the blockchain network.

References

- [1] J. Baron, K. E. Defrawy, J. Lampkins and R. Ostrovsky, *Communication-optimal proactive secret sharing for dynamic groups*, In International Conference on Applied Cryptography and Network Security, Springer, Cham, 23-41, 2015
- [2] F. Benhamouda, C. Gentry, S. Gorbunov, S. Halevi, H. Krawczyk, C. Lin, T. Rabin and L. Reyzin, Can a public blockchain keep a secret?, In Theory of Cryptography Conference, Springer, Cham, 260-290, 2020

- [3] G.R. Blakley, Safeguarding Cryptographic Keys, In Proceedings of the 1979 International Workshop on Managing Requirements Knowledge (MARK), New York, NY, USA, 48, 313–317, 1979
- [4] J. Cha, S. K. Singh, T. W. Kim and J. H. Park, Blockchain-empowered cloud architecture based on secret sharing for smart city, Journal of Information Security and Applications, 57, 102686, 2021
- [5] H. Y. Chien, J. K. Jan and Y. M. Tseng, A practical (t, n) multi-secret sharing scheme, IEICE transactions on fundamentals of electronics, communications and computer sciences, 83(12), 2762-2765, 2000
- [6] V. Goyal, A. Kothapalli, E. Masserova, B. Parno, Y. Song, Storing and retrieving secrets on a blockchain, In IACR International Conference on Public-Key Cryptography 2022 Mar 8, Springer, Cham, 252-282, 2022
- [7] A. Herzberg, S. Jarecki, H. Krawczyk and M. Yung, Proactive secret sharing or: How to cope with perpetual leakage, In Annual International Cryptology Conference, Springer, 339-352, 1995
- [8] Y. Kim, R. K. Raman, Y. S. Kim, L. R. Varshney, and N. R. Shanbhag, Efficient local secret sharing for distributed blockchain systems, IEEE Commun Lett, 23(2), 282–285, 2018
- [9] S. K. D. Maram, F. Zhang, L. Wang, A. Low, Y. Zhang, A. Juels and D. Song, CHURP: Dynamic-committee proactive secret sharing, In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, 2369-2386, 2019
- [10] D. Z. Morris, Bitcoin is not just Digital Currency. It's Napster for finance, (link) accessed on September 9, 2022
- [11] S. Nakomoto, A Peer-to-peer Electronic Cash System. White paper, Available at https://bitcoin.org/bitcoin.pdf, 2008
- [12] R. K. Raman and L. R. Varshney, Distributed storage meets secret sharing on the blockchain, 2018 Information Theory and Applications Workshop (ITA), IEEE, 2018
- [13] A. Shamir, How to share a secret, Commun. ACM, 22, 612–613, 1979
- [14] N. Szabo, Smart Contracts, 1994, (link) accessed on September 9, 2022
- [15] T. T. Thwin and S. Vasupongayya, Blockchain based secret-data sharing model for personal health record system, In 2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA), IEEE, 196-201, 2018
- [16] R. Tso, A Study on Secret Sharing Schemes with Dishonest Dealers and Participants, University of Tsukuba, 2004
- [17] T. M. Wong, C. Wang and J. M. Wing, Verifiable secret redistribution for archive systems, In First International IEEE Security in Storage Workshop, 2002, Proceedings, IEEE, 94-105, 2002

- [18] F. Xiong, R. Xiao, W. Ren, R. Zheng and J. Jiang, A key protection scheme based on secret sharing for blockchain-based construction supply chain system, IEEE access, 7, 126773-126786, 2019
- [19] C. C. Yang, T. Y. Chang, and M. S. Hwang, A (t, n) multi-secret sharing scheme, Applied Mathematics and Computation, 151(2), 483-490, 2004
- [20] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, An overview of blockchain technology: Architecture, consensus, and future trends, In 2017 IEEE international congress on big data (BigData congress), IEEE, 557-564, 2017