Hierarchical secret sharing scheme for WSN based on linear homogeneous recurrence relations

Yan Lin¹, Hongliang Zhu¹,², Guoai Xu¹ and Guosheng Xu¹

Abstract
Wireless sensor network is a key technology in the sensing layer of the Internet of Things. Data security in wireless sensor network is directly related to the authenticity and validity of data transmitted in the Internet of Things. Due to the large number and different types of nodes in wireless sensor networks, layered secret key sharing technology is increasingly used in wireless sensor networks. In a hierarchical secret sharing scheme, participants are divided into sections with different permissions for each team, but the same permissions for participants in the same team. In this article, we follow the approach of the hierarchical secret sharing scheme derived from the linear homogeneous recurrence relations. We design a hierarchical multi-secret sharing scheme for wireless sensor networks on the basis of the elliptic curve public key cryptosystem combined with the linear homogeneous recurrence relations. In the proposed scheme, we do not make sure that the participants are half-truthful. In addition, the participants’ shadows can be reused. Our scheme is computational security. Only one share from each member is required in our hierarchical multi-secret sharing scheme. It is more suitable for wireless sensor networks compared to the up-to-date schemes.

Keywords
Hierarchical secret sharing scheme, elliptic curve public key cryptosystem, linear homogeneous recurrence relations

Date received: 20 December 2021; accepted: 24 February 2022

Handling Editor: Yanjiao Chen

Introduction
In a secret sharing scheme of \((t, n)\) threshold,¹ the secret is passed around in \(n\) participants, and when it is required, any \(t\) or more participants can retrieve the shared secrets by combining their shares. A handy case of secret sharing is the distributed storage of sensitive information. Such a solution is called perfect scheme if any participant of any ineligible subset does not have access to any information about the shared secrets. Shamir¹ and Blakley² raised the threshold secret sharing schemes, two exceptional cases where all the participants have the same privileges.

There are restrictions on threshold secret sharing schemes in reality. Therefore, to enhance the feasibility of secret sharing in practice, many researchers concentrate on particular sets of access structures, for instance, weighted threshold access structure, compartmented access structure, graph-based access structures,³

¹School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing, China
²National Engineering Laboratory for Disaster Backup and Recovery, Beijing University of Posts and Telecommunications, Beijing, China

Corresponding authors:
Hongliang Zhu, School of Cyberspace Security, Beijing University of Posts and Telecommunications, 10 Xitucheng Road, Haidian District, Beijing 100876, China.
Email: zhuhongliang@bupt.edu.cn
Guoai Xu, School of Cyberspace Security, Beijing University of Posts and Telecommunications, 10 Xitucheng Road, Haidian District, Beijing 100876, China.
Email: xga@bupt.edu.cn

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
bipartite access structures and hierarchical access structure. A large number of studies in our community have been focused on the protection (e.g. privacy, security, malicious behaviours) of our lives, towards constructing a safer information environment. Shamir proposed the weighted threshold secret sharing scheme. But the secret sharing scheme is a trivial solution by assigning multiple shares to each participant according to its integral weight, which is inefficient. Simmons presented a multipartite access structure, and the definitions of the compartmented access structure and the hierarchical access structure were given by him. After Simmons, Brickell introduced an approach to give an ideal secret sharing scheme for the multilevel and compartmented access structures. The multipartite access structure can classically be divided into the hierarchical access structure and the compartmented access structure.

Tassa proposed some hierarchical threshold access structures rested on Birkhoff interpolation, and the interpolation matrix must satisfy Polya’s condition. Farras et al. gave a comprehensive characteristic of the ideal multipartite access structures. Then the scholars investigated the hierarchical secret sharing schemes by using the findings on integer polymatroids in the solution given by Farras et al. These techniques offered a characterization of the hierarchical access structures and by these techniques, there admits the ideal and perfect secret sharing scheme existing for the hierarchical access structures. Chen et al. proposed a method to give an ideal scheme that implements the compartmented access structure by the general polymatroid-based method presented in Farras et al.’s solution and the Gabidulin codes. Chen et al. proposed a method of implementing the multipartite access structure. Motivated by the integer polymatroids, the main idea of Chen et al.’s scheme from Brickell provides a polynomial-time algorithm to build such a matrix , which makes all the determinants of those particular multi-matrices non-zero in certain limited domains.

The methods mentioned above do not introduce verifiability. Also, researchers have worked out verifiable resolutions for the classic secret sharing schemes. Motivated by Birkhoff interpolation, Traverso et al. gave a verifiable hierarchical secret sharing method. But this method has the shortcomings of these schemes rested on Birkhoff interpolation. The drawback of the above hierarchical threshold access structure is that the distributor must perform exponential examinations when allocating identifications and shares to the members. Xu et al. proposed an efficient compartmented secret sharing scheme. Yuan et al. provided an ideal polynomial-time hierarchical secret sharing scheme based on linear homogeneous recurrence (LHR) relations. We present a verifiable hierarchical secret sharing scheme rested on Xu et al. So, our method is more efficient than Traverso’s solution.

The rest of the article focuses on the following sections. The second section leads to the secret sharing scheme, LHR relations, as well as Bilinear Pairing and Diffie–Hellman problem. The third section describes the system proposed in our article. In the fourth section, we demonstrate the security analysis of our solution, the significant attributes of our solution and show the comparisons with other solutions. In the fifth section, conclusions are provided.

**Preliminary**

**Secret sharing schemes**

In this part, we show the statement of the hierarchical access structure.

**Definition 1.** A $(t,n)$ threshold secret sharing scheme $\prod : S \times R \to S_1 \times S_2 \times \cdots \times S_n$ over $P$ ($P$ is the aggregate of game participants, that is $P = \{P_1, P_2, ..., P_n\}$ and $|P| = n$), satisfying the following conditions, in which $S$ is the shared secret space, $S_i$ ($1 \leq i \leq n$) is the share space, and $R$ is an aggregate of random feeds.

A brief overview of the hierarchical access structure is provided in the following section.

**Definition 2.** Make $n$ indicate the total number of participants. In the hierarchical secret sharing scheme, the participants $P$ can be grouped into classes $\gamma_1, \gamma_2, ..., \gamma_m$ of the participants, $P = \cup_{i=1}^m \gamma_i$ and $\gamma_i \cap \gamma_j = \emptyset$ for all $1 \leq i < j \leq m$. The class $\gamma_i$ contains $n_i$ participants, in which $i \in \{1, 2, ..., m\}$. Let $K = \{k_i\}_{i=1}^m$ be a set of all $k_i$ in ascending sequence, $0 < k_1 < k_2 < \cdots < k_m$. The $(K,n)$--hierarchical threshold access structure is

$$\text{AS} = \{A \subseteq P : |A \cap \left( \bigcup_{j=1}^m \gamma_j \right)| \geq k_i, \forall i \in \{1, 2, ..., m\}\}$$

**LHR relations**

We briefly describe the LHR relations. The LHR relations are introduced in detail in the studies by Xu and other peers.

**Theorem 1.** Let $h_0, h_1, ..., h_n$ be a series of numbers and $\alpha_1, \alpha_2, ..., \alpha_m$ be the different roots of the following characteristic equation of the LHR relation with constant coefficients

$$h_j = a_1h_{j-1} + a_2h_{j-2} + \cdots + a_nh_{j-n}$$

(1)
where $a_i \neq 0$, $a_i$ is chosen over $GF(q)$ ($j \geq t$), and $q$ is a large prime. 

If $a_i$ is a $t$-fold root of the characteristic equation (1), then the part of the resolution of the recurrence relation with respect to $a_i$ is given as 

$$F_j^{(i)} = c_{i1}\alpha_i^j + c_{i2}\alpha_i^{j^2} + \cdots + c_{in}\alpha_i^{j^{n-1}}$$ 

$$= (c_{i1} + c_{i2}j + \cdots + c_{in}j^{n-1})\alpha_i^j$$

Let $f_i(j) = (c_{i1} + c_{i2}j + \cdots + c_{in}j^{n-1})$. We can have 

$$F_j^{(i)} = f_i(j)\alpha_i^j$$

For recursive relations, the usual solution is derived from the following 

$$h_j = F_j^{(1)} + F_j^{(2)} + \cdots + F_j^{(m)}$$

where $t = \sum_{i=1}^{m} a_i^j$; 

If $\alpha_1 = \alpha_2 = \cdots = \alpha_m = \alpha$, then for the recurrence relation the general solution is 

$$h_j = F_j$$

where 

$$F_j = (c_{11} + c_{12}j + \cdots + c_{1n}j^{n-1})\alpha^j$$

**Diffie–Hellman problem**

Given a cyclic additive group $G = \langle g \rangle$, with $g$ being the generator and the order of $G$ being the large prime $q$,

1. The problem of the discrete logarithm is defined as computing $a$, provided $(g, ag)$, where $\forall a \in \mathbb{Z}_q$; 
2. The computational Diffie–Hellman (CDHP) problem is defined as computing $(ab)g$, provided $(g, ag, bg)$, where $\forall a, b \in \mathbb{Z}_q$; 
3. The decisional Diffie–Hellman problem (DDHP) is defined as deciding whether $(ab)g = cg$, provided $(g, ag, bg, cg)$, where $\forall a, b, c \in \mathbb{Z}_q$.

It is called the GDH group when DDHP is solvable in polynomial time, but no probabilistic algorithm can solve CDHP in the polynomial time with a non-negligible advantage.

**Bilinear pairing**

Suppose that $G_1$ and $G_2$ are an additive group and a multiplicative group, respectively, with the same prime order $q$. Here $q$ is a large prime. Let $g$ be a generator of $G_1$. Suppose that DDHP is easy in $G_1$ and hard in $G_2$.

We assume CDHP in $G_1$ and the DDHP in $G_2$ are hard. We call it the cryptographic bilinear mapping, if a mapping $\hat{e} : G_1 \times G_1 \rightarrow G_2$ satisfies the following properties:

1. Bilinearity: $\forall H_1, Q_1 \in G_1, \forall a, b \in \mathbb{Z}_q$, $\hat{e}(ah_1, bQ_1) = \hat{e}(H_1, Q_1)^{ab}$.
2. Non-degeneracy: If $g$ is a generator of $G_1$, then the generator of $G_2$ is $\hat{e}(g, g)$. In other words, if $g \neq 0$, then $\hat{e}(g, g) \neq 1$.
3. Computability: $\forall H_1, Q_1 \in G_1$, there is a polynomial-time algorithm to compute $\hat{e}(H_1, Q_1)$.

**The proposed scheme**

In the following section, we show how the proposed scheme works. The proposed scheme is built upon the LHR relations and the elliptic curve public key cryptosystem (ECC). $P = \{P_1, P_2, \ldots, P_n\}$ denotes the set of the participants, where $P_i$ is the $i$th participant in the set, $1 \leq i \leq n$. Our scheme is composed of three stages, namely, initialization stage, construction stage and recovery stage. In the initialization stage, we initialize some functions or parameters; in the construction stage, the shared secrets are hidden in these terms $h_1, \ldots, h_t$, and in the recovery stage, we recover the shared secrets by getting these terms $h_1, \ldots, h_t$ first. The distributor is honest in our scheme. But the participants can be malicious, because in our scheme the participants can be detected if they provide fake pseudo shares. The distributor acts the role in the construction stage and the participants act the role in the recovery stage.

We first describe the main idea of the proposed scheme. The proposed scheme contains $n$ participants and one dealer/distributor. Only one shadow $x_i \in \mathbb{Z}_q^*$ is held by each participant $P_i$. The distributor selects $m$ random LHR relations. In the construction stage, the participants in $\gamma_i$ are used to construct $m$ LHR relations and the participants in $\gamma_i$ are used to build $m - i + 1$ LHR relations, where $2 \leq i \leq m$. So some participants can be replaced by other participants who have the higher privilege. That is, the participants in $\gamma_i$ can replace the participants in $\gamma_{i-1}$, where $i - j \geq 1$. Then, the distributor adds the $m$ general terms of the $m$ LHR relations. From Theorem 1, we know that the sum is a general term of an LHR relation and we denote this LHR relation as $F$-LHR relation. The distributor selects the shared secrets and keeps the shared secrets hidden in the first several terms of $F$-LHR relation. So in the recovery stage, the recovery of the shared secrets is achieved through resolving the general term of $F$-LHR relation. After that, we obtain the items that hide the shared secrets. The notations used are shown in Table 1.
the case where D asked by another party.

In this way, it is possible to protect the key distribution in the section 'Bilinear pairing' can be satisfied.

In the construction stage, we define a double-argument one-way function as given in the study by Chien et al. Given random value \( r \) and value \( w \), the \( f(r, w) \) represents a double-argument one-way function that is easily computed. In some cases, though, it is difficult to be computed:

1. Knowing \( f(r, w) \) and \( w \), \( r \) is hard to be worked out.
2. Unknowing \( w \), \( f(r, w) \) is hard to be worked out for any \( r \).
3. Knowing \( f(r, w) \) and \( r \), \( w \) is hard to be calculated.
4. Knowing \( f(r, w) \) and \( r \), for \( r' \neq r \), \( f(r', w) \) is hard to be calculated.
5. Knowing \( w \), it is quite a challenge to get two such distinct characters \( r_2 \) and \( r_1 \) that \( f(r_2, w) = f(r_1, w) \).

**Construction stage**

In the construction stage, the key distributor \( D \) first generates \( m \) different LHR sequences \( \{h_1^{(1)}, h_2^{(1)}, \ldots, h_m^{(1)}\} \). In our scheme, \( k_1, k_2, \ldots, k_m \) are the degrees of the characteristic equation of the LHR relations \( \{h_1^{(1)}, h_2^{(1)}, \ldots, h_m^{(1)}\} \), where \( h_i^{(1)} \) denotes the \( i \)-th LHR relation. Figure 1 shows the construction stage, where \( \gamma_1 + \gamma_2 + \cdots + \gamma_m = 1 \) denotes \( \bigcup_{i=1}^{m} \gamma_i \) and Pshare denotes the pseudo share.

**Theorem 2.** The addition of two general terms of two different LHR relations is a general term of an LHR relation.

**Proof of Theorem 2.** Suppose that \( h_1^{(1)} \) and \( h_2^{(1)} \) are the general terms of two different LHR relations with the different degrees \( t_1 \) and \( t_2 \), respectively, and assume \( h_i = h_1^{(1)} + h_2^{(1)} \). So, it is only necessary to prove that \( h_i \) is the general term of an LHR relation. From Theorem 1, we know \( h_i \) is the generic term of an LHR relation, and \( t_1 + t_2 \) is the degree of the LHR relation \( (h_i) \).

The distributor \( D \) distributes the shares through the following procedure:

1. The distributor \( D \) randomly selects the integer \( r_1, r_2, \ldots, r_m \) and publishes them.
2. \( D \) chooses \( s_0 \in \mathbb{Z}_q^* \), computes \( B_0 = s_0 g \in G_1 \), makes sure \( B_0 \neq B_i \) and publishes \( B_0 \), where \( 1 \leq i \leq n \).
3. The distributor \( D \) chooses \( m \) different integers \( \alpha_1, \alpha_2, \ldots, \alpha_m \) over \( GF(q) \) and publishes them, where each of them is non-zero.
4. \( D \) selects the value \( \alpha_1 \) to make \( (x - \alpha_1)^{t_1} = x^{t_1} + a_{11} x^{t_1-1} + \cdots + a_{1t_1} = 0 \) as

| Notation | Description |
|----------|-------------|
| \( P = \{P_1, P_2, ..., P_n\} \) | The set of participants |
| \( P_i \) | The \( i \)-th participant in set \( P \) |
| key\(_i\) | The \( i \)-th shared secret |
| \( s_i \in \mathbb{Z}_q^* \) | The shadow hold by the \( i \)-th participant \( P_i \) |
| \( s_0 \in \mathbb{Z}_q^* \) | The shadow hold by the distributor \( D \) |
| \( \gamma_i \) | The \( i \)-th subset of the participant |
| \( h_i \) | The \( i \)-th term in the sequences \( \{h_i\}_0^n = 0 \) |
| \( q \) | The large prime |
| \( GF(q) \) | The finite field |
| \( G_1 \) | The additive group with the generator \( g \) and the order of \( G_1 \) is \( q \) |
| \( G_2 \) | The multiplicative group with the order \( q \) |
| \( B_0 \) | The share \( g s_0 \) generated by distributor \( D \) and published |
| \( B_i \) | The share \( g s_i \) was generated by the \( i \)-th participant and published |
| \( R_0 \) | The pseudo share \( s B_0 \) |
| \( f(r, w) \) | The double-parameter one-way function |
| \( h_i^{(j)} \) | The \( j \)-th linear homogeneous recurrence relation |
| \( r_1, r_2, ..., r_m \) | \( m \) random values |
the auxiliary function of an LHR relation, where $q > a_{i1}$ and $1 \leq i \leq k_1$.

5. $D$ computes the pseudo share $R_i = s_0B_i$ and $l_i^j = f(r_1, R_i)$, where $i = 1, 2, \ldots, n_1$, and $n_1$ is the quantity of participants in the sub-group $\gamma_1$.

6. $D$ considers the first LHR sequence which is defined as

$$
\begin{align*}
&h_0^{(1)} = l_1^1, h_1^{(1)} = l_2^1, \ldots, h_{k_1-1}^{(1)} = l_k^1, \\
&h_i^{(1)} + a_{i1}h_j^{(1)} + a_{i2}h_{j+1}^{(1)} + \ldots + a_{ik}h_{j+k-1}^{(1)} = 0 \quad i \geq 0
\end{align*}
$$

(3)

We use the participants in the sub-group $\gamma_1$ to build the first LHR sequence. During the recovery stage, the general term of the first LHR sequence is allowed to be retrieved only by the participants in the sub-group $\gamma_1$.

7. $D$ computes $h_i^{(1)}$, where $k_1 \leq i \leq n_1 - 1$.

8. $D$ computes $y_i^{1} = l_i^1 - h_i^{(1)}$ and publishes $y_i^1$, where $k_1 < i \leq n_1$.

9. $D$ continues to generate $m - 1$ LHR sequences. The $j$th LHR sequence $\{h_i^{(j)}\}$ is generated as follows, for $j \in \{2, 3, \ldots, m\}$.

1. $D$ selects the value $\alpha_j$ to make

$$
(x - \alpha_j)^h = x^h + a_{i1}x^{h-1} + \ldots + a_{ki}x = 0
$$

as an auxiliary function of an LHR relation, where $q > a_{i1}$ and $1 \leq i \leq k_i$.

2. $D$ computes $l_i^j = f(r_j, R_i)$, where $i = 1, 2, \ldots, n_1 + n_2 + \ldots + n_j$, and $n_1 + n_2 + \ldots + n_j$ is the number of the participants in the subset $\bigcup_{j=1}^{i-1} \gamma_i$, where $n_1 + n_2 + \ldots + n_j$ is denoted as $N_j$. That is, $|\bigcup_{j=1}^{i-1} \gamma_i| = n_1 + n_2 + \ldots + n_j = N_j$.

3. $D$ considers the $j$th LHR sequence $\{h_i^{(j)}\}$, which is defined by

$$
\begin{align*}
&h_0^{(j)} = l_1^j, h_1^{(j)} = l_2^j, \ldots, h_{k_{j-1}}^{(j)} = l_{k_j}^j, \\
&h_i^{(j)} + a_{i1}h_j^{(j)} + a_{i2}h_{j+1}^{(j)} + \ldots + a_{ik}h_{j+k-1}^{(j)} = 0 \quad i \geq 0
\end{align*}
$$

(4)

We use the participants in the subset $\bigcup_{j=1}^{i-1} \gamma_i$ to construct the $j$th LHR sequence. In the recovery stage, only the participants in the subset $\bigcup_{j=1}^{i-1} \gamma_i$ can recover the general term of the $j$th LHR sequence.

4. $D$ computes $h_i^{(j)}$, where $k_j \leq i \leq N_j - 1$.

5. $D$ computes $y_i^{j} = l_i^j - h_i^{(j)}$ and publishes $y_i^j$, where $k_j < i \leq N_j$.

10. After the $m$ LHR relations are generated, $D$ combines them into an LHR relation. Let

$$
h_i = h_i^{(1)} + h_i^{(2)} + \ldots + h_i^{(m)}
$$

From Theorem 2, we can determine that $h_i$ is the general term of an LHR relation and the degree of this LHR relation $(h_i)$ is $k_1 + \cdots + k_m$. Since the degrees of these LHR relations $(h_i^{(1)}, \ldots, h_i^{(m)})$ correspond to $k_1, \ldots, k_m$, respectively, so the sum of these degrees is $k_1 + \cdots + k_m$. However, we define the threshold of the hierarchical secret sharing scheme as $k_m$, because $k_m$ different participants can recover the shared secrets (Since at least $k_1$ participants from the subset $\gamma_1$, at least $k_2 - k_1$ participants from the subset $\gamma_2$ and so on, and the last $k_m - k_{m-1}$ participants from the subset $\gamma_m$ can recover the LHR relation $(h_1)$, the threshold of the hierarchical secret sharing scheme is $k_1 + (k_2 - k_1) + \cdots + (k_m - k_{m-1}) = k_m$).

11. $D$ computes $d_i = key_i - h_{i-1}$ and publishes $d_i$, where $1 \leq i \leq l$. This means that we can hide the shared secrets in these terms $h_0, h_1, \ldots, h_{l-1}$.

Remark 1. The double-parameter one-way function makes sure that for the same $R_i$, the random values $f(r_i, R_i)$ and $f(r_j, R_i)$ are different, where $k \neq j$. This means that if some participants’ pseudo shares are used to construct different LHR relations, the double-parameter one-way function makes sure that for the same pseudo share, the random values are different.

Recovery stage

In this section, we present how the shared secrets are recovered. First, a proposition is given as follows.

Proposition 1. Provided that $\alpha_j$ is a $t_j$-fold root of the characteristic formula of an LHR relation and the general resolution of this LHR relation is obtained as

$$
h_j = \sum_{l=1}^{m} \left( \sum_{k=1}^{t} c_{ik} \alpha_{ik}^{k-1} \right) \alpha_{ij}^{j-1}
$$

its coefficient $c_{ik}$ can then be fixed by the $t$ initial values, which is achieved by resolving the system of linear formulas, in which $t = \sum_{i=1}^{m} t_i$.

The participant $P_i$ can get every participant’s pseudo share $R_i$ by exchanging in a qualified sub-group, where $i \neq j$. In the qualified sub-group, the participants need to recover $m$ LHR sequences before obtaining the shared secrets. The $j$th LHR sequence is recovered as follows.

From the construction stage, we know that the order of the $j$th LHR relation is $k_j$. We need at least $k_j$ participants when we are trying to recover the $j$th LHR sequence. But there are $k_{j-1}$ participants from $\bigcup_{i=1}^{j-1} \gamma_i$ (if $k_{j-1}$ participants use their pseudo shares to solve the general term $h_i^{(j-1)}$ of the $(j-1)$th LHR relation, they need to be combined to solve the general term $h_i^{(j)}$ of
obtaining from gh must use their pseudo shares to recover the general terms of the sequence \( f_i \). According to Theorem 1, we can say that the participant \( g_h \) presents different polynomial. According to Theorem 1, \( h_i^{(j)} = p^j(i)\alpha_j^i \), in which \( p^j(i) \) is a \( k_j - 1 \) degree polynomial and \( i \geq 0 \), we can have the general term \( h_i^{(j)} \). After obtaining \( m \) general terms \( h_i^{(1)}, h_i^{(2)}, \ldots, h_i^{(m)} \), the general term \( h_i \) is

\[
h_i^{(1)} + h_i^{(2)} + \cdots + h_i^{(m)}
\]

where the order of the LHR relation \( (h_i) \) is \( k_1 + \cdots + k_m. \) But the global threshold is \( k_m \) (If \( k_j - k_{j-1} \) participants use their pseudo shares to recover the general term \( h_i^{(j)} \), then they must use their pseudo shares to recover the general terms \( h_i^{(j+1)}, \ldots, h_i^{(m)} \). For example, if \( k_1 \) participants from \( \gamma_1 \) use their pseudo shares to recover the general term \( h_i^{(1)} \), \( k_1 \) participants need to use their pseudo shares to recover the general terms from the second LHR sequences to the last LHR sequences. So the scheme needs at least \( k_1 + (k_2 - k_1) + \cdots + (k_m - k_{m-1}) = k_m \) participants to recover the shared secrets). From \( d_i = key_i - h_i \) and the published value of \( d_i \), the shared secrets can be obtained by \( key_i = d_i + h_{i-1} \), \( 1 \leq i \leq l \).

**Remark 2.** In the recovery stage, a participant \( P_j \) can detect whether another participant \( P_i \) is honest or not by the function \( \hat{\epsilon} \). If \( \hat{\epsilon}(g_i, R_i) = \hat{\epsilon}(B_0, B_i) \), then the participant \( P_i \) can say that the participant \( P_j \) is honest, because \( \hat{\epsilon}(g_i, R_i) = \hat{\epsilon}(g_i, B_0) \land \hat{\epsilon}(B_0, B_i) \) and \( B_0, B_i \) are public.

**Illustrative sample**

We present an example to demonstrate the proposed scheme and show how the LHR relation is useful in the hierarchical access structures in this section. For illustration convenience, we suppose that in a company there are \( n \) employees, and the employees are divided into two disjoint levels \( \gamma_1, \gamma_2 \), the managers and the employees (the authorities of the managers are higher than the employees). There are \( n_1 \) managers in the subset \( \gamma_1 \) and \( n_2 \) employees in the subset \( \gamma_2 \), where \( n = n_1 + n_2 \). At least \( k_2 \) staff of them can recover the shared secrets, among which at least \( k_1 \) are managers, where \( k_2 > k_1 \). We select two values \( \alpha_1 \) and \( \alpha_2 \), and make

\[
(x - \alpha_1)^{k_1} = x^{k_1} + a_{11}x^{k_1-1} + \cdots + a_{k_1} = 0
\]

\[
(x - \alpha_2)^{k_2} = x^{k_2} + a_{21}x^{k_2-1} + \cdots + a_{2k_2} = 0
\]

as two different auxiliary functions for the different LHR relations. After the steps from 5 to 9.5 as described in the construction stage have been done (with \( m = 2 \) here), we can have

\[
h_i = h_i^{(1)} + h_i^{(2)}
\]

where

\[
h_i^{(1)} = (b_1 + b_2i + \cdots + b_{k_1}i^{k_1-1})\alpha_1^i
\]

\[
h_i^{(2)} = (c_1 + c_2i + \cdots + c_{k_2}i^{k_2-1})\alpha_2^i
\]

We note that the distributor \( D \) uses \( n_1 \) managers’ pseudo shares to generate the LHR sequence \( \{h_i^{(1)}\} \) and \( n_1 + n_2 \) staff members (include the managers and the employees) to generate the LHR sequence \( \{h_i^{(2)}\} \). That is, only at least \( k_1 \) managers can solve the general term \( h_i^{(1)} \) of the sequence \( \{h_i^{(1)}\} \). Still, at least \( k_2 \) staff members can solve the general term \( h_i^{(2)} \) of the sequence \( \{h_i^{(2)}\} \), where \( k_1 \) out of \( k_2 \) are managers (Because the privilege of the managers is higher than the employees, they should combine the pseudo shares to solve the second general term \( h_i^{(2)} \)). If the employees are not enough, other managers except the \( k_1 \) can combine the employees to pool the pseudo shares to solve \( h_i^{(2)} \). We hide the shared secrets in these terms \( h_1, \ldots, h_l \).

From the above, we can get that our scheme’s global threshold is \( k_2 \), because our scheme needs at least \( k_2 \) participants from \( \gamma_1 \cup \gamma_2 \) to recover the shared secrets.

**The security analysis and discussion of our scheme**

In this section, we first present the possible attacks and prove that our scheme is secure. Then we show the performance of our scheme and compare it with the scheme by Traverso et al.19
Security analysis

In this section, we call all of $s_i$ and $R_j$ share-related information, where $s_i$ is the shadow that is randomly selected by the participant or distributor himself or herself, $B_i = s_i g_i$ (is public), $R_j = s_j g_j$ and $0 \leq i \leq n$, $1 \leq j \leq n$. First, we prove the security of the share-related information. Second, we prove the security for the participants in the unqualified subset. The security of the share-related information is easy to be proved.

Theorem 3. The share-related information is secure, provided the CDH-assumption is intractable.

Proof of Theorem 3. If the adversaries want to break the share-related information, they should obtain useful information from the public values $B_i = s_i g_i$, $0 \leq i \leq n$. We assume that the adversaries can break the share-related information with the non-negligible probability $\epsilon$. That is, given $s_i g_i$ and $s_j g_j$, the adversaries can get $s_i$ or $s_j$ with the non-negligible probability $\epsilon$ as well, where $i \neq j$. This contradicts the Diffie–Hellman problem. The adversaries may give a fake $R'_j$, but the other honest participants can decide whether $\hat{e}(g, R'_j) = \hat{e}(B_0, B_i)$ is right or not (because DDHP is easy in $G_1$). If it matches, then the other honest participants can say that the value $R'_j$ is valid. Otherwise, it is invalid. Thus, the share-related information is secure.

In the following section, we demonstrate why the proposed solution is safe for a subset of participants who do not qualify. When participants in the ineligible subset want to recover the secrets, they must recover every LHR sequence. For illustration convenience, we suppose that the unqualified subset $B$ satisfies the conditions as follows

1. $B \cap \bigcup_{1 \leq t \leq m} \gamma_t = \emptyset$
2. $B \cap \bigcup_{1 \leq t \leq m} \gamma_t = \{1 \leq t \leq m\}$
3. $B \cap \bigcup_{1 \leq t \leq m} \gamma_t = \emptyset$

where $1 \leq t \leq m$, $t \in \{1, 2, ..., m\}$, $k_0 = 0$ and $t \neq j$. That is to say, the unqualified subset can recover other $m - 1$ sequences, except the $j$th LHR sequence. Since $s_i$ is chosen by himself/herself and $I_{ji} = f(r_j, R_i)$, we can say that each $I_{ji}$ is uniformly chosen in $Z_q^*$, that is, for each $I_{ji}$, all values in $Z_q^*$ have the same probability. From the public value $a_i$, the characteristic equation of an LHR relation can be determined, according to Theorem 1. If the relation of the sequences is given, the characteristic equation can be determined, and then the root of the characteristic equation can be found. Thus the public value $a_i$ discloses no information apart from characteristic equation of an LHR relation, $1 \leq i \leq m$.

Theorem 4. The $k_j$ degree of the LHR relation is safe to the unqualified participants when and only when the
polynomials with \((k_j - 1)\) degree is safe to the unqualified participants.\(^{20}\)

**Proof of Theorem 4.** Assume that the \(k_j\) order of the LHR relation is safe to the unqualified participants. Using equations (4), (6) and (2) and the public value \(\alpha_j \neq 0\), we could obtain

\[
h_i^{(j)} = p'(i)\alpha_j^j \Rightarrow h_i/\alpha_j = p'(i) \quad (7)
\]

in which the degree of \(p'(i)\) is \(k_j - 1\). It is clear from the above that public value \(\alpha\) gives away no information except characteristic equation. If the polynomial with degree \((k_j - 1)\) is unsafe for unqualified participants, it means that the \(k_j - 1\) points can determine a polynomial with degree \((k_j - 1)\). We can also deduce from equation (7) that the \(k_j - 1\) value can obtain the LHR relation with the degree \(k_j\). This is contradicted with Proposition 1.

\((\Rightarrow)\) Assume that the polynomial with the degree \((k_j - 1)\) is safe for the unqualified participants. If the LHR relation with degree \(k_j\) is insecure, then \(k_j - 1\) random terms \((h_1^{(j)}, h_2^{(j)}, ..., h_{k_j}^{(j)})\) can determine the LHR sequences. Following equation (7), we select the distinct \(k_j - 1\) terms and then can obtain the \(k_j - 1\) points of the polynomial \(p'(i)\) with degree \(k_j - 1\). Then we can determine the polynomial \(p'(i)\). It can be said that the \(k_j - 1\) points could decide a polynomial with degree \((k_j - 1)\). This contradicts our assumptions. Therefore, the question of recovering the \(j\)th LHR relation from participants in the disqualified subset \(B\) that satisfies the aforementioned condition could be regarded as the question of whether \(k_j - 1\) points could decide the polynomial with the degree \((k_j - 1)\).

We can draw the conclusion that the possibility of cracking our scheme is no better than the possibility of cracking the Diffie–Hellman problem. Therefore, it can be considered that our scheme is safe.

**Performances**

From the section ‘The proposed scheme’ and the security analysis above, we conclude that the proposed scheme is a secure and verifiable hierarchical secret sharing scheme. In this section, we show some essential characteristics. In the proposed scheme, each participant’s ID is posted on a publicly available message board, and each participant’s shadow is selected by himself/herself over \(GF(q)\). Each participant can verify whether the pseudo-random shares of other participants are correct. Throughout the entire process, each participant only needs to hold one shadow, and the shared secrets are selected over \(GF(q)\). So the shadow held by a participant is as long as the shared secret.

Each participant obtain \(k_m - 1\) pseudo-random shares from the other participants, and each participant’s communication and computation is the same (Since \(k_m\) participants are required to recover the secrets, each participant needs to communicate with \(k_m - 1\) participants, so the times of the communication for each participant is \(k_m - 1\)). The computation refers to recovering the secrets with the obtained pseudo-random shares (including their own and those obtained by exchanging with other participants). In the recovery stage, from equation (6), we know that computational complexity of our scheme depends on the degree of polynomial \(p^m(x)\) (because the degree of the polynomial \(p^m(x)\) is highest among these polynomials \(p^i(x)\), where \(1 \leq i \leq m\) and the power of a constant \(\alpha_m\). The computational complexity of the polynomial \(p^m(x)\) is \(O(n^{m-1})\) and the computational complexity of the power of a constant is \(O(\log n)\). So the computational complexity of our scheme is \(O(n^{m-1}\log n)\).

In the case when the system needs to be reinitialized with the LHR relations, our scheme only needs to provide the new values \(r_1, r_2, ..., r_m\) without changing participants’ shadows. It is possible to obtain a further \(I_i = f(r_j, R_i)\) while not modifying the shadows.

**Comparisons**

In the solution by Traverso et al.,\(^{19}\) when allocating identities and shares to participants, the distributor must perform a potentially exponential number of checks. In our scheme, participants do not have to check the non-regularity of many matrices. They just need to construct \(m\) different LHR relations, find the general terms of them and add these terms. More precisely, we can conclude that the participants only need to do the evaluation calculation of polynomials and the summation calculation of different polynomials in our scheme by induction. The computational complexity of the polynomial is \(O(n^{m-1})\) and the computational complexity of the power of a constant is \(O(\log n)\). So the computational complexity of our scheme is \(O(n^{m-1}\log n)\) as given in the study by Yuan et al.\(^{21}\) But more values should be published. So comparing to Traverso’s et al.\(^{19}\) scheme, our scheme is more efficient.

**Conclusion**

We present a hierarchical secret sharing scheme with the LHR relations and the ECC. The performance of the proposed method is also listed below. The LHR relations could be utilized for constructing a verifiable
hierarchical secret sharing scheme. The participants’ shadows are selected by themselves in the proposed scheme. Therefore, no deception exists between the distributor of the shared secrets and the participants. In the whole scheme, only one shadow is held by one participant, and the secret is the same length as the shadow. The proposed scheme is not required to keep a safe channel between the distributor and the participant. The participants cannot cheat each other.

Our scheme overcomes the drawbacks that the distributor will have to do potentially exponential multiple checks for distributing identifications and shadows to participants on the basis of Birkhoff interpolation. But our scheme needs more public values than the existing popular schemes.

Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship and/or publication of this article: This work is supported by the National Natural Science Foundation of China under Grant No.61873069.

ORCID iD
Hongliang Zhu https://orcid.org/0000-0003-2448-2027

References
1. Shamir A. How to share a secret. Commun ACM 1979; 22(11): 612–613.
2. Blakley GR. Safeguarding cryptographic keys. In: Managing requirements knowledge, international workshop on IEEE computer society, New York, 4–7 June 1979, pp.313–317. New York: IEEE.
3. Blundo C, De Santis A, Stinson DR, et al. Graph decompositions and secret sharing schemes. J Cryptol 1995; 8(1): 39–64.
4. Padro C and Sáez G. Secret sharing schemes with bipartite access structure. IEEE Trans Inf Theory 2000; 46(7): 2596–2604.
5. Tassa T and Dyn N. Multipartite secret sharing by bivariate interpolation. J Cryptol 2009; 22(2): 227–258.
6. Li Q, Mi J, Li W, et al. CNN-based malware variants detection method for internet of things. IEEE Internet Things J 2021; 8(23): 16946–16962.
7. Chen M, Liao X and Wu M. Pulseedit: editing physiological signals in facial videos for privacy protection. IEEE Trans Inf Forensics Secur 2022; 17: 457–471.
8. Hu J, Liao X, Wang W, et al. Detecting compressed deepfake videos in social networks using frame-temporality two-stream convolutional network. IEEE Trans Circuits Syst Video Technol 2021; 32(3): 1089–1102.
9. Simmons GJ. How to (really) share a secret. In: Conference on the theory and application of cryptography, Santa Barbara, CA, 21–25 August 1988, pp.390–448. New York: Springer.
10. Brickell EF. Some ideal secret sharing schemes. In: Workshop on the theory and application of cryptographic techniques, Houthalen, 10–13 April 1989, pp.468–475. Berlin: Springer.
11. Chien HY, Jan JK and Tseng YM. A practical (t, n) multi-secret sharing scheme. IEICE Trans Fundam Electron Commun Comput Sci 2000; 83(12): 2762–2765.
12. Tassa T. Hierarchical threshold secret sharing. J Cryptol 2007; 20(2): 237–264.
13. Farras O, Mart-Farré J and Padró C. Ideal multipartite secret sharing schemes. In: Annual international conference on the theory and applications of cryptographic techniques, Barcelona, 20–24 May 2007, pp.448–465. New York: Springer.
14. Farrás O, Mart-Farré J and Padró C. Ideal multipartite secret sharing schemes. J Cryptol 2012; 25(3): 434–463.
15. Farras O and Padró C. Extending brickell–davenport theorem to non-perfect secret sharing schemes. Des Codes Cryptogr 2015; 74(2): 495–510.
16. Chen Q, Tang C and Lin Z. Efficient explicit constructions of compartmented secret sharing schemes. Des Codes Cryptogr 2019; 87(12): 2913–2940.
17. Gabidulin EM. Theory of codes with maximum rank distance. Probl Peredachi Inform 1985; 21(1): 3–16.
18. Chen Q, Tang C and Lin Z. Efficient explicit constructions of multipartite secret sharing schemes. IEEE Trans Inf Theory 2021; 68(1): 601–631.
19. Traverso G, Demirel D and Buchmann J. Dynamic and verifiable hierarchical secret sharing. In: International conference on information theoretic security, Tacoma, WA, 9–12 August 2016, pp.24–43. New York: Springer.
20. Xu G, Yuan J, Xu G, et al. An efficient compartmented secret sharing scheme based on linear homogeneous recurrence relations. Secur Commun Netw 2021; 2021: 5566179.
21. Yuan J, Yang J, Wang C, et al. A new efficient hierarchical multi-secret sharing scheme based on linear homogeneous recurrence relations. Inform Sci 2022; 592: 36–49.
22. Xu G, Yuan J, Xu G, et al. A new multi-stage secret sharing scheme for hierarchical access structure with existential quantifier. Inf Technol Control 2021; 50(2): 236–246.
23. Richard B. Introductory combinatorics. 5th ed. Beijing, China: China Machine Press, 2009, p.258.
24. Dehkordi MH and Mashhadi S. New efficient and practical verifiable multi-secret sharing schemes. Inform Sci 2008; 178(9): 2262–2274.
25. Yuan J and Li L. A fully dynamic secret sharing scheme. Inform Sci 2019; 496: 42–52.