axioms

MDPI

Article

A Verifiable Multi-Secret Sharing Scheme for Hierarchical Access Structure

Irfan Alam [1], Amal S. Alali [2], Shakir Ali [3,4,*] and Muhammad S. M. Asri [4]

[1] Cyber Security and Digital Forensics Division, SCAI, VIT Bhopal University, Sehore 466114, India; irfanalam@vitbhopal.ac.in
[2] Department of Mathematical Sciences, College of Science, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; asalali@pnu.edu.sa
[3] Department of Mathematics, Faculty of Science, Aligarh Muslim University, Aligarh 202002, India
[4] Institute of Mathematical Sciences, Faculty of Science, University of Malaya, Kuala Lumpur 50603, Malaysia; sufi.maths@gmail.com
* Correspondence: shakir.ali.mm@amu.ac.in

Abstract: Sharing confidential information is a critical concern in today's world. Secret sharing schemes facilitate the sharing of secrets in a way that ensures only authorized participants (shareholders) can access the secret using their allocated shares. Hierarchical secret sharing schemes (HSSSs) build upon Shamir's scheme by organizing participants into different levels based on priority. Within HSSS, participants at each level can reconstruct the secret if a specified number, denoted as the threshold value (t), or more of them are present. Each level has a predetermined threshold value. If the number of participants falls below the threshold at any level, higher-level participants must be involved in reconstructing the secret at lower levels. Our paper proposes schemes that implement hierarchical access structures and enable the sharing of multiple secrets. Additionally, our proposed scheme includes share verification. We have analyzed potential attacks and demonstrated the scheme's resistance against them. Through security analysis and comparison with existing schemes, we highlight the novelty and superiority of our proposed approach, contributing to advancements in secure information-sharing practices.

Keywords: access structure; multi-secret; hierarchy; verification; secret sharing; polynomial

MSC: 94A62; 94A60; 94A17

check for updates

Citation: Alam, I.; Alali, A.S.; Ali, S.; Asri, M.S.M. A Verifiable Multi-Secret Sharing Scheme for Hierarchical Access Structure. *Axioms* **2024**, *13*, 515. https://doi.org/10.3390/axioms13080515

Academic Editor: Cristina Flaut

Received: 29 May 2024
Revised: 6 July 2024
Accepted: 17 July 2024
Published: 30 July 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Information such as encryption keys, missile launch codes, and numbered bank accounts must be highly confidential. Exposure to such sensitive information could be dangerous. Secret sharing schemes provide an efficient way of storing such sensitive and vital information and prevent unauthorized access. Apart from core cryptography, researchers have been using secret-sharing concepts in various applications such as the cloud and IoT. Recently, Gutte and Paraser [1] have used secret sharing for visual cryptography and suggested a weed optimization algorithm for image sharing. Similarly, Wang et al. [2], and Ren et al. [3] have worked on low latency cloud-based indoor localization systems and secure, anonymous data aggregation schemes, respectively. Considerable work has been conducted on this topic in the last few years (cf. [4–12], where further references can be found).

Initially, some core concepts of algebra were used by authors to design secret-sharing schemes. Shamir [13] used polynomials, Blakley [14] used hyperplane geometry and Simmons [15] as well as Asmuth and Bloom [16] used Chinese remainder theorem. In secret sharing schemes, secret S is distributed among n shareholders $P_1, P_1, \ldots, P_n$ in such a way that t shareholders or more than t shareholders can reconstruct the secret but less than t shareholders know nothing about the secret. Such a scheme is known as the (t, n)-threshold scheme. Schemes given by [13,14,16] have several common drawbacks as follows:

1. These are single secret sharing schemes.
2. For every new secret, a new share has to be generated for every participant after the reconstruction of the previous secret
3. Private channels are essential for the communication between dealers and participants and among the participants.
4. These schemes are not capable of identifying the cheater.

To solve the first and the second problem, "multi secret sharing schemes" have been introduced [17–19]. Instead of having a single secret, multiple secrets are shared among the participants in a multi-secret sharing scheme. To include verification and cheater detection, a "Verifiable secret sharing scheme" was proposed in [20,21]. In [22,23], Harn et al. proposed the protected secret sharing scheme to avoid a separate communication channel while exchanging shares among different shareholders. The application of Shamir's scheme is ideal for the condition where all participants (shareholders) play the same role and there is no distribution of shares based on priority or any unique properties. However, employees are categorized based on their work responsibilities in organizations such as multinational companies and educational institutions. Among several structures, the Hierarchical structure is trendy in which every participant has some weight according to his/her role. Shares are distributed or assigned according to their weight. Hierarchical secret sharing schemes are proposed in the literature [15,24–27].

In hierarchical secret sharing schemes, all the participants are divided into m disjoint sets called levels say $l_1, l_2, ...l_m$. The ith level consists of n_i participants with t_i threshold. In the reconstruction of the secret, there may be two situations: In the first situation, the particular level l_i has t_i or more participants on the same level while in the other situation, the number of participants in level l_i is less than t_i. In the second situation, the involvement of the higher level participant is needed.

Let us assume that at a particular level, l_i participants are less in number say r_i, than the t_i. So, $t_i - r_i$ remaining participants are needed from the upper level to reconstruct the secret. In this paper, the level l_i is higher than the level $l_{i+1}, 1 \leq i < m$.

In 2004, Yang et al. [28] proposed a unique multi-secret sharing scheme using a single polynomial, known as the YCH scheme. Low computation and fewer public values are required in the YCH scheme. In the YCH scheme, a single polynomial is used for multiple secrets instead of using separate polynomials for individual secrets. We have included a YCH scheme for each level in our work, making the scheme efficient. We have used two variable one-way functions for verifiability, which verify the dealer and other participants. The following summary highlights the key innovations and benefits of your proposed hierarchical secret-sharing scheme in a concise and structured manner

1. Efficiency through YCH Scheme Integration: Our proposed hierarchical secret sharing scheme leverages the YCH scheme at each level, optimizing computational resources and reducing the number of public values required.
2. Enhanced Security with Cheater Detection: We introduce mechanisms to swiftly identify both malicious dealers and dishonest participants, ensuring the integrity of the secret-sharing process.
3. Dual Functionality of Participant Shares: Participants' shares in our scheme serve dual purposes: facilitating secret reconstruction and enabling verification processes, enhancing overall scheme flexibility.
4. Secure Communication Independent Verification: Utilizing two-variable one-way functions, our scheme eliminates the dependency on secure channels for share transmission, ensuring robustness against interception.

The remainder of the paper is structured as follows: Section 2, reviews relevant literature. Section 3, introduces preliminaries pertinent to our proposed scheme. Section 4, addresses identified problems and motivations. Our proposed scheme is detailed in Section 5. Section 6, covers the security and performance analysis of the scheme. Section 7, provides a comparison with existing schemes. Finally, Section 8, concludes with future research directions.

2. Related Work

The Shamir [13] and Blakley [14] threshold secret sharing schemes are two particular examples of hierarchical secret sharing (HSS) in which all participants have the same privileges. In order to improve the applicability of hierarchical secret sharing, many researchers have focused on specific families of access structures. Blundo et al. [29] have focused on graph-based access structure, Pardo et al. [30] explained the bipartite access structure and Tentu et al. [31] include multipartite access structure, compartmented access structure, and hierarchical access structure.

In 1979, the weighted threshold secret sharing mechanism was suggested by Shamir [13]. However, this approach is inefficient since it assigns multiple shares to each participant equal to its integral weight. Simmons [15] then proposed a multipartite access structure in 1988, defining the compartmented access structure and the hierarchical access structure. Following Simmons, Brickell [32] proposed a strategy for constructing an optimal secret-sharing scheme that takes into account multilevel and compartmented access arrangements. However, the approach is inefficient since nonsingular matrices require exponential operations. The multipartite access structure is defined as the split of all members of a group into subsets, with members of the same subset having the same rights. The compartmented access structure and the hierarchical access structure are two types of multipartite access structures. The conjunctive hierarchical access structure and the disjunctive hierarchical access structure belong to the hierarchical access structure family. In 2009, Lin et al. [33] incorporated some modifications into Shamir's scheme [13] and explained the hierarchical secret sharing (HTSS) scheme with two types of variations. "Multilevel threshold secret sharing (MTSS), and compartmented threshold secret sharing (CTSS)".

Verification of shares is also one of the major concerns in secret sharing schemes. In [20,21] authors have specifically explained the verification of shares in a secret sharing scheme. In 2015, Chanu et al. [34] used Two variable one-way functions to verify the correctness of the received share. Further, in 2017, Basit et al. [35] used Shamir's scheme with the successive application of a one-way function and shifted and key technique to propose Multi-stage Multi-secret sharing for hierarchical Access Structure.

Apart from the above-mentioned papers, the Hierarchical secret sharing (HSSS) scheme is discussed in [22,33–38], but their schemes lack fairness. In 2018, Banerjee et al. [36] proposed Cheating detection and cheater Identification for hierarchical structure but lacks correctness. In 2021, Bisht and Deshmukh [39] proposed work on multi-level secret sharing but lacks fairness and perfectness. Similarly, Yuan et al. [38] used homogenous recurrence relation to propose a new efficient scheme that deals with the multi-secret hierarchical scheme.

This scheme has been proposed to overcome the shortcomings of the above-discussed schemes and incorporate the features in the following ways:

- It supports the hierarchical access structures as discussed in the first paragraph of the current section. It improves and enhances the applicability of the hierarchical access structure.
- It supports weighted threshold secret sharing as discussed in the second paragraph without any exponential operation and extra burden.
- It supports secret sharing with multi-stage, multi-level properties with verification of shares lacking in schemes discussed in the third and fourth paragraphs.
- In the last paragraph of this section, novel schemes are discussed. These schemes do not explain security features such as correctness and forward/backward secrecy.

The proposed schemes offer crucial security features like fairness and correctness, with significantly lower computational and storage costs. All the previously discussed related works have been summarized in Table 1, for the reader's clarity and ease of observation.

Table 1. Related Work Comparison.

Existing Literature	Main Methodology	Positive Aspects	Weaknesses
Basit et al. [35], Chanu et al. [34]	Multi-stage, Multi-level Secret Sharing	Supports complex hierarchical access with verification of shares.	Issues with fairness and perfectness (Bisht [39]); lacks complete security features.
Blundo et al. [29]	Graph-based Access Structure	Provides flexible access control based on graph relationships.	Specific to graph-based scenarios; may not generalize well.
Pardo et al. [30]	Bipartite Access Structure	Simplifies access control into two distinct groups.	Limited to scenarios with clear two-part division.
Tentu et al. [31]	Multipartite Access Structure	Allows for multiple subsets with varied access rights.	Complexity increases with the number of subsets.
Shamir et al. [13]	Weighted Threshold Secret Sharing	Incorporates weighted shares for different participant capabilities.	Inefficient due to multiple shares per participant.
Shamir et al. [13], Blakley et al. [14]	Hierarchical Secret Sharing	Supports hierarchical access structures; applicable to groups with uniform privileges.	Lacks fairness in hierarchical structure (Banerjee [36]); inefficient with non-singular matrices (Brickell [32]).
Banerjee et al. [36], Bisht et al. [39]	Novel Schemes	Introduces efficient schemes with homogenous recurrence relations.	Often lacks security features like correctness and backward secrecy.

3. Preliminaries/Foundations

This section describes some of the preliminaries required to design our scheme.

3.1. Polynomial Interpolation

The use of polynomials in designing secret-sharing protocols is one of the easiest and most popular methods [40–42]. The following two points are very common regarding polynomials:

- Polynomial can be utilized to plot a curve using the points that satisfy the same polynomial.
- We can determine polynomials of a fixed degree by knowing the points that satisfy the polynomials. The number of such points must be at least one greater than the degree of polynomial.

There are various interpolation techniques in the literature such as Linear Interpolation, Polynomial Interpolation (Lagrange and Newton), Spline Interpolation (Natural Cubic Spline, Clamped Cubic Spline,), Piecewise Linear Interpolation, Inverse Distance Weighting (IDW) and Kriging [43–45]. Lagrange Interpolation is one of the techniques and it is used in Shamir's secret sharing scheme [46–48].

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

In secret sharing, the principle objective is to partition the secret S into n pieces $s_1, s_2, \ldots, s_n$ such that:

1. Learning of t or more s_i pieces makes S uniquely determined.
2. Learning of any $t-1$ or fewer s_i pieces leaves S totally unpredictable.

This scheme depends on polynomial interpolation. The fundamental idea of Adi Shamir's threshold scheme can be understood by a simple example:

1. At least 2 points are necessary to draw a line. (i.e., one point is not sufficient)
2. At least 3 points are necessary to draw a parabola (i.e., less than 3 points are not sufficient)
3. Similarly, it takes at least 't' points to draw a polynomial of degree '$t-1$' (i.e., less than t points are not sufficient).

Following are the main phases of (t, n) threshold scheme, where n is the total number of shareholders and t is the minimum number of shareholders necessary to reconstruct the secret S.

Distribution phase:

1. Select a prime number Q

2. Randomly select a function $g(x) = b_0 + b_1x + b_2x^2 + \cdots + b_{t-1}x^{t-1}$
3. Compute (i, g(i)) corresponding to the ith shareholder, $i = 1, 2, \ldots, n$.
4. These points (i, g(i)) are distributed securely to n shareholders/participants

Reconstruction phase:

1. Compute Lagrange's interpolating polynomial using t shares

$$g(x) = \sum_{i=1}^{t} g(i) \prod_{j=1, j\neq i}^{t} \frac{x - x_j}{x_i - x_j} mod\ Q$$

2. In this way, we obtain the polynomial in the form

$$g(x) = b_0 + b_1x + b_2x^2 + \cdots + b_{t-1}x^{t-1},$$

where b_0 = S is the secret.

3.3. Hierarchical Access Structure

Access Structure (Γ) Recovery of secrets is authorized for some groups of people and it is unauthorized for another group of people. Those sets that are authorized are known as access structures. **Adversary Structure ($\overline{\Gamma}$)** The set of all non-authorized sets that do not have any information related to the secret is said to be an adversary structure.

In (t, n) threshold access structure any set of t or more participants out of n is said to be an authorized set and any set less than t in number is said to be a non-authorized set. Let λ be the set of n participants. In Set builder form a (t, n) threshold access structure and the corresponding adversary structures are as follows:

$$\Gamma = \{X \in 2^{\lambda} : |X| \geq t\}$$

and

$$\overline{\Gamma} = \{X \in 2^{\lambda} : |X| < t\}$$

respectively.

In 2006, Herranz et al. [49] explained importance of multipartite structure. According to Herranz "In multipartite structure the set of players is divided into K disjoint classes, and all players in each class play exactly the same role within the access structure. These access structures can make a lot of sense in real life applications, where persons or machines are divided into different groups according to their position in a company, their responsibilities, their computational resources or their probability of being corrupted by an attacker ".

A multipartite access structure splits the set of participants in λ into m disjoint sets $l_1, l_2 \cdots, l_m$ called levels and all participants in each level play exactly the same role inside the particular access structure.

3.4. Overview of YCH Scheme

Initialization phase

In this scheme following notations are used:

1. (t, n)-scheme, where t is for the threshold and n is for the number of participants.
2. $B_1, B_2, \ldots, B_k$, denotes the k secrets to be shared.
3. n secret shadows $s_0, s_1, s_2, \ldots, s_n$ are randomly chosen by the dealer and distributed to the participants through a secure channel.
4. A random value 'r' is chosen.
5. A 2-variable 1-way function h(r,s_i) , $i = 1, 2, \ldots, n$. is chosen.

Construction phase

1. $k \leq t$ (Number of secrets is less than the threshold)
   (a) A prime number 'Q' is chosen by the dealer.
   (b) The dealer choose a polynomial $f(x)$mod Q. Degree of polynomial is $(t - 1)$ where,

(c) $B_1, B_2, \ldots, B_k$ are the secrets to be shared and $b_1, b_2, b_3, \ldots b_{t-k}$ are random numbers.
(d) $f(x) = B_1 + B_2x+, \ldots, +B_kx^{k-1} + b_1x^k + b_2x^{k+1} + \cdots + b_{t-k}x^{t-1} mod\ Q$
(e) For every ith participant the dealer computes $g_i = f(h(r, s_i))$ mod Q
(f) Publish $(r, g_1, g_2, \ldots, g_n)$.

2. $k > t$ (Number of secrets is greater than the threshold)

(a) A prime number Q and a polynomial $f(x) mod\ Q$ are chosen by the dealer. The degree of the polynomial is $(k-1)$. Where $B_1, B_2, \ldots, B_k$ are the secrets.

$$f(x) = B_1 + B_2x + \cdots + B_kx^{k-1} mod\ Q$$

(b) For i = 1 to n. $g_i = f(h(r, s_i)) mod\ Q$ is computed.
(c) For $i = 1\ to\ n$. $f(i) mod\ Q$ is computed
(d) calculated values like $r, g_1, g_2, \ldots, g_n$ are publicly published.
(e) $f(1), f(2), \ldots, f(k-t)$ are also published in public.

Recovery phase
In order to recover the secrets $B_1, B_2, \ldots, B_k$,

1. Each participant uses his/her share to compute $h(r, s_i)$ (for i = 1 to t)
2. The polynomial $f(x)$ is determined as follows:
3. $k \leq t$ (Number of secrets is less than the threshold)

$$\begin{aligned} f(x) &= \sum_{i=1}^{t} g_i \prod_{j=1, j\neq i}^{t} \frac{x - h(r, s_j)}{h(r, s_i) - h(r, s_j)} mod\ Q \\ &= B_1 + B_2x + \cdots + B_kx^{k-1} + a_1x^k + a_2x^{k+1} + \cdots + a_{t-k}x^{t-1} mod\ Q. \end{aligned}$$

4. For $k > t$ (Number of secrets is greater than the threshold)

$$\begin{aligned} f(x) &= \sum_{i=1}^{k} g_i \prod_{j=1, j\neq i}^{k} \frac{x - h(r, s_j)}{h(r, s_i) - h(r, s_j)} + \sum_{i=1}^{k-t} f(i) \prod_{j=1, j\neq i}^{k-t} \frac{x - j}{i - j} mod\ Q \\ &= B_1 + B_2x + \cdots + B_kx^{k-1} mod\ Q. \end{aligned}$$

5. From the above equations, we get the secrets

$$B_1, B_2, \ldots, B_k$$

3.5. 2-Variable 1-Way Function

Definition: A 2-variable, 1-way function $h(r, s)$ is a function that maps a random value r and a share s onto a bit string of fixed length.

Properties: It contains the following properties:

1. when r and s are given $h(r, s)$ is easily computable. But for a given s and $h(r, s)$, it is very difficult to compute r.
2. It is hard to compute $h(r, s)$ when there is no knowledge of s.
3. For the given s, it is hard to find two different values r_1 and r_2 that satisfy the situation $h(r_1, s) = h(r_2, s)$.
4. It is tough to compute s, for the given r and $h(r, s)$.
5. If we have pairs of r and $h(r, s)$, it is difficult to find $h(r', s) = h(r, s)$ for which $r' \neq r$.

4. Identification of Problem and Motivation

In all the existing schemes, a separate polynomial is taken corresponding to each secret which results in overhead of public values and calculations. Verification in a hierarchical system is included in papers like [36] with some limitations. In the present work, we try to reduce the limitations in the verification phase.

Contribution

In our study, we propose a "multi-secret sharing scheme for hierarchical access structures" utilizing the YCH scheme. This approach offers several advantages:

- Efficient Parallel Reconstruction: The YCH scheme allows for parallel reconstruction of multiple secrets, enhancing efficiency in scenarios requiring simultaneous access to different shared secrets.
- Dynamic Distribution of Secrets: Our scheme supports dynamic determination of the number of secrets to be distributed, providing flexibility in managing shared information.
- Optimized Resource Utilization: It minimizes storage requirements and computing time by utilizing fewer public values, making it more efficient compared to traditional schemes.

Additionally, employing a two-variable one-way function provides the following benefits:

- Cheater Identification: Any participant can identify dishonest behavior, whether from the dealer or other participants, ensuring the integrity of the sharing process.
- Secure Communication Elimination: There is no dependency on secure channels between the dealer and participants for share transmission, simplifying implementation and reducing overhead.
- Invalid Share Detection: The scheme includes mechanisms to detect and reject invalid shares, enhancing overall security against fraudulent activities.

This comprehensive approach not only enhances the efficiency and flexibility of hierarchical secret sharing but also strengthens security measures, making it suitable for diverse applications requiring robust information protection and sharing capabilities

5. Proposed Scheme

5.1. Overview

All participants are classified into m levels. Each level has a fixed (t, n) pair where t is the threshold out of n participants. The dealer chooses pseudo share s_i^l for the ith participant at level l. Pseudo-share is distributed through a secure channel. $B_1, B_2, \ldots, B_k$ are the secrets. Using the YCH scheme, the dealer computes the actual share (d_i^l) for each participant. public share (Z_i^l) is calculated on the addition of actual shares in the one-way function.

In the Reconstruction phase, the actual share of the participants is calculated by subtracting the one-way function from the public share. This actual share is used in the Lagrange interpolation polynomial and generates the polynomial having coefficients as secrets. If the share's number at a particular level is less than its threshold value, then the upper-level shareholder provides his share to reconstruct the secret.

5.2. Initialization

1. Number of participants is n.
2. The number of levels is m. They are $l_1, l_2, \ldots, l_m$.
3. Each level is associated with a (t_j, n_j), $j \in [1, m]$ access structure.
4. Dealer chooses n shares s_i^j, $i \in [1, n_j]$, $j \in [1, m]$.

5.3. Distribution

At each level l there may be two situations:

- The number of secrets k is less than the threshold
- A number of secrets k is more than the threshold.

1. $k \leq t_j$ (number of secret is less than or equal to t_j)
   (a) A prime number Q is chosen by the dealer.
   (b) The dealer constructs polynomial $f(x)$mod Q. The degree of the polynomial is $(t_j - 1)$. Let,

$$f(x) = B_1 + \cdots + B_k x^{k-1} + b_1 x^k + \cdots + b_{t-k} x^{t-1} mod\ Q,$$

where $B_1, B_2, \ldots, B_k$, are the secrets,

(c) $b_1, b_2, b_3, \ldots, b_{t_j-k}$,$j \in [1, m]$ are randomly chosen numbers.

(d) For the ith participant the dealer computes $g_i = f(h(r, s_i))mod\ Q$, $i \in [1, n]$.

(e) Publish $(r, g_1, g_2, \ldots, g_n)$.

2. $k > t_j$ (Number of secrets is greater than t_j).

(a) A prime number Q is chosen.

(b) Dealer constructs a polynomial $f(x)mod\ Q$ of degree $(t_j - 1)$. Let

$$f(x) = B_1 + B_2x + \cdots + B_kx^{k-1} mod\ Q,$$

where $B_1, B_2, \ldots, B_k$ are the secrets.

(c) For $i = 1, 2, 3, \ldots, n$. $g_i = f(h(r, s_i))mod\ Q$ is Computed.

(d) For $i = 1$ *to* n. $f(i)mod\ Q$ is Computed

(e) calculated values like $r, g_1, g_2, \ldots, g_n$ are made public

(f) $f(1), f(2), \ldots, f(k - t)$ are published in public.

For both cases, the dealer performs the following calculations:

1. Calculate actual share (d_i^l) and pseudo share(Z_i^l) for the ith participant of level l using the following formulas:

$$d_i^l = f(ID_i^l) \ \ and \ \ Z_i^l = d_i^l + g(s_i^l)$$

where g is a one-way function in which s_i^l denotes the share of ith participant of level l

2. Calculate the actual share ($d_i^{l_u}$) and pseudo share ($Z_i^{l_u}$) of the ith participant of upper level u using the formula:

$$d_i^{l_u} = f(ID_i^{l_u}) \ \ and \ \ Z_i^{l_u} = d_i^{l_u} + g(s_i^{l_u})$$

where $ID_i^{l_u}$ is the identifier for ith element of uth level.

3. s_i^l and $s_i^{l_u}$(if needed) are distributed to each participant using a secure channel.

4. All $Z_i^{l_r}$, r values are published.

5.4. Reconstruction

Each participant computes the actual share of other participants involved in the reconstruction by using the formula

$$d_i^l = Z_i^l - g(s_i^l)$$

and then the following two cases are considered.

Case 1

A particular level has a sufficient number of participants, i.e., greater or equal to the threshold, then participants of the same level exchange their pseudo share and use the following formula:

1. for $k \leq t$

$$\begin{aligned} h(x) &= \sum_{i=1}^{n} d_i \prod_{j=1, j\neq i}^{n} \frac{x - ID_j}{ID_i - ID_j} mod\ Q \\ &= B_1 + B_2x + \cdots + B_kx^{k-1} + a_1x^k + a_2x^{k+1} + \cdots + a_{t-k}x^{t-1} mod\ Q. \end{aligned}$$

2. for $k > t$

$$h(x) = \sum_{i=1}^{n} d_i \prod_{j=1, j\neq i}^{n} \frac{x - ID_j}{ID_i - ID_j} + \sum_{i=1}^{k-t} f(i) \prod_{j=1, j\neq i}^{k-t} \frac{x-j}{i-j} mod\ Q$$
$$= B_1 + B_2 x + ... + B_k x^{k-1} mod\ Z.$$

Case 2

Particular levels have an insufficient number of thresholds, then the share of the upper-level participant for this level is used in the above formula. Thus, we obtain the secret $B_1, B_2, B_3, \ldots, B_K$.

5.5. Verification

The following steps are involved in the verification phase:

1. Pseudo share s_i^l is distributed to participants securely by the dealer.
2. Each ith participant uses his actual share (d_i^l) in a two-variable one-way function with random variable r. Let that two-way variable function be $q_i(r, d_i^l)$.
3. Calculated q_i values made public.
4. Public share Z_i^l is also published.

Now at the time of the exchange of shares:

5. Each participant computes the actual share of each participant

$$d_i^l = Z_i^l - g(s_i^l).$$

6. Using that actual share, $q_i(r, d_i^l)$ is calculated for i_{th} participant.
7. If $q_i(r, d_i^l)$ is equal to the already public value of q_i then the participant share is valid; otherwise, the actual share of the participant is not valid.
8. In a similar way, an individual participant will be able to check the legitimacy of his/her share given by the dealer.

5.6. Example

We can understand the proposed scheme more clearly by following examples with small parameters.

Let us consider the hierarchical system with two levels l_1 and l_2. The upper level is l_1 and the lower is l_2. The dealer has two secrets, i.e., $B_1 = 2$ and $B_2 = 3$. Identifiers $ID_1^{l_1} = 1$, $ID_2^{l_1} = 2$, $ID_3^{l_1} = 3$ for the level l_1 and $ID_1^{l_2} = 4$, $ID_2^{l_2} = 5$, $ID_3^{l_2} = 6$ for the level l_2.

The dealer chooses shares $s_1^1 = 10$, $s_2^1 = 11$, $s_3^1 = 12$ for the level-1 and $s_1^2 = 13$, $s_2^2 = 14$, $s_3^2 = 15$ for the level-2.

5.6.1. Distribution

For the sake of easiness, we perform calculations on the second level.

$$h(x) = 2 + 3x + 4x^2 mod\ 23$$

calculate actual share for the ith participants for the level-2 by using formula: $d_i^{l_r} = \mathrm{h}(ID_i^{l_r})$ are

$$d_1^{l_2} = 9,\ d_2^{l_1} = 2,\ d_3^{l_1} = 3.$$

The actual share of the ith participants of the upper level for the level-2

$$d_1^{l_{2,1}} = h(ID_1^{l_1}) = 9.$$

Similarly, we obtain $d_2^{l_{2,1}} = 1$ and $d_3^{l_{2,1}} = 1$.

Choose one-way function $g = 2^{s_i^{l_r}} \bmod 23$.

Compute the pseudo share of the *i*th participants of the level R.

$$Z_i^{l_r} = d_i^{l_r} + g(s_i^{l_r}).$$

Therefore, we obtain $Z_1^{l_2} = 13$, $Z_2^{l_2} = 10$, $Z_3^{l_2} = 19$.
Similarly, the pseudo share of the participants of the upper level for level 2 are

$$Z_1^{l_{2,1}} = 21,\ Z_2^{l_{2,1}} = 2,\ Z_3^{l_{2,1}} = 3.$$

Publish all pseudo share and distribute $S_i^{l_r}$ to every participant through a secure channel.

5.6.2. Reconstruction

In the reconstruction phase, each participant has a pseudosphere and the share $s_i^{l_r}$ for the particular level actual share is computed by using

$$d_i^{l_r} = Z_i^{l_r} \text{-} g(s_i^{l_r}).$$

Hence, we have

$$d_1^{l_2} = 9,\ d_2^{l_2} = 2,\ d_3^{l_2} = 16.$$

Similarly, the actual share of the upper-level participants for level 2 is as follows:

$$d_1^{l_1} = 9,\ d_2^{l_1} = 1,\ d_3^{l_3} = 1.$$

Now, there is the possibility of two cases:

- A particular level (here level 2) has a sufficient number of participants.
- A particular level has fewer participants than the upper-level participant and takes part in the reconstruction of the secret.

Case-1

$$\begin{aligned} h(x) &= \sum_{i=1}^{t} d_i^l \prod_{j=1, j\neq i}^{t} \frac{x - ID_j}{ID_i - ID_j} mod\ Q \\ &= B_1 + B_2 x + \cdots + B_k x^{k-1} + a_1^i x^k + a_2^i x^{k+1} + a_{t-k}^i x^{t-1} mod\ Q \\ &= 9 * \frac{(x-5)(x-6)}{(4-5)(4-6)} + 2 * \frac{(x-4)(x-6)}{(5-4)(5-6)} + 3 * \frac{(x-4)(x-5)}{(6-4)(6-5)}. \end{aligned}$$

That is,

$$h(x) = 2 + 3x + 4x^2.$$ So the secrets are 2, and 3.

Case-2

The participants in level 2 are one less than the threshold. Therefore, one of the upper-level participants takes part in the reconstruction. Hence, the pair of (ID, actual share) are: (4, 9) (5, 2) (1, 9) applying the same formula as in Case 1.

$$h(x) = 9 * \frac{(x-5)(x-1)}{(4-5)(4-1)} + 2 * \frac{(x-4)(x-1)}{(5-4)(5-1)} + 3 * \frac{(x-4)(x-5)}{(1-4)(1-5)}.$$

Therefore, $h(x) = 2 + 3x + 4x^2$. Hence, the secrets are 2 and 3.

6. Security and Performance Analysis

While choosing a scheme, we must be clear about the capabilities of the adversary, along with an explanation of the security properties it supports. There must be a strong analysis of the computational and communication costs. In this section, we have explained all these concepts in detail.

6.1. Adversary Model

There must be an adversary model against which the scheme is safe. The most common adversarial model for analyzing security protocols was presented by Dolev and Yao [50] in 1983. Apart from this, we have considered two types of adversaries:

- **Insider Adversary**, these are legitimate shareholders who acquired shares from the dealer.
- **Outsider Adversary**, the external adversary is an attacker who does not own any of the dealer's shares but may try to gain unauthorized access.

In general, a dealer is considered as trustworthy as in [36], but here we have considered the worst situation that the dealer is committing fraud by providing fake shares to the participants.

6.2. Security Analysis

Any scheme must pass through formal and informal analysis to verify its applicability in the current scenario. In the proposed scheme, we have used the Random Oracle Modal (RoM) for formal analysis. In informal analysis, we have proved our scheme safe from several attacks.

6.2.1. Formal Security Analysis (Random Oracle Model)

A cryptographic hash function H is treated as a really random function by the random-oracle model. The random-oracle model more particularly hypothesizes the existence of a public, random function H that can only be evaluated by "querying" an oracle, which can be thought of as a "black box", that returns $H(x)$ when given input x. A formal approach that can be used to create and verify cryptographic methods is provided by the random-oracle model. In 2014, Herranz et al. [51] provided the formal definition of security for MSS in the random oracle model. Moreover, they proposed an MSS formally proved its computational security in ROM. As far as we know, that is an MSS's first formal security analysis. Security analysis of the multi-secret sharing scheme has been performed by [52]. In this section, we prove the computational security of the proposed scheme Ω assuming that the hash function H behaves as a random oracle.

Before we proceed with the proof, the following are the assumptions:

- We assume the proposed scheme as the set of tuples, $\Omega = (Int, dist, Rec)$ where Int stands for the initialization/setup phase, dist stands for distribution and Rec is for the recovery phase.
- PP (Public parameters) = (p, H, ρ, t)
  1. p: prime number p > n such that p is at least λ bits long
  2. H: a hash function, $H : \mathbb{Z}_p^* \to \mathbb{Z}$
  3. ρ: Set of participants
  4. t: Threshold
- There is an adversary A_1 containing a set of participants ρ and threshold value t.

Theorem 1. *For an adversary λ, we have*

$$\left| pr\left[GMS_{A_1}^{SA}(\lambda) = 1\right] - \frac{1}{2} \right| \leq \left| \frac{q_H^n}{\lambda^{\lambda+2}} + 0(\frac{q_H^n}{2^{2\lambda+1}}) \right|$$

here adversary makes at most qH queries to the random oracle for H against the GMS and Ω.

Proof. For the proof we follow the following steps:

Step 1 We act as the challenger of the security game $G1$ described in [52].

Step 2 We pass public parameters to the initialization algorithm and send the result to the adversary.

Step 3 $A1$ broadcast $B \epsilon \rho$, we choose $sh_i \epsilon Z_p$, $\forall p_i \epsilon \rho$

Step 4 After masking hash query x to the random oracle H, if $x \epsilon \{sh_i\} p_i \epsilon B$ then abort the game, otherwise proceed for the next step.

Step 5 A random value is $k \epsilon Z_p$ is chosen and sent to the adversary. At the same time, (x, k) is saved in the table.

Step 6 Two global secrets

$$S^0 = (s_1^0, s_2^0, \ldots, s_l^0)$$
$$\neq (s_1^1, s_1^2, \ldots, s_l^1) = S^1$$

are broadcast by adversary A_1.

Step 7 We choose a random value $r \epsilon Z_p^*$ such that $r \neq sh_i$ for $P_i \epsilon \rho$ and the adversary does not query $r, 2r, \ldots, lr$ to the random oracle H.

Step 8 A random polynomial $f(x)$ is chosen where, $f(0) = r$ and compute $h_i = H(sh_i)$ and $r_i = f(i) - h_i mod p$ and again store (sh_i, h_i) in the hash table. Similarly, choose l no. of random values $(k_1, \ldots, k_l) \epsilon Z_p$ and store $(r, k_1), (2r, k_2), \ldots, (lr, k_l)$ in the hash table.

Step 9 Choose a random bit $b \epsilon \{0, 1\}$ and compute $y_j = k_j - s_b^j mod p \forall 1 \leq j \leq l$ and give shares of corrupted players $(i, sh_i)_{p_i \epsilon B}$. along with this public output $Out_{pub} = \{r_1, r_2, \ldots, r_n, y_1, \ldots, y_l\}$ which are also shared.

Step 10 after continuous query to $H(.)$, adversary A_1, outputs a bit b' which is defined in the following equation

$$o/p = \begin{cases} 1, \text{if } b = b' \\ 0, \text{if } b \neq b'. \end{cases} \quad (1)$$

From the above steps, we conclude that A_1 is not allowed to query sh_i such that $p_i \epsilon B$ and $1 \geq j \leq l$. □

6.2.2. Informal Security Analysis

Any proposed protocol must be passed through the following goals:

1. **Correctness**: In the verification phase, each participant computes the actual share of other participants $d_i^l = Z_i^l - g(s_i^l)$ using that actual share, $q_i(r, d_i^l)$ is calculated for ith participant. If $q_i(r, d_i^l)$ is equal to the already public value of q_i then the participant share is valid otherwise the actual share of the participant is not valid.
2. **Forward secrecy**: keys like $q_i(r, d_i^l)$ can only be computed or stored by members of the closed communication group; if a member leaves the group, the departing member will be unable to access the content of future conversations.
3. **Perfectness**: We use Shamir's (t, n) secret sharing scheme for share distribution at each level in the proposed scheme. It is well known that fewer than t participants in Shamir's (t, n) secret-sharing scheme cannot reconstruct the secret. Hence, our scheme is also perfect.
4. **Fairness of secret sharing** One desirable quality in secret sharing is Fairness, which indicates that if one member obtains the secret, the other participants are not harmed. Halpern and Teague [53] were the first to offer rational cryptographic protocols in 2004. They pointed out that any method for reassembling secrets with a well-known upper constraint on the running duration is unstable and that parties will not submit anything in the final round since they have no reason to do so because the other participant does. Regrettably, earlier secret-sharing systems necessitated numerous rounds with high overheads. Zhang et al. [26,27] explain the scheme. Fairness but leave out the access mechanism. To summarize, existing fairness schemes necessitate a trusted third party or many rounds of communication. The proposed scheme provides Fairness without the dependency on a third party and extra overheads in communication.
5. **Freshness of keys**: In the proposed scheme, all the exchangeable values depend upon the random bi-variate polynomial. It makes pseudo-share, and shares are always fresh. Hence, it is impossible to impersonate a member by recording a previously used key.
6. **Eavesdropping attack** In the proposed scheme, each ith participant uses his actual share (d_i^l) in a two-variable one-way function with random variable r which makes it

independent of public parameters. Therefore, an adversary will not be able to know any secret information from communication parameters. So, the proposed scheme is safe from eavesdropping activities.

7. **Cheating identification** In the proposed scheme, an individual participant can check the legitimacy of his/her share given by the dealer. Thus, the proposed scheme provides cheating identification.
8. **Verifiability** In the reconstruction phase, participants can verify shares received from other participants and dealers. On the other hand, a dealer can also verify the participant's shares. Thus, the proposed scheme provides verifiability.
9. **Unconditional security** While exploring the security analysis, both types of adversaries are considered with their capacity to the full extent.

Recently, Li et al. [54] have used secret sharing with blockchain for the Internet of Medical Things (IoMT) and proposed a more energy-efficient protocol that lacks a hierarchy.

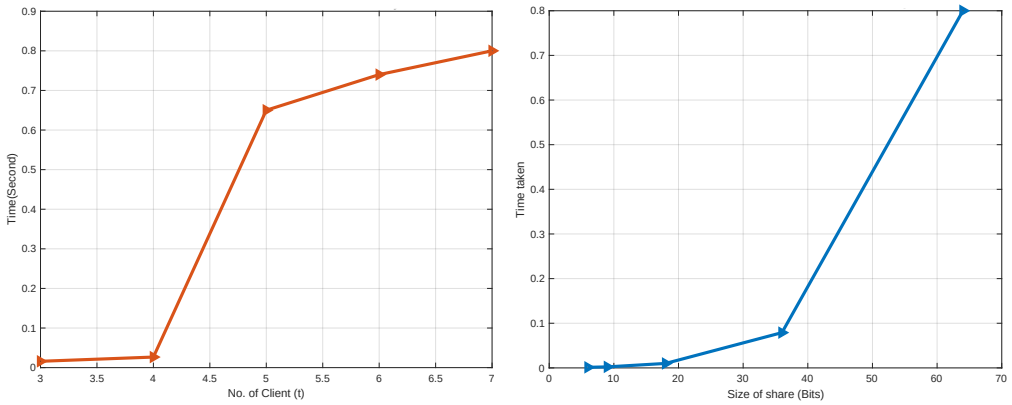
6.3. Performance Analysis

6.3.1. Evaluation of Stealth Share

In this subsection, we analyze the time required for the entire process from share generation to secret reconstruction, examining how share size and the number of participants influence it. Figure 1, provides two types of analysis based on participant count and share size. Part (a) illustrates the relationship between the number of clients (t) and the time (in seconds) taken for secret creation and reconstruction. Initially, with a small number of clients, the time remains just above 0 s, indicating minimal impact. However, as the number of clients increases, the time escalates in a step-like manner, approaching just below 1 s with seven clients. Conversely, part (b) explores the impact of share size (in bits) on the time required for creation and reconstruction. Here, the time remains consistently close to zero, suggesting that a share size of up to 40 bits does not significantly affect the process. We have used a Python library along with resources available at [55] for the evaluation of our proposed scheme as well as existing schemes. We perform the proposed scheme and other similar schemes on a Windows 10 laptop with AMD Ryzen 7, 3.0 GHz and 16 GB RAM. Furthermore, mathematically we see that each participant stores a univariate polynomial of $\mathbb{P}$, resulting in a storage requirement of $t \log \mathbb{P}$ bits per participant. This polynomial-based modulus is notably smaller than the moduli used in public-key systems [56]. The threshold here is t. To ensure collusion-free authentication, each participant must concurrently manage t + 1 univariate polynomials. Consequently, the overall storage cost is considerably lower compared to symmetric key-based schemes.

6.3.2. Computational Cost

In the proposed scheme, each member must compute pairwise shared keys with others in the verification phase. According to Horner's rule [57], t and h computations are required for each univariate polynomial of $t-1$ and $h-1$ degree. In the proposed protocol, calculation reduces to Shamir's secret sharing scheme. According to [58] computational time for eight people is 0.0039 s. There is a slight change in computation time as we increase the number of people; Basit et al. [35] use an extra polynomial, and Tentu et al. [31] use modular exponentiation, which increases computational cost. The computational cost for the same no. of persons is 0.2439 s and 80.003 s of [35] and [31], respectively. Similarly, the computational cost for eight people in [27] is 0.23 s. We conclude that the computation cost is meager compared to the other existing schemes. For more clarity, we have presented a graphical comparison of schemes in terms of computational time and security features it supports in Figure 2.



(**a**) Time with respect to no.of client (**b**) Time with respect to size of share

Figure 1. Time in stealth share with respect to no.of client and size of share.

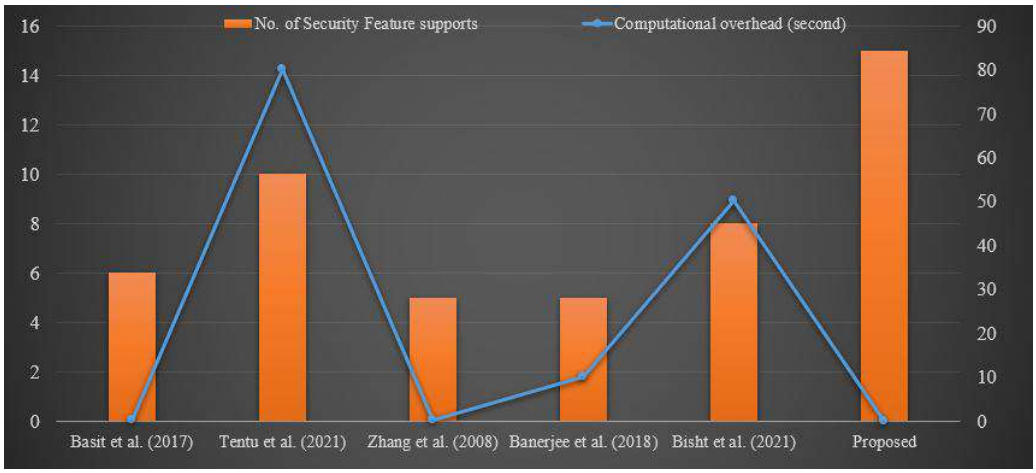


Figure 2. Comparison of computational cost and security features supported by Basit et al. [35], Tentu et al. [31], Zhang et al.[27], Banerjee et al. [36], Bisht et al. [39] and Propose protocol.

7. Comparison with Existing Schemes

The proposed scheme uses the YCH scheme, while all other schemes for hierarchical access structures use simply Shamir's scheme. In the proposed scheme "Multiple-secrets" are shared among a group of people for different levels. Our scheme provides a mechanism for verifying the shares, which is also new for hierarchical access structures. This is more efficient than other schemes. Basit et al. [35] used Shamir's scheme with the successive application of one-way function and shift key technique to propose Multi-stage, Multi-secret sharing for hierarchical Access Structure. Apart from the above papers, the Hierarchical secret sharing (HSSS) scheme is discussed in [33,36,39] but their schemes lack fairness. Banerjee et al. [36] proposed Cheating detection and cheater Identification for hierarchical structure but lacks correctness. Bisht and Deshmukh [39] proposed work on multi-level secret sharing but lacks fairness and perfectness. Similarly, Yuan et al. [28] used homogenous recurrence relation to propose a new efficient scheme that deals with a multi-secret hierarchical scheme. It is a novel work but lacks an explanation about correctness and forward/backward secrecy. Our scheme also avoids the problem of Chen et al. scheme's [59] which requires checking the non-singularity of multiple matrices. Furthermore, during the entire scheme, each participant keeps only one share, which is as long as the secret, indicating that our method is optimum. Furthermore, despite requiring more public value, our scheme can simultaneously disclose many secrets. In Table 1, we can see the importance of the present scheme in comparison with other schemes.

In our proposed scheme, we have used a one-way function, which restricts the outsider Adversary from stealing or breaching information about the secret. To deal with the situation of Case-1 and Case-2, two-variable one-way functions are used. Using a two-variable one-way function, every participant can verify the share received from the dealer and other participants. Our scheme also satisfies the security goals like correctness and secrecy. In Table 2, we have compared the recent scheme for multi-secret sharing. From there, we conclude that our scheme can provide many security properties in the case of IoT authentication. Furthermore, in Figure 2, we can see that computational overhead (0.23 s) is significantly less than others and security features (almost 20) are more in numbers supported by the proposed scheme.

Table 2. Comparison in tabular form.

Comparison (Security) among the Existing Scheme						
Property	**Basit et al. [35]**	**Banerjee et al. [36]**	**Zhang et al. [26]**	**Tentu et al. [31]**	**Chen et al. [59]**	**Proposed Scheme**
Hierarchy	Yes	Yes	No	Yes	Yes	yes
Reusable	Yes	Yes	Yes	No	No	Yes
Multi-secret	Multi	Multi	Multi	Single	Single	Multi
Based on	LPI	LPI	YCH	LPI	Polynomial	YCH
Ideal and Perfect	Yes	Yes	Yes	Yes	Yes	Yes
Cheating detection	partially	Conditional	Complete	Partially	Complete	Compete
Correctness	No	No	No	No	yes	yes
Forward secrecy	No	No	No	No	No	yes
Fairness	No	No	Yes	No	yes	yes

8. Conclusions

In the present work, we categorized participants into different levels with varying thresholds, aligning with a hierarchical structure. Subsequently, we proposed a hierarchical multi-secret sharing scheme based on the YCH scheme. The YCH scheme is highly efficient, and the inclusion of a one-way function enhances its security, making it unconditionally secure. The reusability of shares eliminates the need for frequent refreshment of shares for future communication. In addition to being reusable, multi-secret, and hierarchical, our scheme includes a valuable feature: the verification of shares. Specific participants can verify the shares they receive from other participants or the dealer. This verification feature makes our proposed scheme particularly useful for robust security solutions. Comparisons with existing schemes demonstrate its superiority in preserving security. Our scheme not only improves operational efficiency but also provides a scalable solution adaptable to various access structures. Future work could explore further optimizations and applications in complex, multi-stage security environments.

Author Contributions: I.A. conceptualized the study, developed the methodology, conducted validation, performed formal analysis, participated in investigation, curated data, and contributed to visualization. A.S.A. contributed to methodology development, software implementation, validation, formal analysis, investigation, data curation, and visualization. S.A., as the corresponding author, contributed to conceptualization, methodology development, software implementation, validation, formal analysis, investigation, data curation, and provided supervision throughout the study. M.S.M.A. contributed to methodology development, validation, investigation, and data curation. Together, they ensured comprehensive coverage of the research process from conception to final analysis and presentation of results. All authors have read and agreed to the published version of the manuscript.

Funding: This study was carried out with financial support from Princess Nourah Bint Abdulrahman University (PNU), Riyadh, Saudi Arabia. The authors extend their appreciation to Princess Nourah Bint Abdulrahman University (PNU), Riyadh, Saudi Arabia for funding this research under Researchers Supporting Project Number (PNURSP2024R231).

Data Availability Statement: Data sharing is not applicable as no datasets were generated or analyzed during the current study.

Acknowledgments: The authors are very thankful to the anonymous referees for their valuable comments and suggestions which have improved the manuscript immensely. Moreover, we are very thankful to the "SageMath" open source community. This paper has been executed utilizing free open source arithmetic programming in SageMath [60]. The authors extend their appreciation to Princess Nourah Bint Abdulrahman University (PNU), Riyadh, Saudi Arabia for funding this research under Researchers Supporting Project Number (PNURSP2024R231).

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Gutte, V.S.; Parasar, D. Sailfish invasive weed optimization algorithm for multiple image sharing in cloud computing. *Int. J. Intell. Syst.* **2022**, *37*, 4190–4213. [CrossRef]
2. Wang, X.; Zhang, X.; Zu, C.; Yang, Z.; Bian, G.; Zhang, Y.; Ruan, W.; Wu, B.; Wu, X.; Yuan, L. An accurate cloud-based indoor localization system with low latency. *Int. J. Intell. Syst.* **2022**, *37*, 4794–4809. [CrossRef]
3. Ren, P.; Li, F.; Wang, Y.; Zhou, H.; Liu, P. Ipsadas: Identity-privacy-aware secure and anonymous data aggregation scheme. *Int. J. Intell. Syst.* **2022**, *37*, 5290–5324. [CrossRef]
4. Bufalo, M.; Bufalo, D.; Orlando, G. Some properties of the computation of the modular inverse with applications in cryptography. *Computation* **2023**, *11*, 70. [CrossRef]
5. Chen, H.-Y.; Wu, Z.-Y.; Chen, T.-L.; Huang, Y.-M.; Liu, C.-H. Security privacy and policy for cryptographic based electronic medical information system. *Sensors* **2021**, *21*, 713. [CrossRef]
6. Hazzazi, M.M.; Attuluri, S.; Bassfar, Z.; Joshi, K. A novel cipher-based data encryption with galois field theory. *Sensors* **2023**, *23*, 3287. [CrossRef]
7. Hernández-Álvarez, L.; Bullón Pérez, J.J.; Batista, F.K.; Queiruga-Dios, A. Security threats and cryptographic protocols for medical wearables. *Mathematics* **2022**, *10*, 886. [CrossRef]
8. Jara-Vera, V.; Sánchez-Ávila, C. Some notes on a formal algebraic structure of cryptology. *Mathematics* **2021**, *9*, 2183. [CrossRef]
9. Masood, F.; Ahmad, J.; Shah, S.A.; Jamal, S.S.; Hussain, I. A novel hybrid secure image encryption based on julia set of fractals and 3d lorenz chaotic map. *Entropy* **2020**, *22*, 274. [CrossRef]
10. Rauf, A.; Wang, Z.; Sajid, H.; Ali Tahir, M. Secure route-obfuscation mechanism with information-theoretic security for internet of things. *Sensors* **2020**, *20*, 4221. [CrossRef]
11. Richter, M.; Bertram, M.; Seidensticker, J.; Tschache, A. A mathematical perspective on post-quantum cryptography. *Mathematics* **2022**, *10*, 2579. [CrossRef]
12. Tito-Corrioso, O.; Borges-Quintana, M.; Borges-Trenard, M.A.; Rojas, O.; Sosa-Gómez, G. On the fitness functions involved in genetic algorithms and the cryptanalysis of block ciphers. *Entropy* **2023**, *25*, 261. [CrossRef]
13. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
14. Blakley, G.R. Safeguarding cryptographic keys. In Proceedings of the 1979 International Workshop on Managing Requirements Knowledge (MARK), New York, NY, USA, 4–7 June 1979; IEEE Computer Society: New York, NY, USA, 1979; pp. 313–318.
15. Simmons, G.J. How to (really) share a secret. In *Conference on the Theory and Application of Cryptography*; Springer: New York, NY, USA, 1988; pp. 390–448.
16. Asmuth, C.; Bloom, J. A modular approach to key safeguarding. *IEEE Trans. Inf. Theory* **1983**, *29*, 208–210. [CrossRef]
17. He, J.; Dawson, E. Multistage secret sharing based on one-way function. *Electron. Lett.* **1994**, *30*, 1591–1592. [CrossRef]
18. He, J.; Dawson, E. Multisecret-sharing scheme based on one-way function. *Electron. Lett.* **1995**, *31*, 93–95. [CrossRef]
19. Chien, H.Y.; Jan, J.K.; Tseng, Y.M. A practical (t, n) multi-secret sharing scheme. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2000**, *83*, 2762–2765.
20. Choc, B.; Goldwasser, S.; Micali, S.; Awerbuch, B. Verifiable secret sharing and achieving simultaneity in the presence of faults. In *Annual Symposium on Foundations of Computer Science (Proceedings)*; Elsevier: Amsterdam, The Netherlands, 1985; pp. 383–395.
21. Tompa, M.; Woll, H. How to share a secret with cheaters. *J. Cryptol.* **1989**, *1*, 133–138. [CrossRef]
22. Harn, L.; Hsu, C.-F. (t, n) Multi-secret sharing scheme based on bivariate polynomial. *Wirel. Pers. Commun.* **2017**, *95*, 1495–1504. [CrossRef]
23. Harn, L.; Hsu, C.-F.; Xia, Z.; Zhou, J. How to share secret efficiently over networks. *Secur. Commun. Netw.* **2017**, *2017*, 5437403. [CrossRef]

24. Ballico1, E.; Boato, G.; Fontanari, C.; Granelli, F. Hierarchical secret sharing in ad hoc networks through birkhoff interpolation. In *Advances in Computer, Information, and Systems Sciences, and Engineering: Proceedings of IETA 2005, TeNe 2005, EIAE 2005*; Springer: Dordrecht, The Netherlands, 2006; pp. 157–164.
25. Ma, C.; Cheng, R. Key management based on hierarchical secret sharing in ad-hoc networks. In *Information Security and Cryptology: Third SKLOIS Conference, Inscrypt 2007, Xining, China, August 31–September 5, 2007, Revised Selected Papers 3*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 182–191.
26. Zhang, E.; Li, M.; Yiu, S.-M.; Du, J.; Zhu, J.-Z.; Jin, G.-G. Fair hierarchical secret sharing scheme based on smart contract. *Inf. Sci.* **2021**, *546*, 166–176. [CrossRef]
27. Zhang, Y.; Liu, Z.; Huang, G. Sure interpolation and its application to hierarchical threshold secret sharing scheme. In Proceedings of the 2008 International Symposium on Computer Science and Computational Technology, Shanghai, China, 20–22 December 2008; IEEE: Piscataway, NJ, USA, 2008; Volume 1, pp. 447–450.
28. Yang, C.-C.; Chang, T.-Y.; Hwang, M.-S. A (t, n) multi-secret sharing scheme. *Appl. Math. Comput.* **2004**, *151*, 483–490. [CrossRef]
29. Blundo, C.; De Santis, A.; Stinson, D.R.; Vaccaro, U. Graph decompositions and secret sharing schemes. *J. Cryptol.* **1995**, *8*, 39–64. [CrossRef]
30. Padró, C.; Sáez, G. Secret sharing schemes with bipartite access structure. *IEEE Trans. Inf. Theory* **2000**, *46*, 2596–2604.
31. Tentu, A.N.; Bhavani, K.; Basit, A.; Venkaiah, V.C. Sequential (t, n) multi secret sharing scheme for level-ordered access structure. *Int. J. Inf. Technol.* **2021**, *13*, 2265–2275. [CrossRef]
32. Brickell, E.F. Some ideal secret sharing schemes. In *Workshop on the Theory and Application of of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 1989; pp. 468–475.
33. Lin, C.; Harn, L.; Yea, D. Ideal hierarchical (t, n) secret sharing schemes. In Proceedings of the Fifth International Conference on Information Assurance and Security (IAS09), Xi'an, China, 18–20 August 2009; Citeseer: Princeton, NJ, USA, 2009.
34. Chanu, O.B.; Tentu, A.N.; Venkaiah, V.C. Multi-stage multi-secret sharing schemes based on chinese remainder theorem. In Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology (ICARCSET 2015), Unnao, India, 6–7 March 2015; pp. 1–6.
35. Basit, A.; Kumar, N.C.; Venkaiah, V.C.; Moiz, S.A.; Tentu, A.N.; Naik, W. Multi-stage multi-secret sharing scheme for hierarchical access structure. In Proceedings of the 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, India, 5–6 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 557–563.
36. Banerjee, S.; Gupta, D.S.; Biswas, G. Hierarchy-based cheating detection and cheater identification in secret sharing schemes. In Proceedings of the 2018 4th International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, India, 15–17 March 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–6.
37. Liu, Y.; Zhang, F.; Zhang, J. Attacks to some verifiable multi-secret sharing schemes and two improved schemes. *Inf. Sci.* **2016**, *329*, 524–539. [CrossRef]
38. Yuan, J.; Yang, J.; Wang, C.; Jia, X.; Fu, F.-W.; Xu, G. A new efficient hierarchical multi-secret sharing scheme based on linear homogeneous recurrence relations. *Inf. Sci.* **2022**, *592*, 36–49. [CrossRef]
39. Bisht, K.; Deshmukh, M. A novel approach for multilevel multi-secret image sharing scheme. *J. Supercomput.* **2021**, *77*, 12157–12191. [CrossRef]
40. Prashanti, G.; Bhat, M.N. Cheating identifiable polynomial based secret sharing scheme for audio and image. *Multimed. Tools Appl.* **2024**, *83*, 403–423. [CrossRef]
41. Chattopadhyay, A.K.; Saha, S.; Nag, A.; Nandi, S. Secret sharing: A comprehensive survey, taxonomy and applications. *Comput. Sci. Rev.***2024**, *51*, 100608. [CrossRef]
42. Kamal, A.A.A.M.; Iwamura, K. Privacy preserving multi-party multiplication of polynomials based on (k, n) threshold secret sharing. *ICT Express* **2023**, *9*, 875–881. [CrossRef]
43. Liu, C.; Min, S.; Pang, Y.; Chai, Y. The meshfree radial point interpolation method (RPIM) for wave propagation dynamics in non-homogeneous media. *Mathematics* **2023**, *11*, 523. [CrossRef]
44. Chen, C.S.; Noorizadegan, A.; Young, D.L.; Chen, C.S. On the selection of a better radial basis function and its shape parameter in interpolation problems. *Appl. Math. Comput.* **2023**, *442*, 127713. [CrossRef]
45. Dupac, M. Mathematical modeling and simulation of the inverse kinematic of a redundant robotic manipulator using azimuthal angles and spherical polar piecewise interpolation. *Math. Comput. Simul.* **2023**, *209*, 282–298. [CrossRef]
46. Sauer, T.; Xu, Y. On multivariate Lagrange interpolation. *Math. Comput.***1995**, *64*, 1147–1170. [CrossRef]
47. Nevai, P. Mean convergence of Lagrange interpolation. III. *Trans. Am. Math. Soc.* **1984**, *282*, 669–698. [CrossRef]
48. Gautschi, W. On mean convergence of extended Lagrange interpolation. *J. Comput. Appl. Math.***1992**, *43*, 19–35. [CrossRef]
49. Herranz, J.; Sáez, G. New results on multipartite access structures. *IEE Proc.-Inf. Secur.* **2006**, *153*, 153–162. [CrossRef]
50. Dolev, D.; Yao, A. On the security of public key protocols. *IEEE Trans. Inf. Theory* **1983**, *29*, 198–208. [CrossRef]
51. Herranz, J.; Ruiz, A.; Sáez, G. New results and applications for multi-secret sharing schemes. *Des. Codes Cryptogr.* **2014**, *73*, 841–864. [CrossRef]
52. Mashhadi, S. Toward a formal proof for multi-secret sharing in the random oracle model. *Inf. Secur. J. A Glob. Perspect.* **2020**, *29*, 244–249. [CrossRef]
53. Halpern, J.; Teague, V. Rational secret sharing and multiparty computation. In Proceedings of the Thirtysixth Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, 13–16 June 2004; pp. 623–632.

54. Li, C.; Dong, M.; Xin, X.; Li, J.; Chen, X.B.; Ota, K. Efficient privacy-preserving in IoMT with blockchain and lightweight secret sharing. *IEEE Internet Things J.* **2023**, *10*, 22051–22064. [CrossRef]
55. Buchanan, W.J. Shamir Secret Sharing (SSS). Asecuritysite.com. 2024. Available online: https://asecuritysite.com/shares (accessed on 30 April 2024).
56. Wu, S.; Hsu, C.; Xia, Z.; Zhang, J.; Wu, D. Symmetric-bivariate-polynomial-based lightweight authenticated group key agreement for industrial internet of things. *J. Internet Technol.* **2020**, *21*, 1969–1979.
57. Knuth, D.E. *The Art of Computer Programming*; Pearson Education: London, UK, 2005.
58. Security and So Many Things—Asecuritysite.com. Available online: https://asecuritysite.com/ (accessed on 30 April 2024).
59. Chen, Q.; Tang, C.; Lin, Z. Efficient explicit constructions of multipartite secret sharing schemes. *IEEE Trans. Inf. Theory* **2021**, *68*, 601–631. [CrossRef]
60. SageMath. Use SageMath Online—Cocalc.com. Available online: https://cocalc.com/features/sage (accessed on 30 April 2024).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.