Hindawi
Security and Communication Networks
Volume 2021, Article ID 1721925, 12 pages
https://doi.org/10.1155/2021/1721925

Research Article

Secret Image-Sharing Scheme Based on Multinode Authentication in the Internet of Things

Lina Zhang,[1,2] **Tong Wang**,[1] **Xiangqin Zheng**,[1] **Junhan Yang**,[1,2] **and Liping Lv**[3]

[1] *Xi'an University of Science and Technology, Xi'an 710054, China*
[2] *Shaanxi Normal University, Xi'an 710119, China*
[3] *College of Information Engineering, Zhengzhou Shengda University, Zhengzhou 451191, China*

Correspondence should be addressed to Lina Zhang; zhangln@xust.edu.cn

Received 5 June 2021; Revised 6 July 2021; Accepted 13 October 2021; Published 12 November 2021

Academic Editor: Qingqi Pei

Copyright © 2021 Lina Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Internet of things (IoT) has been developed and applied rapidly because of its huge commercial value in recent years. However, security problem has become a key factor restricting the development of IoT. The nodes of IoT are easy to be impersonated or replaced when attacked, which leads to the mistake of the uploaded data, the abnormal use of the application, and so on. Identifying the authenticity of the data submitted by the nodes is the top priority. We propose a scheme to verify the authenticity of multinode data. In this scheme, the authenticity of node data is checked through visual secret recovery and XOR operation together. The least significant bit (lsb) operation converts data from nodes into a bit, which improves the efficiency of data verification and reduces the risk of data leakage. This scheme achieves the purpose of verifying the data provided by the node, which avoids malicious attacks from illegal nodes. By analyzing the experiment result and comparing with other works, our scheme has the advantages of high verification efficiency, lightweight storage of nodes, and security verification.

1. Introduction

IOT (Internet of things) is a network based on information carriers such as the Internet and traditional telecommunications networks, allowing all ordinary physical objects that can be independently addressed to achieve interconnection [1]. It is an important part of a new generation of information technology, which has developed rapidly in recent years and has broad application prospects. With the upgrade of communication networks and the continuous development of IoT technology, its related services and related technologies [2–3] have been rapidly developed and popularized in all walks of life [4–8]. In IoT-related business, the label technology for identifying objects with unique identifiers has also received attention and a lot of research. Although IoT adopts the form of connecting things, it must rely on Internet technologies such as computer communication and information transmission. Therefore, in the IoT environment, there are security risks such as technical security issues, signal interference, malicious intrusion, and communication. In recent years, as an information carrier, images are widely used in various fields. Especially, in the related applications of IoT, image security is particularly important.

The sensor node is an important part of IoT [9–11], mainly responsible for information collection, data transmission, and data fusion. Its important function has attracted the attention of criminals; physical packet capture and brute force cracking have become important ways of attacking the IoT. Once a node in the IoT is cracked, the attacker has a legal identity and can attack inside the IoT, pouring massive amounts of redundant data and causing network congestion. From this perspective, malicious nodes take a great threat for the IoT.

In scheme [12], an integrated approach for authentication and access control is presented for communication with wireless sensor nodes in IoT networks, which provides strong protection against known attacks such as energy exhausting and Man-In-The-Middle. Li et al. [13] utilize blockchain technology, which serves as a secure tamper-proof distributed ledger to IoT devices. The processing method is to assign a unique ID for each individual device, recording them into the blockchain so that they can authenticate each other without a central authority. Lau et al. [14] use blockchain technology to authenticate IoT devices before it joins an IoT network. Based on the characteristics of blockchain, this method can be used to create the digital identification of IoT devices and authenticate them. However, the abovementioned schemes only verify the node and do not verify data stored by the nodes, so the attack of data tampering cannot be completely avoided.

At present, the idea of protecting image security is to use secret sharing; secret shares can be stored in sensing nodes. In order to restore the original secret completely and effectively, we must check the legitimacy of shares from nodes. For this reason, we propose a nodes' authentication in the IoT, which can reduce the storage space of every node and complete mutual authentication between nodes.

Image security prevents the adversary from getting any information about the original image in transmission or storage. Encryption algorithm, information hiding, and secret sharing have been used in the field of image security [15–17]. Encryption [18] is using mature cryptographic algorithms to process digital images. However, the processed image is very different from the normal image, which may be attacked by adversary. Information hiding technology [19] is that it hides secret information in an image carrier. If the carrier is processed during transmission, the origin secret information can still be recovered with a low failure rate. However, if the image carrier is damaged or lost, secret information cannot be recovered. Fortunately, secret sharing plays an important role in preventing carriers from being lost, damaged, maliciously destroyed, or operated by criminals. Secret sharing is processing the secret image into multiple shares, and multiple participants commonly save the shares. The secret image can be reconstructed when the number of participants comes to the threshold. Even if part of the share destroyed or lost, the secret image still can be reconstructed. Obviously, secret sharing can solve the shortcomings of encryption algorithms and information hiding technologies. With the deepening of research, many secret-sharing schemes [20–23] have been designed and improved.

Schemes [20–23] do not mention the share authentication. Many applications need identity authentication ability for program security, such as online banking [24], electronic voting [25], and e-commerce [26]. The verification-type secret-sharing scheme is that the processor processes the secret into several shares, and participants can verify the received shares. Therefore, the risk of the original secret and shares being leaked can be avoided. Chor [27] first proposes the concept of verification-type secret sharing. The verification work is mainly done through real-time webcasting. With the secret image-sharing scheme (SISS) widely used, share authentication has become especially important. Stadler [28] proposed an open verification-type secret-sharing scheme. Each participant in the scheme can verify the authenticity of the share; in addition, it will not cause any leakage of the original secret and share information. Unfortunately, the algorithm designed by Stadler cannot satisfy the real program requirements.

In addition, there are many new verification-type schemes [22, 29–33]. Feldman [29] proposes a SISS. The method of this scheme is that the third party compares the submitted data with the original data and completes the purpose of authenticating the data. If the submitted data are found to be false, the secret recovery work will be stopped. Unfortunately, this scheme requires a large amount of information to be disclosed in advance, and there is a risk of information leakage. The scheme [30] divides the secret image into nonoverlapping L blocks with containing $2k-2$ pixels. The processor constructs two $k-1$ degree polynomials for each block, calculating shares by relying on these two polynomials. In the verification phase, the processor checks the reconstructed two polynomials. If the same common integer exists between the two polynomials, the group of shares is correct; otherwise, the group of shares is forged. In fact, the scheme has some drawbacks, such as it cannot accurately screen out the forged shares and the reconstructed image is lossy. The scheme [22] is based on the symmetry of bivariate polynomial and the linearity of interpolation polynomial. However, this scheme has the disadvantages of lots of data calculation and low verification efficiency, and the verification accuracy is low.

This paper focuses on nodes' verification problem with/without a third party. The key of scheme is the combination of SISS and visual secret-sharing (VSS), which cleverly realizes nodes' authentication under different occasions. The scheme involves two types of images: public binary image that is used as an authentication password image and secret image is shared as secret information. The scheme has the advantages of no pixel expansion, lossless recovery, and high certification accuracy. In addition, this scheme can reduce the amount of calculation to save storage space. It allows users to choose a model in the scheme according to their needs, which effectively increases the flexibility and practicability of the scheme.

The rest of this article is organized as follows. Section 2 introduces some preliminaries. Section 3 describes the motivation and contribution of our proposed schemes. Section 4 proposes a share verifiable image secret-sharing scheme. Section 5 analyzes the correctness and security of proposed scheme and compares with related work. Section 6 is a conclusion of the article.

2. Preliminaries

In this section, we represent many related concepts and schemes. Shamir's secret-sharing scheme [34] based on Lagrange polynomials and the visual secret-sharing scheme based on random grids [35, 36] are introduced in this section. We combine these two schemes to achieve the purpose of share verification. In addition, we introduced related symbols that can be used in the scheme.

2.1. Shamir's Secret-Sharing Scheme. In scheme [34], there are n shareholders $U = \{U_1, U_2, \ldots, U_n\}$ and a mutually trusted dealer D. The scheme consists of two algorithms: share generation and secret reconstruction.

Share generation: the dealer D selects a prime number p and random a degree polynomial $f(x) = a_0 + a_1x^1 + a_2x^2 + \cdots + a_{k-1}x^{k-1}\text{mod}p$, where all coefficients a_i, $i = 1, 2, \ldots, k-1$, are in z_p. The secret S is equal to the constant term of $f(x)$, i.e., $S = f(0) = a_0$; D picks n different positive integers $x_1, x_2, \ldots, x_n$ from z_p and computes $S_i = f(x_i)$ for $i = 1, 2, \ldots, n$. Finally, D distributes each share S_i to the corresponding shareholder U_i securely and x_i is public information associated with U_i.

Secret reconstruction: suppose that $m(m \geq t)$ shareholders $U_1, U_2, \ldots, U_m$ team up for secret reconstruction. Each shareholder U_i sends his private share S_i to the other shareholders. After that, a shareholder has m shares $S_1, S_2, \ldots, S_m$, and it can use Lagrange interpolation polynomial to evaluate the secret S as

$$\begin{aligned} S &= f(0) \\ &= \sum_{i=1}^{m} S_i \prod_{j=1, j\neq i}^{m} \frac{x_j}{x_j - x_i} \text{mod}p. \end{aligned} \tag{1}$$

2.2. Random Grids (RG)-Based Visual Secret-Sharing (VSS) Scheme. RG-VSS [35] is a probabilistic visual secret-sharing scheme. $(2, 2)$ threshold RG-VSS is generally divided into two stages: the sharing stage and the recovery stage. At the same time, the black and white in the secret image are represented by 1 and 0, respectively.

2.2.1. Secret Generation. Pseudorandom generation s_1c_1 calculated s_1c_2 by

$$s_1c_2(h, w) = \begin{cases} s_1c_2(h, w), & \text{if } (s_1(h, w)) = 0, \\ \overline{s_1c_2(h, w)}, & \text{if } (s_1(h, w)) = 1, \end{cases} \tag{2}$$

where $s_1(h, w)$ is the position of each pixel in the binary image s_1, h represents the column coordinate of s_1, w is that the row coordinate of s_1, and $\overline{s_1c_1(h, w)}$ denotes the negation of $s_1c_1(h, w)$.

2.2.2. Secret Reconstruction. In equation (3), $s_1(h, w) = s_1c_1(h, w) \otimes s_1c_2(h, w)$. If $s_1(h, w) = 1$, the reconstruction result $s_1c_1(h, w) \otimes s_1c_2 = (h, w) = s_1c_1(h, w) \otimes \overline{s_1c_1(h, w)} = 1$ is determined to be black. If secret pixel is $s_1 = s_1(h, w) = 0$, then the restored result $s_1c_1 \otimes s_1c_2 = s_1c_1(h, w) \otimes s_1c_1(h, w)$ has a 50% chance to be black or white. Because s_1c_1 is pseudorandomly generated,

$$\begin{aligned} s_1(h, w) &= s_1c_1(h, w) \otimes s_1c_2(h, w) \\ &= \begin{cases} s_1c_2(h, w), & \text{if } (s_1(h, w)) = 0, \\ \overline{s_1c_2(h, w)}, & \text{if } (s_1(h, w)) = 1, \end{cases} \end{aligned} \tag{3}$$

where the symbol $\otimes$ represents the meaning of stacking; other symbols have the same meaning as in equation (2).

In fact, equation (2) can be seen as $s_1(h, w) = s_1c_1 \oplus s_1c_2(h, w)$; thus, we can recover $s_1(h, w)$ in this way. $\oplus$ expresses the exclusive XOR operation.

2.3. (K, N) VSS-Based Random Grids (RG). The scheme [36] proposes a (k, n) VSS scheme based on random grids. The sharing stage of the algorithm is as follows:

Input: A $M \times N$ binary secret image S and a pair of threshold parameters (k, n).

Output: n shadows $SC_i, i = 1, 2, \ldots, n$.

Step 1: for each position $(h, w) \in \{(h, w) | 1 \leq h \leq M, 1 \leq w \leq N\}$, repeat Steps 2 to 4

Step 2: sequentially calculate $b_1, b_2, \ldots, b_k$ repeatedly using equation (2), where b_x is the provisional pixels, $x = 1, 2, \ldots, n$

Step 3: set $b_{k+1} = b_1, b_{k+2} = b_2, \ldots, b_{2k} = b_k, b_{2k+1} = b_1, \ldots$ if $(n \,\text{mod}\, k) = 0$, $b_n = b_k$, else $b_n = b_{(n\,\text{mod}k)}$

Step 4: arrange $b_1, b_2, \ldots, b_n$ to $SC_1, SC_2, \ldots, SC_n$

Step 5: output n shadows $SC_1, SC_2, \ldots, SC_n$

2.4. Related Symbols. In this part, we give a table of summarizing the main used symbols in this paper for easy reading. They are shown in Table 1.

3. Motivation and Contributions

In our life, digital images are used widely, such as copyrighted pictures and QR codes. At this stage, how to ensure the correctness and completeness of the digital image has become very important. In order to solve problems mentioned, we use secret image sharing to solve image security issues, which can more effectively guarantee the integrity and correctness of the image. We propose a scheme combining the traditional SISS with visual secret sharing. It can complete the verification of the share verification work in a visual way. At the same time, considering the interactive and noninteractive protocols, we design two types of algorithms that satisfy real-life scenarios and the needs of different users.

The strategy is that nodes in the Internet of things are regarded as participants in our scheme. The shares of participants are considered nodes' data. In the data fusion stage, verifying data submitted by the participants to ensure that the final result is correct.

We have four pictures to explain the purpose of our scheme. The scheme involves two roles: participants and a third party. In order to allow readers to better understand the application scenarios of the scheme, assume that there is only one dishonest participant in the scheme to carry out the attack, and all cases are based on $(3, 4)$ threshold. In addition, we also analyze the processing ideas when there are multiple dishonest participants.

For Case 1: $(3, 4)$ secret sharing with a third party when there is no dishonest participant, in Figure 1, the third party can recover the original secret by obtaining the shadows of any three participants.

Table 1: Related symbols.

Number	Symbols	Description
1	S_2	Secret image
2	S_1	Authentication image
3	p	Divisible parameters
4	lsb (x)	Least significant bit processing
5	$\otimes$	Stacking operation
6	$\oplus$	XOR operation
7	W	Image width
8	H	Image height
9	$\lfloor\rfloor$	Round down
10	s_2c_i	Shares generated by S_2
11	s_1c_i	Shadows of S_1
12	thir	Share of third party
13	block$_i$	Subimage of secret image
14	$a \bmod p$	The remainder of a divided by p

Figure 2 describes Case 2: $(3,4)$ secret sharing with a third party when there is a dishonest participant. Any three participants send their shadows to the third party; if the third party finds a false share, it stops recovering the secret and broadcast dishonest participants to others.

For Case 3: $(3,4)$secret-sharing scheme without a third party when there is not fake participant, in figure 3, any three participants send their private shares to other participants and authenticate each other. Actually, each participant can receive two shares from others if all the shares are verified to be correct, and the original secret can be recovered.

For Case 4: $(3,4)$ threshold secret sharing without a third party when there is a fake participant, in Figure 4, for dishonest participant P_1, participant P_2, and participant P_3, they send their private shares to each other, and then, each participant verifies the shares of the other two participants. P_2 and P_3 detected that the fake share is sent by P_1, then stopped refactoring the secret, and broadcasted the fake behavior of P_1 to P_4.

The above cases only describe the existence of one forger, which should be done if there are multiple forgers.

The third-party verification model is still feasible; however, the model of mutual authentication between participants must rely on the voting mechanism.

Suppose that k participants complete the secret reconstruction work. Each participant authenticates other $k-1$ participant's shadow and votes for them. If any participant gains $k-1$ votes (note: dishonest participants are afraid of revealing their identities and give up voting opportunities), we determine that the participants in the group are honest. If any participant gains less than $k-1$ votes, we let the remaining $n-k$ participants perform auxiliary verification. The remaining participants vote for the k participants. If any of the k participants gains less than $T = \lfloor((n-1)/2)\rfloor$ votes from $n-1$ participants, we determine that this participant is dishonest. In contrast, if the number of votes more than $T = \lfloor((n-1)/2)\rfloor$, we judge that if more than half of the T votes are true; the participant is considered to be honest; otherwise, the participant is dishonest. Therefore, the scheme stipulates more than $T = \lfloor(n/2)\rfloor + 1$ honest participants among all n participants to achieve the threshold.

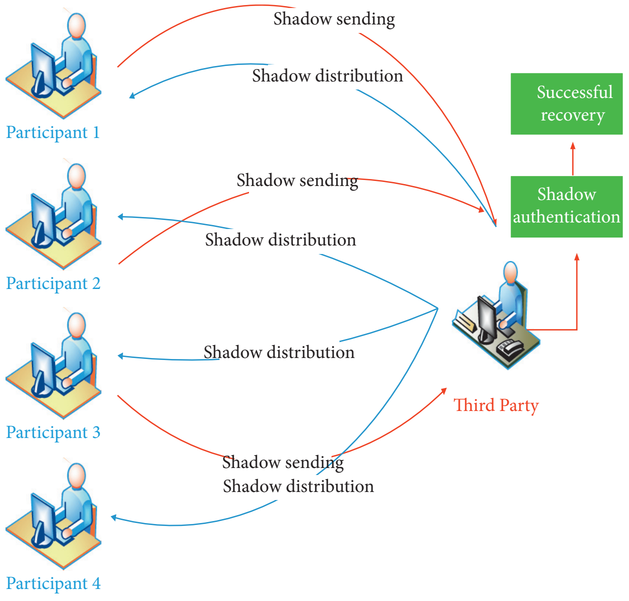

Figure 1: Case 1: $(3,4)$ threshold SISS with a third party.

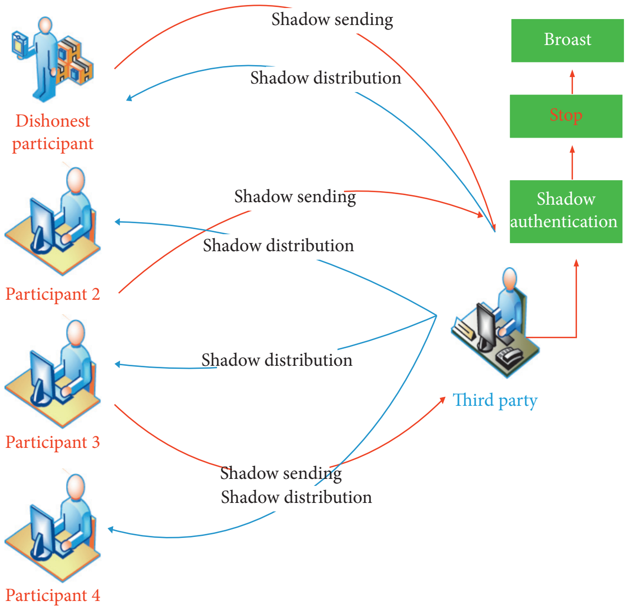

Figure 2: Case 2: $(3,4)$ threshold SISS with a third party when there is a fake participant.

4. Secret Image-Sharing Scheme with Shares' Verifiable

In this section, we describe the specific algorithms for the two types of models. The first model is designed for batch verification. In the secret-sharing phase, we first process the n shares generated by the secret image S_2. Then, let the n shadows generated by the authentication image S_1 match the processed n shares. If the matching is unsuccessful, let S_1 regenerate new shadows with the help of the (k,n) VSS scheme and perform the matching again until it succeeds. After the match is successful, one of the n shadows is selected

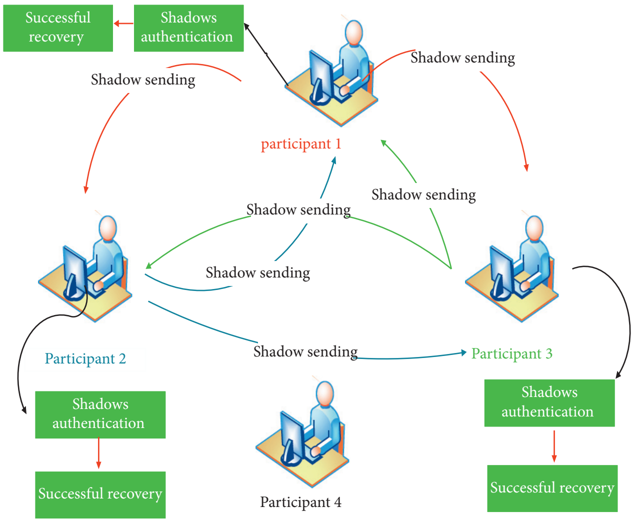

Figure 3: Case 3: $(3, 4)$ threshold SISS without a third party when there is no fake participant.

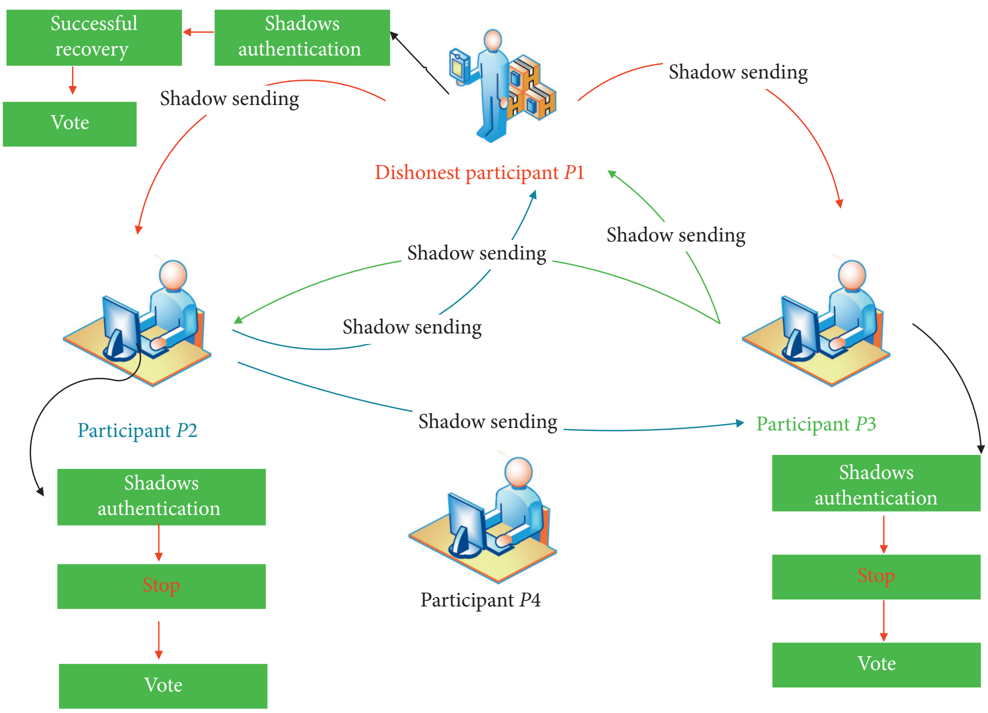

Figure 4: Case 4: $(3, 4)$ threshold SISS without a third party when there is a fake participant.

and sent to the third party (since the (k, n) VSS scheme can only generate k kinds of shadows, and n is an integer multiple of k), denoted as thir. Assuming that we want to know quickly whether there is a dishonest participant among a group of participants, the third party uses the private shadow thir and the submitted share to calculate for judgment. If equation lsb$(s_2c_1)\oplus, \ldots, \oplus$ lsb $(s_2c_i)\oplus$thir$\triangleq\oslash$ for $i \in [i, k]$ $(s_2c_i$ represents the submitted shares) holds, there is no dishonest participant in this group; on the contrary, there is a dishonest participant.

The second model is that participants verify each other, which can screen out dishonest participants. Each participant in a group sends private share to other $k - 1$ participants. Next, participants use their own computing power to judge the received shares and then vote on them. Finally, judging dishonesty by voting results, the specific voting mechanism has been specified in the third part, so we will not describe it in detail here.

The specific steps of the two models have been described in Algorithm 1 and Algorithm 2 as follows.

4.1. Algorithm 1

Input: a secret image S_2 with $H \times W$, a authentication image S_1 with $(H \times W)/k$, and the threshold parameters (k, n), $2 \leqq k \leqq \lfloor n/2 \rfloor$.

Output: shadows $s_2c_1, s_2c_2, \ldots, s_2c_n$ and a binary authentication shadow thir.

Step 1: a prime number $p = 257$. Dividing S_2 into L nonoverlapping subimages of size is $H \times W/k$, and subimages are expressed as $\text{block}_1, \text{block}_2, \ldots, \text{block}_L$.

Step 2: construct a polynomial of degree $k - 1$, which is shown as

$$f(x) = \left(a_0 + a_1x^1 + \cdots + a_{k-1}x^{k-1}\right)\text{modp}. \tag{4}$$

Then, calculate $s_2c_i = f(x)$, $i \in [1, n]$, where the coefficient $a_0, a_1, \ldots, a_{k-1}$ is taken from block_i, $i \in [1, L]$.

Step 3: utilize (k, n) RG-VSS to split $S_1(h, w)$ to n temporary bits and denote as $b_1, b_2, \ldots, b_n$. Assign $b_1, b_2, \ldots, b_n$ to $s_1c_1, s_1c_2, \ldots, s_1c_n$.

Note: $1 \leq h \leq H, 1 \leq w \leq (W/k)$.

Step 4: define a sequence $\text{seq} = \{1, 2, \ldots, n\}$, scrambling seq to generate an new sequence $\text{seq}_1 = \{u_1, u_2, \ldots, u_n\}$. Going to execute Step 5, otherwise execute Step3.

Where lsb (x) means that it gets the least significant bit of x.

Step 5: if a third party exists, go to execute step 6, otherwise execute Step 7.

Step 6: randomly pick a numbers from $\{b_1, b_2, \ldots, b_k\}$, denoted as thir. thir is assigned to the third party.

Step 7: assign specify shares s_2c_j, $j = u_1, u_2, \ldots, u_n$, to s_2c_i, $i \in [1, n]$.

For Algorithm 1,

(1) We set a prime number $p = 257$ to guarantee the value of the shadows pixel is within [0, 255] and lossless recovery by effective shares.

(2) use polynomials to ensure that there is no pixel expansion.

(3) Step 4 and Step 5 cooperate to achieve shares' authentication when a third party exists.

(4) have restrictions on the relationship between k and n, in which suggestion is $n/k \leq 3/5$.

4.2. Algorithm 2

Input: obtain k shares from s_2c_i, $i \in [1, n]$, the public password binary image S_1, and verification share thir held by the third party.

Output: recovered secret image S_2.

Step 1: if the third party verifies shares, firstly, calculate the result of the K shares, and they are expressed as $\text{lsc}_j = \text{lsb}\ (s_2c_i)$, $j \in [1, k]$, $i \in [1, n]$. Then, through the stacking or XORing operation the number of lsc_j and thir, the result is recorded as S_1^*. If S_1^* is recognized as S_1 by HVS or $S_1^* = S_1$, the shares are valid, and go to Step 3. Otherwise, there is forgery among the k participants, and it broadcasts the dishonest person's message to other participants in the group.

Step 2: if participants authenticate each other, each participant sends his share s_2c_i to other $k - 1$ participants. All participants process received $k - 1$ shares; the processing result is expressed as $\text{lsc}_j = \text{lsb}\ (s_2c_i)$, $j \in [1, k]$, $i \in [1, n]$. Next, each participant relies on stacking or XOR to complete the processing of all lsc_j, $j \in [1, k - 1]$ and own share s_2c_i, $i \in [1, n]$. The processing result is recorded as S_1^*. If S_1^* is recognized as S_1 by HVS or $S_1^* = S_1$, the k participants are all honest and go to Step 3; Otherwise, use voting mechanism to complete shares' authentication.

Step 3: recovering all subimages block_i, $i \in [1, L]$, repeat Step 4 to Step 5.

Step 4: constructing the interpolation polynomial (5) through valid shares,

$$\begin{aligned} f(i_1) &= \left(a_0 + a_1 i_1^1 + \cdots + a_{k-1} i_1^{k-1}\right)\text{modp}, \\ f(i_2) &= \left(a_0 + a_1 i_2^1 + \cdots + a_{k-1} i_2^{k-1}\right)\text{modp}, \\ &\vdots \\ f(i_k) &= \left(a_0 + a i_k^1 + \cdots + a_{k-1} i_k^{k-1}\right)\text{modp}, \end{aligned} \tag{5}$$

where $1 \leq i \leq L$.

Step 5: computing $a_0, a_1, \ldots, a_{k-1}$.

Step 6: outputting reconstruction secrets S_2.

For Algorithm 2,

(1) In Step 2, every participant authenticates other $k - 1$ participants here. If k number of S_1^* is recognized as S_1 by HVS or $S_1^* = S_1$, the authentication result is true, and go to Step 3.

(2) The authentication ways of the two models are different. When participants authenticate each other, the voting mechanism will be carried out.

5. Experimental Results and Analysis

In this part, we will give some experiments to verify the feasibility and effectiveness of the proposed scheme. In addition, we also analyze and compare our scheme with other schemes in detail.

5.1. Experimental Results. The operating environment required for this experiment is as follows: Windows10, CPU

(2.60 GHz inter(R) i7-9750H CPU), 64G RAM, and Matlab application.

Due to the characteristics of no pixel expansion of the proposed SIS, the size of the secret image is 132×132 in our experiments. Here, we introduce the experimental results of $(4, 8)$ threshold.

Figure 5 shows the experimental results of different verification modes. As we can see all, the secret S_2 is (a) in Figure 5, the authentication image S_1 is (b) in Figure 5, and picture $(c)-(j)$ denotes the output shares $s_2c_1, s_2c_2, s_2c_3, s_2c_4, s_2c_5, s_2c_6, s_2c_7, s_2c_8$. Result of $s_2c_1, s_2c_2, s_2c_3, s_2c_4, s_2c_5, s_2c_6, s_2c_7, s_2c_8$ processed by lsb is displayed in pictures $(k)-(r)$.

Figure 6 of $(a)-(h)$ represent many shadows $s_1c_1, s_1c_2, s_1c_3, s_1c_4, s_1c_5, s_1c_6, s_1c_7, s_1c_8$ produced by S_1, satisfying $s_1c_i(h, w) = \text{lsb } s_2c_j$, $i \in [1, 8]$, $j \in [u_1, u_8]$, and s_1c_4 is sent to the third party as a verification password. Picture (i) illustrates a fake share which is denoted by wro. Picture $(j)-(m)$ denotes results through the XORing operations of S_1 and $s_2c_1, s_2c_2, s_2c_3, s_2c_4$; finally, the recovered result can be well recognized. Thus, the shares $s_2c_1, s_2c_2, s_2c_3, s_2c_4$ have passed the third-party verification; shares $s_2c_1, s_2c_2, s_2c_3, s_2c_4$ can be used in secret reconstruction work.

Supposing there is a dishonest participant, which is shown in Figure 6 of (i) is wro and shares s_2c_1, s_2c_2, s_2c_3 want to join the secret S_2 reconstruction work, the third party verifies the provided shares s_2c_1, s_2c_2, s_2c_3, wro. The result is presented in Figure 6 of (n). So, there are faked shares in this group. The third party stops the recovery of the secret S_2 and looks for new effective shares.

Each participant acts as a restructioner; they will verify the received shares $k-1$. Suppose the share s_2c_1 saved by participant P_1, the share s_2c_2 saved by participant P_2, the share s_2c_3 saved by participant P_3, and the share s_2c_4 saved by participant P_4. P_1, P_2, P_3, P_4 verify the received $k-1$ shares; results are shown in figure of $(0)-(r)$. So, shares provided by the participants P_1, P_2, P_3, P_4 are valid. If P_1 provided a forged, we use the voting mechanism to vote for P_1, P_2, P_3, P_4 and finally screen out dishonest participants P_1.

5.2. Safety and Correctness Analysis. We analyze the security and correctness of the proposed scheme. Note that the gray secret image S_2 and binary password image S_1 are not related in the scheme. In addition, the obtained k restored pixel values are expressed as s_2c_i, $i \in [1, k]$; the third party is expressed as sk and holds data ths. The attacker is represented as ak.

Lemma 1. *Since the pixel value in the gray secret image S_2 is limited to [0,255], the shadows s_2c_i, $i \in [1, k]$ required to be generated are limited to [0,255].*

Proof. Because the pixel value of the secret image is S_2 limited in[0,255], in the S_2 sharing stage, the generated shadows s_2c_i must satisfy $s_2c_i < p-1$, $i \in [1, n]$, $p = 257$. Therefore, the pixel of sharing shadows s_2c_i is limited in [0,255]. □

Theorem 1. *When the data saved by the node is attacked, a single data cannot reveal any information of S_2.*

Proof. Since the scheme in this paper is based on Lagrange secret sharing. S_2 is shared into n data that are stored by nodes; these data do not carry any information of S_2. At the same time, the security features are derived from the threshold, and only k valid data can recover the secret S_2. □

Theorem 2. *The two types' modes in the scheme can realize the legitimacy detection of node data, ensuring that the recovered secret S_2 is correct.*

Proof. The scheme stipulates that more than 51% of the node data will not be attacked, and the third party is credible. When the third party verifies the data, avoid sk colluding with ak to tamper with the verification result of data. When participants authenticate each other and when the voting mechanism is used to determine the accuracy of the data, since the proportion of the attacked data is set, malicious participants can be prevented from jointly affecting the accuracy of the data verification result. Therefore, both verification modes are safe during data verification. □

Theorem 3. *The secret image S_2 can be restored lossless through k shadows s_2c_i, $i \in [1, k]$.*

Proof. In the recovery phase, a_0 and a_i, $i \in [1, k-1]$ can be calculated by the Lagrange interpolation formula. Because Lemma 1 has proved that all shadows $s_2c_i < p-1$; finally, the secret reconstructed by shadows is also lossless. □

Theorem 4. *Both the third-party verification model and the participant's mutual verification model can run correctly, completing the task of efficiently and accurately screening forged data.*

Proof. In the first mode, the verification key is held by sk; in addition, sk is honest in the scheme. During the verification phase, we just need to judge whether lsb $(s_2c_i) \oplus \text{ths} \neq \oslash$, $i \in [1, k]$ or lsb $(s_2c_i) \oplus \text{ths} = \oslash$. If the final result is not $\oslash$, the data are judged to be forged. In the second mode, each participant submits data to other $k-1$ people, and the participants vote for each other. The algorithm stipulates that dishonest persons cannot vote, and there are more than 51% honest persons among n participants. The principle of the voting mechanism is that the minority obeys the majority. We record the number of votes among the k participants, and the number of votes is equal $k-1$; we judge the participant to be honest; otherwise, let the remaining $n-k$ participants help with authentication. Finally, by determining the proportion of all votes that is true for each participant, participants who have true votes more than half of the total votes are considered honest. □

5.3. Experimental Example. Let us take the threshold (4,8) as an example. Supposing the secret image S_2 is composed of

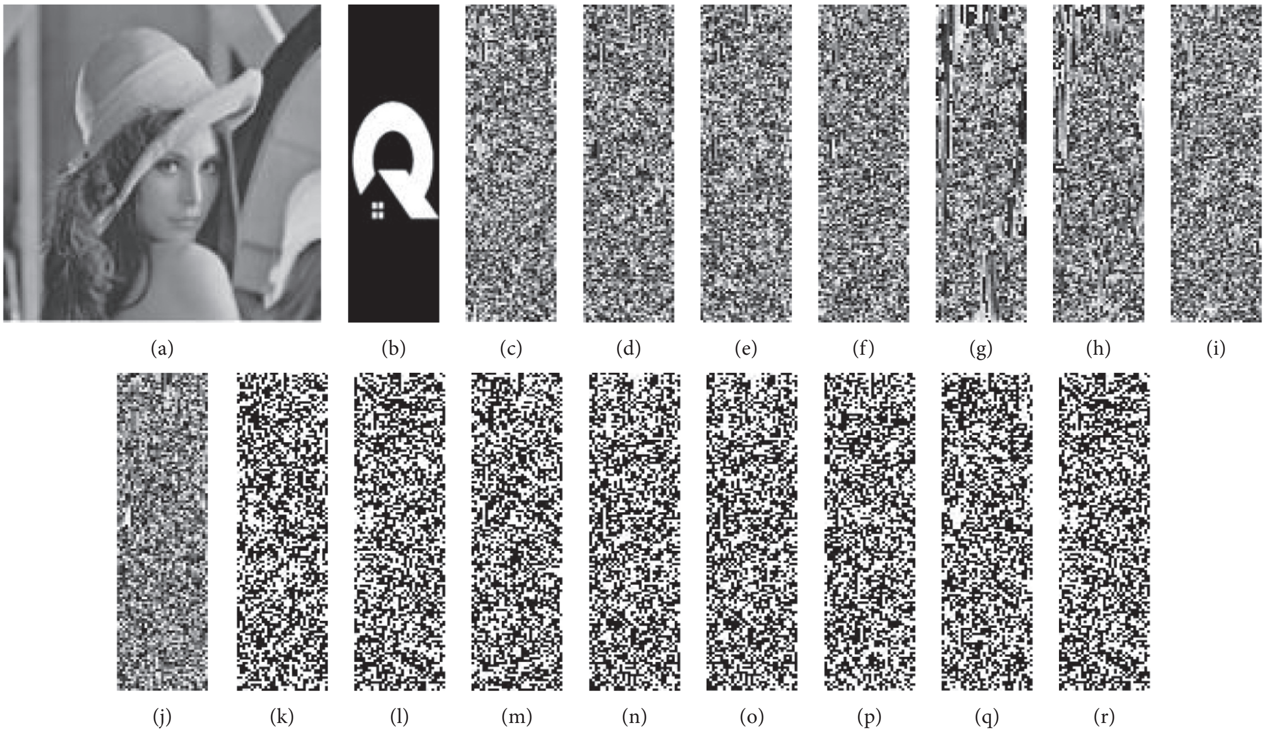

FIGURE 5: S_2 generated shares and the processing results of all shares. (a) S_2. (b) S_1. (c) s_2c_1. (d) s_2c_2 (e) s_2c_3 (f) s_2c_4. (g) s_2c_5 (h) s_2c_6 (i) s_2c_7 (j) s_2c_8. (k) lsb(s_2c_1) (l) lsb(s_2c_2) (m) lsb(s_2c_3) (n) lsb(s_2c_4). (o) lsb(s_2c_5) (p) lsb(s_2c_6) (q) lsb(s_2c_7) (r) lsb(s_2c_8)

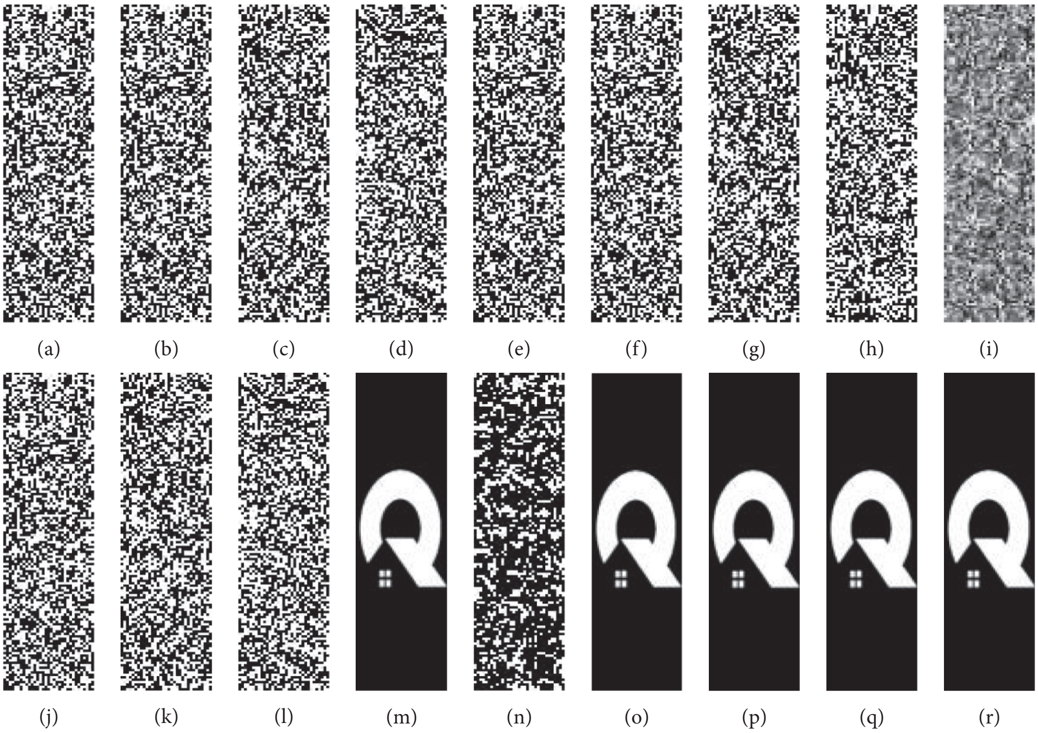

FIGURE 6: The shadow generated by S_1 and the verification result of the submitted shares. (a) s_1c_1. (b) s_1c_2 (c) s_1c_3 (d) s_1c_4. (e) s_1c_5 (f) s_1c_6 (g) s_1c_7 (h) s_1c_8. (i) wro (j) lsb (s_2c_1), (k) lsb (s_2c_1) ⊕ lsb (s_2c_2), (l) lsb (s_2c_1) ⊕ lsb (s_2c_2) ⊕ lsb (s_2c_3), (m) lsb (s_2c_1) ⊕ lsb (s_2c_2) ⊕ lsb (s_2c3) ⊕ lsb (s_2c_4), (n) lsb (s_2c_1) ⊕ lsb (s_2c_2) ⊕ lsb (s_2c_3) ⊕ lsb (wro), (o) lsb (s_2c_1) ⊕ lsb (s_2c_2) ⊕ lsb (s_2c_3) ⊕ lsb (s_2c_4), (p) lsb (s_2c_2) ⊕ lsb (s_2c_1) ⊕ lsb (s_2c_3) ⊕ lsb (s_2c_4), (q) lsb (s_2c_3) ⊕ lsb (s_2c_1) ⊕ lsb (s_2c_2) ⊕ lsb (s_2c_4), (r) lsb (s_2c_4) ⊕ lsb (s_2c_1) ⊕ lsb (s_2c_2) ⊕ lsb (s_2c_3).

block = $\{153, 154, 154, 152\}$, the authentication image S_1 is $S_1(h, w) = 0$.

In shares' generation phase, first of all, we construct the polynomial, which is $f(x) = 153 + 154x^1 + 154x^2 + 152x^3$. Next, the processor goes to calculate the eight shares; they are $s_2c_1 = f(1)$, $s_2c_2 = f(2)$, $s_3c_2 = f(3)$, $s_2c_4 = f(4)$, $s_2c_5 = f(5)$, $s_2c_6 = f(6)$, $s_2c_7 = f(7)$, and $s_2c_8 = f(8)$ and are generated by the secret image S_2. And, the processor uses (4,8) RG-VSS to generate shadows $s_1c_1(h, w) = 1$, $s_1c_2(h, w) = 0$, $s_1c_3(h, w) = 0$, $s_1c_4(h, w) = 1$, $s_1c_5(h, w) = 1$, $s_1c_6(h, w) = 0$, $s_1c_7(h, w) = 0$, and $s_1c_8(h, w) = 1$ from $S_1(h, w)$. Next, the dealer executes the most important step that determine whether requirement $s_1c_i(h, w) = \text{lsb}\ s_2c_j$, for $i \in [1, 8]$ and $j \in [u_1, u_8]$ is satisfied. If the result is not satisfied, the requirement re-executes (4,8) RG-VSS until the requirement is satisfied. If the result is a match, S_1 is assigned to the third party, assuming that the calculated result satisfies the requirement.

In the shares' authentication phase, when the third party is responsible for verifying and recovering secret image S_2, the third party uses the verification algorithm to calculate that lsb $(s_2c_1)\oplus$ lsb $(s_2c_2)\oplus$ lsb $(s_2c_3)\oplus$ lsb $(s_2c_4)\oplus S_1 = 1\oplus 0\oplus 0\oplus 1\oplus 0 = \oslash$, and the remaining shares, that is, lsb $(s_2c_5)\oplus$ lsb $(s_2c_6)\oplus$ lsb $(s_2c_7)\oplus$ lsb $(s_2c_8)\oplus S_1 = 1\oplus 0\oplus 0\oplus 1\oplus 0 = \oslash$, as we know that these shares $s_2c_1, s_2c_2, s_2c_3, s_2c_4, s_2c_5, s_2c_6, s_2c_7, s_2c_8$ are valid. If the result of the third-party calculation is not $\oslash$, it means that there are one or more invalid shares in the group. When participants authenticate each other, each participant will receive shares sent by 3 people. Participants verify the received shares and then vote for the owners of the shares. If each of the 4 participants gets 3 votes, the 4 participants are judged to be honest. If any of the 4 participants gets less than 3 votes, then the remaining 4 participants will be requested to give auxiliary verification. Finally, if any of the 4 participants gets less than 3 votes, the participant is considered dishonest. If 4 participants get more than 3 votes, we check whether the proportion of true in the total votes exceeds half. If it exceeds 50%, we judge that the participant is honest; otherwise, the participant is dishonest.

If the four shares ($s_2c_1 = 99$, $s_2c_2 = 237$, $s_2c_2 = 194$, $s_2c_2 = 111$) pass verification, they can be used for secret S_2 reconstruction work. Using equation (6) to calculate S_2 pixel,

$$\begin{aligned} f(1) &= \left(a_0 + a_1 + a_2 + a_3\right) \bmod 257, \\ f(2) &= \left(a_0 + a_1 2^1 + a_2 2^2 + a_3 2^3\right) \bmod 257, \\ f(3) &= \left(a_0 + a_1 3^1 + a_2 3^2 + a_3 3^3\right) \bmod 257, \\ f(4) &= \left(a_0 + a_1 4^1 + a_2 4^2 + a_3 4^3\right) \bmod 257, \end{aligned} \tag{6}$$

the result is $a_0 = 153$, $a_1 = 154$, $a_2 = 154$, and $a_3 = 152$ by Lagrange interpolation. Thus, the secret S_2 is successfully restored.

5.4. Comparisons with Relative Schemes. In this part, we compare the proposed scheme with related schemes [22, 30] from many aspects, showing the advantages of our proposed scheme. First of all, we discuss the size of share. In the secret-sharing scheme, the degree of the constructed polynomial depends on the threshold of the scheme. In addition, there are also different ways to select coefficients in polynomials. In our scheme, the coefficient values in the constructed polynomial all come from the secret image. Thus, the size of the generated share is $1/k$ times original secret, expressed as $|1/k \times (H \times W)|$. In the scheme [30], the scheme is dividing the original image into L nonoverlapping blocks, and each block contains $2k$ pixels. The processor constructs two polynomials of degree $k - 1$ for each block, and the coefficients of each polynomial all come from this block. In this way, the size of the share is $1/k$ times the original image, denoted as $|1/k - 1 \times (H \times W)|$. In the scheme [22], only one coefficient in the polynomial comes from the secret image, and the remaining $k - 1$ coefficients are obtained from the processor data so that the size of the share is the same as the original image, which is $|H \times W|$.

Analyzing the efficiency of share generation in the encryption phase, in our scheme, first, we perform lsb processing on the calculated share s_2c_i, and the processed results are matched with shadows from the authentication image S_1. Ideally, the matching can be done only once, and in the worst case, it takes times to complete. In the scheme [30], the first step is to divide the secret image into nonrepeated blocks B_i, $i \in [1, L]$, $B_i = [a_{i,0}, \ldots, a_{i,k-1}, b_{i,0}, \ldots, b_{i,k-1}]$, and then, the processor randomly selects an integer r_i to satisfy $r_i a_{i,0} + b_{i,0} = 0$ and $r_i a_{i,1} + b_{i,1} = 0$. Fortunately, it only needs one time; in the worst case, it needs n times. In scheme [22], the processor chooses a symmetric bivariate polynomial $F(x, y)$ of degree $k - 1$. The secret S_2 is hidden in the constant term by $F(x, y)$. In any case, encryption can be completed at one time.

We discuss the certification efficiency of the certification party. In scheme [22], the verifier holds the share size as $|H \times W|$, who only needs to use the private share to compare with the provided share. Here, we mark the authentication efficiency of the scheme is ef1. In our scheme, the third party determines whether the provided share is true or false through an exclusive OR operation. The share size saved by the third party is $|(H \times W)/k|$; therefore, the verification efficiency will be improved. Because our scheme has the same verification way as scheme [22], and the size of the share held by the verifier is reduced. Marking certification efficiency is $\text{ef}2 = k \times \text{ef}1$ in our scheme. In scheme, the size of the private share stored by authenticator is $H \times W/k - 1$; obviously, the verification efficiency in [30] also is improved, denoted as $\text{ef}3 = (k - 1) \times \text{ef}1$.

Table 2 shows the differences between our scheme and related schemes in three aspects: verification efficiency, encryption efficiency, and share size. Table 2 visually shows the performance comparison between our scheme and related schemes [22, 30] (note: here, we mark both the encryption efficiency and verification efficiency of Scheme [22] as 100%).

Table 3 covers four aspects. Readers can see differences between our scheme and related schemes more intuitively. Our scheme includes the advantages of simple share

Table 2: Performance comparison between our scheme and related schemes [22, 30].

Scheme	Our scheme	[30]	[22]
Verification efficiency	$k \times 100\%$	$(k-1) \times 100\%$	100%
Encryption efficiency	$k \times 100\%$	$(2k-2) \times 100\%$	100%
Share size	$\lvert H \times W/k \rvert$	$\lvert H \times W/k - 1 \rvert$	$\lvert H \times W \rvert$

Table 3: Comparison of our scheme, scheme [30], and scheme [22].

Scheme	Our scheme	[30]	[22]
Authentication operation	VCS (XOR/OR)	Common value	Common value
Pixel expansion	NO	NO	YES
Restructure	Lagrange	Lagrange	Lagrange
Reconstruction quality	Losses	Loss	Loss

Table 4: Security comparison.

Scheme	Our scheme	[30]	[22]	[12]	[13]	[14]
Information hiding	YES	YES	YES	NO	NO	NO
Decentralized	NO	NO	NO	YES	YES	YES
Channel security	YES	YES	NO	NO	NO	NO
Impersonation attack	YES	YES	YES	YES	YES	YES
Replay attack	YES	YES	YES	YES	YES	YES
Hierarchical access	YES	NO	NO	NO	NO	NO

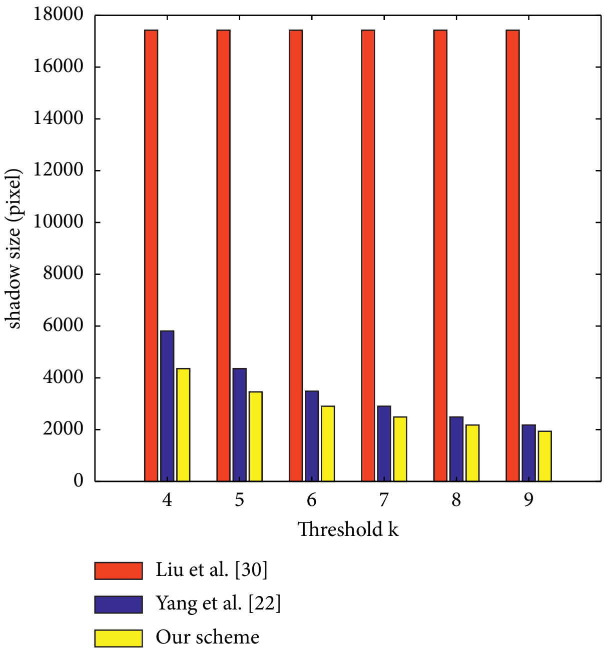

Figure 7: Bar graph of shadow size under different thresholds.

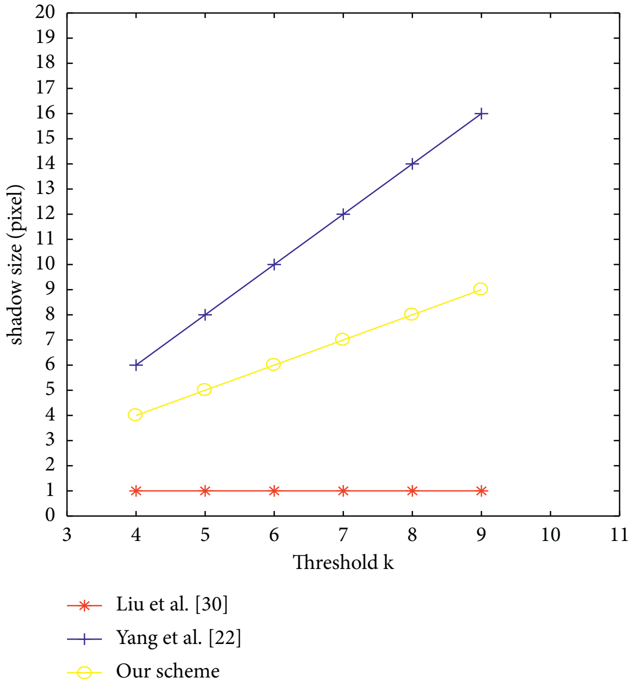

Figure 8: Line graph of shadow encryption efficiency under different thresholds.

authentication method, no pixel expansion, and lossless recovery of secret.

We compare related schemes [12, 14, 22, 30] with our scheme from multisecurity perspectives, as shown in Table 4.

We assume that the size of the secret image is 132 × 132 PX and calculate the specific size of the shadow of the related scheme according to the conclusion in Table 2. The bar chart shows result in Figure 7, the comparison of the shares' size in three schemes. Readers can see the difference more intuitively. We select the thresholds k are 4, 5, 6, 7, 8, and 9 and then accurately obtain the results of the corresponding schemes.

In our scheme, the coefficient values in the constructed polynomial all come from the secret image. Thus, the size of the generated shadow is $1/k$ times original secret. In the scheme [30], the scheme is dividing the original image into L nonoverlapping blocks, and each block contains $2k-2$ pixels. The processor constructs two polynomials of degree $k-1$ for each block, and the coefficients of each polynomial all come from this block. In this way, the size of the share is $1/k$ times the original image. In the scheme [22], only one coefficient in the polynomial comes from the secret image, and the remaining $k-1$ coefficients are obtained from the processor data so that the size of the share is the same as the original image, which is $|H \times W|$. In the final equivalent size secret analysis result, the encryption efficiency of our scheme is kf, scheme [30] is f (note: assume that the encryption efficiency of [30] is f), and [22] is $(k+1)f$. Figure 8 reflects the result. Here, we take the scheme [30] as a benchmark to better present the encryption effect of our scheme and scheme [22]. We adopt the thresholds k are 4, 5, 6, 7, 8, and 9, respectively. It achieves the purpose of readers to understand the comparison results of related schemes more quickly and effectively.

6. Conclusion

With the rapid development of network science and technology, virtual demand-based products are designed to meet people's daily convenience. At this stage, there are many applications based on IoT in the living environment, and most of them have become our daily necessities, such as intelligent transportation medical security and agriculture. We are eager to use these IoT applications without data attacks.

Our scheme satisfies the identity authentication function and the requirements of different application scenarios. The scheme can accurately screen out dishonest participants, so as to ensure that the final reconstruction result is correct. In addition, the scenarios where the scheme can be applied are online banking business processing, facial attendance electronic voting, and e-commerce. However, our scheme also has many shortcomings. The maximum threshold depends on the number of nodes, leading to a high correlation between the threshold and the number of nodes. If the number of tampered nodes exceeds 50%, the second pattern in the proposed scheme will not be able to complete the authentication work. These issues are what we will focus on in the next stage.

Data Availability

The data used to support the findings of this study are available from corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Shaanxi Provincial Department of Science and Technology Youth (project nos. 2021JQ-575 and 2021JQ-576), Shaanxi Provincial Department of Education (project no. 19JK0526), Yulin Science and Technology Bureau (project nos. 2016-24-4 and 2019–173), and National Natural Science Foundation of China (no. 62102309).

References

[1] M. Ammar, G. Russello, and B. Crispo, "Internet of things: a survey on the security of iot frameworks," *Journal of Information Security and Applications*, vol. 38, pp. 8–27, 2018.

[2] L. Yang, Z. Yu, M. A. E. Meligy, A. M. E. Sherbeeny, N. Wu, and N. Wu, "On multiplexity-aware influence spread in social networks," *IEEE Access*, vol. 8, Article ID 106705, 2020.

[3] X. Li, Z. Yu, Z. Li, and N. Wu, "Group consensus via pinning control for a class of heterogeneous multi-agent systems with input constraints," *Information Sciences*, vol. 542, pp. 247–262, 2021.

[4] S. A. Dalvi and M. Z. Shaikh, "Internet of things for smart cities," *Imperial journal of interdisciplinary research*, vol. 3, 2017.

[5] A. Li, X. Ye, and H. Ning, "Thing relation modeling in the internet of things," *IEEE Access*, vol. 5, Article ID 17117, 2017.

[6] S. Tanwar, P. Patel, K. Patel, S. Tyagi, N. Kumar, and M. S. Obaidat, "An advanced internet of thing based security alert system for smart home," in *Proceedings of the 2017 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pp. 25–29, IEEE, Dalian, China, July 2017.

[7] S. Zhou, Z. Yu, E. S. A. Nasr, H. A. Mahmoud, E. M. Awwad, and N. Wu, "Homomorphic encryption of supervisory control systems using automata," *IEEE Access*, vol. 8, Article ID 147185, 2020.

[8] E. Viciana, A. Alcayde, F. G. Montoya, R. Baños, F. M. C. Arrabal, and F. A. Manzano, "An open hardware design for internet of things power quality and energy saving solutions," *Sensors*, vol. 19, 2019.

[9] T. Nguyen Gia, V. K. Sarker, I. Tcarenko et al., "Energy efficient wearable sensor node for iot-based fall detection systems," *Microprocessors and Microsystems*, vol. 56, pp. 34–46, 2018.

[10] M. Kocakulak and I. Butun, "An overview of wireless sensor networks towards internet of things," in *Proceedings of the 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 1–6, IEEE, Las Vegas, NV, USA, January 2017.

[11] A. Raj and D. Steingart, "Review—power sources for the internet of things," *Journal of the Electrochemical Society*, vol. 165, 2018.

[12] M. Azarmehr, A. Ahmadi, and R. Rashidzadeh, "Secure authentication and access mechanism for iot wireless sensors," in *Proceedings of the 2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–4, IEEE, Baltimore, MD, USA, May 2017.

[13] D. Li, W. Peng, W. Deng, and F. Gai, "A blockchain-based authentication and security mechanism for iot," in *Proceedings of the 2018 27th International Conference on Computer Communication and Networks (ICCCN)*, vol. 1, no. 1–6, July 2018.

[14] C. Lau, A. Yeung, and F. Yan, "Blockchain-based authentication in iot networks," in *Proceedings of the 2018 IEEE Conference on Dependable and Secure Computing (DSC)*, pp. 1–8, IEEE, Kaohsiung, Taiwan, December 2018.

[15] R. Enayatifar, A. H. Abdullah, I. F. Isnin, A. Altameem, and M. Lee, "Image encryption using a synchronous permutation-

diffusion technique," *Optics and Lasers in Engineering*, vol. 90, pp. 146–154, 2017.

[16] S. Zhang, X. Li, Q. Li, and Q. Zhou, "Image information hiding method for jpeg data flow," in *Proceedings of the 2018 8th International Conference on Intelligent Systems, Modelling and Simulation (ISMS)*, pp. 67–71, IEEE, Kuala Lumpur, Malaysia, May 2018.

[17] X. Yan, Y. Lu, and L. Liu, "A general progressive secret image sharing construction method," *Signal Processing: Image Communication*, vol. 71, pp. 66–75, 2019.

[18] N. K. Pareek, V. Patidar, and K. Sud, "Image encryption using chaotic logistic map," *Image and Vision Computing*, vol. 24, pp. 926–934, 2006.

[19] H. A. Dmour and A. A. Ani, "Quality optimized medical image information hiding algorithm that employs edge detection and data coding," *Computer Methods and Programs in Biomedicine*, vol. 127, pp. 24–43, 2016.

[20] C. C. Thien and J. C. Lin, "Secret image sharing," *Computers & Graphics*, vol. 26, no. 5, pp. 765–770, 2002.

[21] Y. X. Liu, C. N. Yang, C. M. Wu, Q. D. Sun, and W. Bi, "Threshold changeable secret image sharing scheme based on interpolation polynomial," *Multimedia Tools and Applications*, vol. 78, pp. 1–15, 2019.

[22] Y. Liu, C. N. Yang, Y. Wang, L. Zhu, and W. Ji, "Cheating identifiable secret sharing scheme using symmetric bivariate polynomial," *Information Science*, vol. 453, pp. 21–29, 2018.

[23] Y. X. Liu and C. N. Yang, "Scalable secret image sharing scheme with essential shadows," *Signal Processing: Image Communication*, vol. 58, 2017.

[24] K. S. Aparnaa, M. Sathyasundaram, and P. Santhi, "Securing internet banking with a two - shares visual cryptography secret image," *International Journal of Engineering Research and Technology*, vol. 5, 2016.

[25] F. S. Ibraheema, "A new electronic voting protocol using secret sharing based on set of path domination," *Journal of Qadisiyah Computer Science Mathematics*, vol. 10, pp. 6–14, 2018.

[26] J. Zhang, M. Huang, B. Gong, Z. Jia, and L. Wang, "A blind signature scheme applying on electronic payment scene based on quantum secret sharing," in *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering*, J. Li, Z. Liu, and H. Peng, Eds., Springer, Cham, Switzerland, 2019.

[27] B. Chor, S. Goldwasser, S. Micali, and B. Awerbuch, "Verifiable secret sharing and achieving simultaneity in the presence of faults," in *Proceedings of the 26th Annual Symposium on Foundations of Computer Science (sfcs 1985)*, pp. 383–395, IEEE, Portland, OR, USA, October 1985.

[28] M. Stadler, "Publicly verifiable secret sharing," in *Advances in Cryptology — EUROCRYPT*, U. Maurer, Ed., vol. 96, pp. 190–199, Springer, Berlin, Heidelberg, 1996.

[29] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *Proceedings of the 28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*, pp. 427–438, IEEE, Los Angeles, CA, USA, October 1987.

[30] Y. X. Liu, Q. D. Sun, and C. N. Yang, "(k,n) secret image sharing scheme capable of cheating detection," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, pp. 1–6, Article ID 72, 2018.

[31] X. Yan, Q. Gong, L. Li, G. Yang, Y. Lu, and J. Liu, "Secret image sharing with separate shadow authentication ability," *Signal Processing: Image Communication*, vol. 82, Article ID 115721, 2020.

[32] C. Charnes, K. Martin, J. Pieprzyk, and R. N. Safavi, *Secret Sharing in Hierarchical Groups,* ICICS, Spring, Berlin, Heidelberg, 1997.

[33] C. Hu, R. Li, Bo Mei, W. Li, A. Alrawais, and R. Bei, "Privacy-preserving combinatorial auction without an auctioneer," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, pp. 1–8, Article ID 38, 2018.

[34] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, pp. 612-613, 1979.

[35] T. Chen and K. Tsao, "Threshold visual secret sharing by random grids," *Journal of Systems and Software*, vol. 84, pp. 1197–1208, 2011.

[36] X. Yan, X. Liu, and C. N. Yang, "An enhanced threshold visual secret sharing based on random grids," *Journal of Real-Time Image Processing*, vol. 14, pp. 61–73, 2015.