An Integratable Verifiable Secret Sharing Mechanism

Yanjun Liu[1,2] and Chin-Chen Chang[2,3]
(Corresponding author: Chin-Chen Chang)

Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education,
School of Computer Science and Technology, Anhui University[1]
No. 111 Jiulong Rd., Hefei 230601, China
Department of Computer Science and Information Engineering, Asia University[2]
No. 500, Lioufeng Rd., Wufeng, Taichung 413, Taiwan
Department of Information Engineering and Computer Science, Feng Chia University[3]
No. 100 Wenhwa Rd., Seatwen, Taichung 407, Taiwan
(Email: alan3c@gmail.com)

(Received Nov. 26, 2013; revised and accepted Mar. 5 & May 13, 2014)

Abstract

Threshold secret sharing (SS), also denoted as $(t,\ n)$ SS, has been used extensively in the area of information security, such as for group authentication, cloud storage schemes, secure parallel communication and wireless multipath routing protocols. However, a $(t,\ n)$ SS cannot detect any deceptions among the dealer and shareholders. Verifiable secret sharing (VSS) overcomes the weakness of $(t,\ n)$ SS in such a way that it is able to detect cheaters by verifying the validity of shares or the correctness of the recovered secret under the condition that both shares and the secret are not compromised. Recently, two non-interactive VSSs based on Asmuth-Bloom's SS were proposed by Harn et al. and Liu et al., respectively. Both VSSs require shareholders to examine the range of values of some integers related to the secret before recovering the secret, which is a time-consuming operation. In this paper, we propose a novel integratable VSS mechanism that integrates the concepts of the generalized Chinese remainder theorem (GCRT), Shamir's SS and Asmuth-Bloom's SS. Our proposed VSS can verify that the secret reconstructed by any t or more shareholders is the same as the one that the dealer has generated. Analysis shows that our proposed VSS can provide perfect secrecy and better efficiency.

Keywords: Generalized Chinese remainder theorem (GCRT), hash function, secret sharing (SS), verifiable secret sharing (VSS)

1 Introduction

Threshold secret sharing (SS) [1, 3, 6, 9, 10, 11, 12, 20, 21, 22, 23, 25, 26] is a widely-used cryptographic mechanism for managing a secret or a key among a set of participants. A threshold SS is also denoted as a $(t,\ n)$ SS in which a dealer does not release the secret itself, but divides the secret into n shares that are distributed among n shareholders. By using a specific algorithm, any subset of t shares can recover the original secret. There are two security goals that a $(t,\ n)$ SS should achieve: 1) the secret can be recovered by any t or more than t shares; and 2) the secret cannot be determined by fewer than t shares.

Shamir's $(t,\ n)$ SS [25] is the first $(t,\ n)$ SS and was proposed in 1979. It is based on the Lagrange interpolating polynomial and can ensure perfect secrecy. Perfect secrecy means that even if no computational assumption is made, both security goals can still be achieved. Later in 1983, Mignotte [22] introduced another $(t,\ n)$ SS that is based on the Chinese remainder theorem (CRT) [5, 9, 20]. Mignotte's $(t,\ n)$ SS generates a particular integer sequence and selects the secret in the t-threshold range [11, 12, 21]. According to the sequence, any t or more than t shares can recover the secret by using the CRT. However, Mignotte's $(t,\ n)$ SS is not perfectly secure since it cannot accomplish the second security goal. In the same year, Asmuth and Bloom [1] proposed an enhanced version of Mignotte's $(t,\ n)$ SS which can guarantee perfect secrecy. Nowadays, Shamir's $(t,\ n)$ SS, Mignotte's $(t,\ n)$ SS and Asmuth-Bloom's $(t,\ n)$ SS have become fundamental tools applied in many areas of information security, such as for key distribution protocols, group authentication, cloud storage schemes, secure parallel communication and multipath routing protocols in wireless networks [6, 9, 10, 20, 23, 26].

A $(t,\ n)$ SS assumes that the dealer and shareholders are all honest, but this is not always the case. Therefore, the weakness of a (t, n) SS is that it cannot discover whether the dealer has transmitted inconsistent shares to shareholders or whether shareholders have released invalid shares when recovering the secret. An incorrect secret may be reconstructed without detection in these two cases. In order to overcome this weakness, in 1985, Chor et al. [7] introduced the concept of verifiable secret sharing (VSS). Verifiability is the property of detecting cheaters by verifying the validity of shares or the correctness of the recovered secret under the condition that both the shares and the secret are not compromised. Interactive and non-interactive VSSs are two types of VSS. Interactive VSSs require shareholders to interact with the dealer to execute the verification, which consumes a large amount of communication time. To reduce the communication cost, non-interactive VSSs have been proposed to replace interactive VSSs.

VSS expands the range of applications of SS and has been researched deeply in a great number of recently published literature [2, 8, 11, 12, 13, 14, 21, 24]. Benaloh [2] defined the concept of t-consistency and proposed an interactive VSS to verify that shares generated by the dealer are t-consistent (i.e. any subset of t shares defines the same secret). Feldman [8] proposed the first non-interactive VSS using encrypted functions. The security of Feldman's VSS depends on the hardness of solving the discrete logarithm. Qiong et al. [24] and Iftene [13] presented non-interactive VSSs based on Asmuth-Bloom's SS and Mignotte's SS, respectively. Kaya et al. [14] pointed out the security weaknesses in these two VSSs and developed a VSS based on Asmuth-Bloom's SS. Harn and Lin [11] extended a (t, n) VSS to a (n, t, n) VSS in which each shareholder also acts as a dealer. Based on Benaloh's VSS [2], their VSS can verify that shares satisfy the requirement of strong t-consistency. In 2013, Harn et al. [12] proposed a non-interactive VSS based on Asmuth-Bloom's SS in which additional verification secrets are used during the verification. Later, Liu et al. [21] proposed a more efficient VSS by also using Asmuth-Bloom's SS as a building block. Both VSSs require shareholders to examine the range of values of some integers related to the secret before recovering the secret, which is a time-consuming operation.

In this paper, we propose a novel integratable VSS mechanism based on the generalized Chinese remainder theorem (GCRT) [4, 15, 16, 17, 18, 19]. Our proposed VSS can verify that the secret reconstructed by any t or more shareholders is the same as the one that the dealer has generated. The contributions of our proposed VSS are listed below:

1) Our proposed VSS integrates the concepts of Shamir's SS, Asmuth-Bloom's SS, and GCRT. To the best of our knowledge, no research on VSS has adopted this approach. Thus, we are the first to combine these three fundamental elements in a VSS.

2) A one-way hash function is used to verify the correctness of the secret, thereby removing the operation of examining the range of values of additional integers.

3) Our proposed VSS can provide perfect secrecy.

4) Our proposed VSS simplifies two related works [12, 21] on VSS and achieves better efficiency.

The rest of this paper is organized as follows. Section 2 addresses some background knowledge related to VSS. Our proposed VSS is described in Section 3. Section 4 gives security and performance analyses of our proposed VSS. Finally, conclusions appear in Section 5.

2 Preliminaries

This section introduces some background knowledge related to VSS. We first introduce two famous SSs: Shamir's [25] and Asmuth-Bloom's (t, n) SS [1]. Then, we address the principle and features of the GCRT [4, 15, 16, 17, 18, 19]. Finally, we review two recently developed VSSs [12, 21].

2.1 Shamir's (t, n) SS

Shamir's (t, n) SS [25] is one of the most famous SSs, which is based on the Lagrange interpolating polynomial. Shamir's (t, n) SS has been adopted widely in the design of VSSs since it was proposed in 1979. Assume that there is one dealer D and n users $U = \{u_1, u_2, ..., u_n\}$. Dealer D first generates a secret s and divides it into n shares, and then issues these shares to n users secretly, in such a way that each user obtains one share. To achieve the objective that any t users (also called shareholders) can collaborate with each other by using their shares to recover the secret s generated by dealer D, Shamir's (t, n) SS executes the following two phases as follows.

Share Generation:

Step 1. Dealer D randomly selects a polynomial $g(x)$ of degree t-1: $g(x) = s+a_1x+a_2x^2+...+a_{t-1}x^{t-1}\text{mod}p$, where $s = g(0)$ is the secret, and t coefficients $s, a_1, a_2, ..., a_{t-1}$ are in the finite field $GF(p)$.

Step 2. Dealer D generates n shares $s_i = g(x_i)$ for $i = 1, 2, ..., n$, where x_i can be considered as some information of shareholder u_i, such as u_i's ID number.

Step 3. Dealer D sends share s_i to shareholder u_i in a private channel.

Secret Reconstruction:

Any t shareholders can use their received shares to reconstruct the secret s generated by dealer D. Supposing that $s_{lj} \in \{s_1, s_2, ..., s_n\}$ for $j = 1, 2, ..., t$ denote shares of t shareholders, secret s can be reconstructed by computing $s = g(0) = \sum_{j=1}^{t} g(x_{lj}) \prod_{m=1, m\neq j}^{t} \frac{x_{lm}}{x_{lm}-x_{lj}}\text{mod}p$.

According to these two phases, parameter t is usually regarded as a threshold value that defines the fewest number of shares for recovering the secret. Shamir's (t, n) SS is quite simple and can ensure perfect secrecy. Due to this merit in security, Shamir's (t, n) SS has become a practical tool in realizing secret sharing and a common building block in VSSs.

2.2 Asmuth-Bloom's (t, n) SS

Different from Shamir's (t, n) SS that is based on the Lagrange interpolating polynomial, Asmuth and Bloom [1] proposed a novel SS based on the CRT. Asmuth-Bloom's (t, n) SS can also provide perfect secrecy, which has gathered increasing attention in VSS research. If a dealer D and n shareholders $U = \{u_1, u_2, ..., u_n\}$ participate in this SS, it can be described as follows.

Share Generation:

Step 1. Dealer D selects $n+1$ pairwise, co-prime integers, $p_0, p_1, p_2, ..., p_n$, that satisfy two requirements, i.e., $p_1 < p_2 < ... < p_n$ and $p_0 \cdot p_{n-t+2} \cdot p_{n-t+3} \cdot ... \cdot p_n < p_1 \cdot p_2 \cdot ... \cdot p_t$.

Step 2. Dealer D generates an integer s as the secret such that $0 \leq s < p_0$.

Step 3. Dealer D generates another integer $A = s + bp_0$, where b is an arbitrary integer such that $0 \leq A < \prod_{i=1}^{t} p_i$.

Step 4. Dealer D creates n shares $s_i = A(\text{mod} p_i)$ for $i = 1, 2, ..., n$, and then sends s_i to shareholder u_i in a private channel.

Secret Reconstruction:

Any t shareholders can use their received shares to reconstruct secret s generated by dealer D. Supposing that $s_{lj} \in \{s_1, s_2, ..., s_n\}$ for $j = 1, 2, ..., t$ denote shares of t shareholders, the secret s can be reconstructed according to the following steps:

Step 1. Integer A is recovered by using the CRT. First, the following system of equations is constructed:

$$s_{l1} = A(\text{mod} p_{l1}),$$

$$s_{l2} = A(\text{mod} p_{l2}),$$

$$\vdots$$

$$s_{lt} = A(\text{mod} p_{lt}).$$

Then, the unique integer A can be computed as $A = \sum_{j=1}^{t} M_j \cdot M_j^{'} \cdot s_{lj}(\text{mod} P)$, where $P = \prod_{j=1}^{t} p_{lj}$, $M_j = \frac{P}{p_{lj}}$, and $M_j \cdot M_j^{'} \equiv 1(\text{mod} p_{lj})$.

Step 2. Secret s is reconstructed by computing $s = A(\text{mod} p_0)$.

However, Harn et al. [12] pointed out that if integer A selected by dealer D is in the range of $[0, p_1 \cdot p_2 \cdot ... \cdot p_t)$, Asmuth-Bloom's (t, n) SS is actually not perfectly secure. This is because in this case, secret s could be recovered by fewer than t shares, which indicates that both security goals cannot be fulfilled at one time. Harn et al. modified Asmuth-Bloom's (t, n) SS by confining A in a smaller range, $(p_{n-t+2} \cdot p_{n-t+3} \cdot ... \cdot p_n, p_1 \cdot p_2 \cdot ... \cdot p_t)$. They called this range as the ***t-threshold range*** [12] and denoted it as $Z_{p_{n-t+2} \cdot p_{n-t+3} \cdot ... \cdot p_n, p_1 \cdot p_2 \cdot ... \cdot p_t}$. They proved that only if A is selected in the t-threshold range, Asmuth-Bloom's (t, n) SS can ensure perfect secrecy.

2.3 Generalized Chinese Remainder Theorem (GCRT)

The generalized Chinese remainder theorem (GCRT) is an extension of CRT that adds a parameter k into the theorem. GCRT uses the following basic elements:

1) n positive integers, x_1, x_2, ..., x_n;

2) n positive, pairwise, co-prime integers, p_1, p_2, ..., p_n;

3) an integer k satisfying $\text{Max}\{x_i\}_{1 \leq i \leq n} < k < \text{Min}\{p_i\}_{1 \leq i \leq n}$.

To build the system of equations below:

$$x_1 = \lfloor X/p_1 \rfloor (\text{mod} k),$$

$$x_2 = \lfloor X/p_2 \rfloor (\text{mod} k),$$

$$\vdots$$

$$x_n = \lfloor X/p_n \rfloor (\text{mod} k).$$

From the GCRT, the unique integer X can be computed as $X = \sum_{i=1}^{n} N_i \cdot N_i^{'} \cdot B_i(\text{mod} k \cdot P)$, where $P = \prod_{i=1}^{n} p_i$, $N_i = k \cdot \frac{P}{p_i}$, $N_i \cdot N_i^{'} \equiv k(\text{mod} k \cdot p_i)$, and $B_i = \lceil \frac{x_i \cdot p_i}{k} \rceil$.

The GCRT can accomplish the same functionality as the CRT. However, the flexibility of the GCRT is better than the CRT due to the use of integer k. In the GCRT, only a change of k can generate a new integer X. On the contrary, if integer X needs to be updated, all parameters, such as x_1, x_2, ..., x_n, and p_1, p_2, ..., p_n, need to be modified. Therefore, the GCRT is regarded as an enhanced version of the CRT and it has been extensively applied in the fields of cryptography. As stated in Subsection 2.2, the CRT is used to recover the secret in Asmuth-Bloom's (t, n) SS. In 2012, Guo and Chang [9] analyzed the correctness of Asmuth-Bloom's (t, n) SS based on the GCRT. Inspired by their approach, we will use the GCRT-based Asmuth-Bloom's (t, n) SS as one of the building blocks of our novel VSS.

2.4 Review of Related VSS Work

In this subsection, we review two VSSs, one by Harn et al. [12] and the other by Liu et al. [21]. Both VSSs can verify whether the shares received by shareholders are consistent under the condition that the secrecy of both shares and the secret are not compromised. Their common characteristics are listed below:

1) Both VSSs rely on the assumption that dealer D may transmit a fake share to a shareholder; however, all shareholders behave honestly;

2) They are based on Asmuth-Bloom's (t, n) SS that depends on the CRT;

3) In the verification, all shareholders work together to verify that their shares are t-threshold consistent [12] by examining the range of values of some integers related to secret s;

4) They can verify all shareholders' shares simultaneously and conclude whether there exist any invalid shares, but invalid shares cannot be identified.

Next, we investigate the detailed processes of these two VSSs, respectively. In Harn et al.'s VSS, dealer D generates secret s and n shares according to Asmuth-Bloom's (t, n) SS. Moreover, the dealer selects additional r secrets (also called ***verification secrets***) in the t-threshold range and creates their corresponding shares. Afterwards, the dealer distributes one share of the secret s along with one share of each verification secret to each shareholder u_i secretly. In order to verify the validity of shares, shareholders should first open (recover) $r/2$ verification secrets and inspect whether they are in the t-threshold range. If this holds, it indicates that the remaining, unopened $r/2$ verification secrets are also in the t-threshold range. Based on $(r/2+1)$ shares owned by each shareholder, the liner combinations of secret s and each unopened verification secret can be recovered by using the CRT. If the recovered values are in a certain range [12], the secret can be proven as in the t-threshold range, thereby verifying the t-threshold consistency of shares. If the verification is passed, any t or more than t shareholders can reconstruct secret s by using the CRT; otherwise, shareholders require that the dealer redistribute shares.

In comparison, Liu et al.'s VSS is an improvement over that proposed by Harn et al. It uses a similar, but simpler method to accomplish share verification. In Liu et al.'s VSS, each shareholder generates an adjustment value instead of receiving r verification secrets as in Harn et al.'s VSS. All shareholders combine their shares with adjustment values to recover an integer that has a relationship with secret s by using the CRT. Consequently, this strategy saves considerable time by eliminating the recovery of opened and unopened verification secrets. If the recovered integer is in the modified t-threshold range, the secret is proven to be in the t-threshold range, thereby verifying the t-threshold consistency of shares. In addition, the process of secret reconstruction in Liu et al.'s VSS is the same as that in Harn et al.'s VSS. According to the performance analysis in [21], Liu et al.'s VSS reduces both computational and communication costs.

3 Our Proposed VSS Mechanism

In this section, we first discuss the motivations for improving the two previously described VSSs, and then propose an integratable VSS that is based on the concepts of Shamir's SS, Asmuth-Bloom's SS, and the GCRT.

3.1 Motivations

In Subsection 2.4, we described the main properties of Harn et al.'s VSS [12] and Liu et al.'s VSS [21]. In these two VSSs, share verification and secret reconstruction are two separate phases. In the share verification, all shareholders must collaborate with each other to recover some integers that have a relationship with the secret. Then the range of value of the recovered integer is investigated to ensure the validity of shares. If the verification is passed, it implies that each shareholder received a correct share from the dealer. Thus, t distinct shares can reconstruct the real secret. However, if the verification is not passed, there is no need to reconstruct the secret and the VSS stops at this point.

From the process in these two VSSs, it can be inferred that an integer related to the secret and the secret itself must be reconstructed by the CRT in the share verification and the secret reconstruction, respectively, if the shareholder shares are valid. These are two time-consuming operations where the efficiency can be improved. In fact, the phases of share verification and secret reconstruction can be integrated into a single phase that verifies whether the secret reconstructed by any t or more shareholders is the same as the one that the dealer has generated. Therefore, the validity of shares can also be verified without recovering another integer before secret reconstruction. This strategy can increase the efficiency to some extent. Moreover, the two previously proposed VSSs can only detect the cheating behavior of the dealer based on the assumption that all shareholders act honestly. This can be improved to detect the honesty of either the dealer or any shareholder.

3.2 Proposed VSS

Inspired by the VSSs presented by Harn et al. and Liu et al., we propose a novel integratable VSS that improves on their work. The word "integratable" means that the proposed mechanism integrates three fundamental methods used in secret sharing: Shamir's SS, Asmuth-Bloom's SS and the GCRT. In the following, we first address the model of our design and then give the detailed VSS mechanism.

Like the two previously discussed VSSs, our proposed VSS involves two parties: a dealer D and n shareholders $U = \{u_1, u_2, ..., u_n\}$. Dealer D generates a secret

s and divides it into n shares that are shared among n shareholders. t or more shareholders are responsible for reconstructing secret s. However, dealer D may deceive shareholders and deliver an invalid share to a shareholder. On the other hand, shareholders may also act dishonestly by releasing invalid shares when performing the reconstruction of secret s. Consequently, our proposed VSS must be able to verify the correctness of the reconstructed secret to check whether there exists any deception among either the dealer or shareholders.

In our proposed VSS, dealer D selects the secret s and then computes a one-way hash function $k = h(s)$. The hash code k is used as a parameter in the GCRT and n shares of the secret s are generated by the approach used in the GCRT-based Asmuth-Bloom (t, n) SS. In addition, dealer D constructs a Shamir (t, n) SS scheme in which the dealer selects a polynomial $g(x)$ of degree t-1 such that $g(0) = k$. Then, dealer D distributes shares of s and shares of k to shareholders. After receiving all the messages sent by dealer D, t shareholders recover k and then use k to recover the secret s via the GCRT. In the end, we check whether $h(s)$ is equal to the recovered k. If it is true, shareholders can conclude that the recovered secret s is identical to the real secret generated by the dealer.

Our proposed VSS consists of two phases, a setup phase and a verification phase. Figure 1 illustrates the flowchart of the setup phase and the detailed steps are presented as follows.

Setup Phase:

Step 1. Dealer D selects n+1 pairwise, co-prime integers, p_0, p_1, p_2, ..., p_n, that satisfy two requirements: (1) $p_0 < p_1 < p_2 < ... < p_n$, and (2) $p_0 \cdot p_{n-t+2} \cdot p_{n-t+3} \cdot \ldots \cdot p_n < p_1 \cdot p_2 \cdot \ldots \cdot p_t$.

Step 2. Dealer D generates the secret s such that $0 \leq s < p_0$.

Step 3. Dealer D computes $k = h(s)$, where h is a collision-free, one-way hash function and $0 < k < \text{Min}\{p_i\}_{1\leq i\leq n}$.

Step 4. Dealer D generates an integer $A = s+bp_0$, where b is an arbitrary integer which should make sure that $A \in Z_{k\cdot p_{n-t+2}\cdot p_{n-t+3}\cdot\ldots\cdot p_n, k\cdot p_1\cdot p_2\cdot\ldots\cdot p_t}$.

Step 5. Dealer D creates n shares $s_i = \lfloor A/p_i \rfloor (\text{mod} k)$ for $i = 1, 2, ..., n$.

Step 6. Dealer D selects a polynomial $g(x)$ of degree t-1: $g(x) = k + a_1x + a_2x^2 + ... + a_{t-1}x^{t-1}\text{mod}q$, where $k = g(0)$ and t coefficients $k, a_1, a_2, ..., a_{t-1}$ are in the finite field $GF(q)$.

Step 7. Dealer D generates n shares of k as $s^{'}_i = g(x_i)$ for $i = 1, 2, ..., n$.

Step 8. Dealer D sends s_i and $s^{'}_i$ to shareholder u_i in a private channel.

Verification Phase:

Assume that $u_{l1}, u_{l2}, ..., u_{lt} \in U$ are t shareholders and shareholder u_{lj} $(j = 1, 2, ..., t)$ received s_{lj} and $s^{'}_{lj}$ in the setup phase.

Step 1. $u_{l1}, u_{l2}, ..., u_{lt}$ use $s^{'}_{l1}, s^{'}_{l2}, ..., s^{'}_{lt}$ to recover k following Shamir's (t, n) SS.

Step 2. u_{lj} uses s_{lj}, p_{lj}, and the recovered k to release $C_j = N_{lj} \cdot N^{'}_{lj} \cdot B_{lj}(\text{mod}k \cdot P^{'})$, where $P^{'} = \prod_{j=1}^{t} p_{lj}$, $N_{lj} = k \cdot \frac{P^{'}}{p_{lj}}$, $N_{lj} \cdot N^{'}_{lj} \equiv k(\text{mod}k \cdot p_{lj})$, and $B_{lj} = \lceil \frac{s_{lj} \cdot p_{lj}}{k} \rceil$.

Step 3. $u_{l1}, u_{l2}, ..., u_{lt}$ work together to compute $A = \sum_{j=1}^{t} C_j(\text{mod}k \cdot P^{'})$ following the GCRT. Then, the secret s is reconstructed by computing $s = A(\text{mod}p_0)$.

Step 4. Check whether $h(s)$ is equal to k. If these two values are identical, we can conclude that the reconstructed secret s is correct; otherwise, the reconstructed s is not a valid value.

Remark 1. *In the setup phase, dealer D needs to generate two secret messages: s and k. s is the real secret needed to recover and k is used to verify the correctness of s. More specifically, k has multiple functionalities that can be described as follows: (1) Dealer D makes k as the hash code of the one-way hash function $h(s)$ in the setup phase. (2) k is an important parameter in the GCRT to generate shares of secret s as $s_i = \lfloor A/p_i \rfloor (\text{mod}k)$. Later, shareholders will use their shares and k to recover s according to the GCRT-based Asmuth-Bloom (t, n) SS. (3) To increase the degree of security, dealer D does not transmit k directly to shareholders, but establishes Shamir's (t, n) SS scheme in which k is considered as the secret. Then, in the verification phase, shareholders can recover k easily according to the method of Shamir's (t, n) SS. (4) We can check whether $h(s)$ is equal to k to verify the correctness of the recovered secret s. In summary, the use of k provides an additional level of security for our proposed VSS.*

Remark 2. *The verification phase of our proposed VSS can verify that the recovered secret s is identical to the real secret generated by the dealer. This verification process is completed by the one-way hash function $h(s)$. Thus, it is unnecessary to recover some integers related to the secret and to examine the range of this integer like in the two VSSs mentioned before. This can simplify the verification process. Furthermore, if the verification fails, we can conclude that either the dealer or the shareholder is dishonest. However, it is impossible to identify two situations: (1) the dealer sends invalid shares to shareholders; and (2) shareholders release invalid shares to recover the secret. Lastly, similar to Harn et al. and Liu et al.'s VSSs, our proposed VSS can verify all shares at one time.*

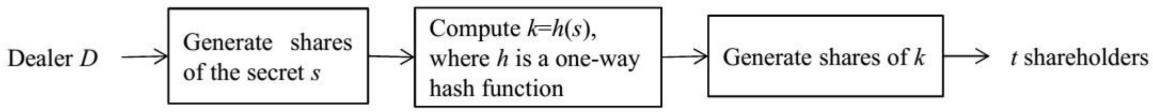
Dealer D
Generate shares of the secret s
Compute k=h(s), where h is a one-way hash function
Generate shares of k
t shareholders

Figure 1: Flowchart of the setup phase

4 Security and Performance Analyses

In this section, we first give security analysis of our proposed VSS, and then compare the performance among our proposed VSS and two other VSSs.

4.1 Security Analysis

Now we analyze that our proposed VSS can provide perfect secrecy. Our proposed VSS uses Shamir's (t, n) SS to share k among n shareholders, and later, t shareholders recover k and use it in the GCRT to recover secret s. Since Shamir's (t, n) SS is perfectly secure, fewer than t shares cannot recover k. Therefore, secret s cannot be recovered from fewer than t shares. Moreover, each shareholder u_{lj} releases a value $C_j = N_{lj} \cdot N^{'}_{lj} \cdot B_{lj}(\mathrm{mod} k \cdot P^{'})$, which combines s_{lj}, p_{lj}, and k. Thus, hidden in the value of C_j, share s_{lj} cannot be compromised during the verification phase.

In the following, we will analyze a situation where even if k is compromised, our proposed VSS is still perfectly secure, since fewer than t shareholders cannot obtain any useful information about secret s. Firstly, if fewer than t shareholders know the value of k, they cannot obtain secret s from $h(s) = k$ due to the intrinsic characteristic of the hash function. Next, we will prove that fewer than t shareholders cannot recover secret s from the GCRT.

Assume that $u_{l1}, u_{l2}, ..., u_{l(t-1)}$ are t-1 shareholders and each shareholder u_{lj} receives share s_{lj}. According to the GCRT, these t-1 shareholders cooperate to compute an integer $A^{'} = \sum_{j=1}^{t-1} N_{lj} \cdot N^{'}_{lj} \cdot B_{lj}(\mathrm{mod} k \cdot P^{''})$, where $P^{''} = \prod_{j=1}^{t-1} p_{lj}$, $N_{lj} = k \cdot \frac{P^{''}}{p_{lj}}$, $N_{lj} \cdot N^{'}_{lj} \equiv k(\mathrm{mod} k \cdot p_{lj})$, and $B_{lj} = \lceil \frac{s_{lj} \cdot p_{lj}}{k} \rceil$. However, $A^{'}$ is not equal to the real secret A since the range of $A^{'}$ is $Z_{k \cdot p_1 \cdot p_2 \cdot \ldots \cdot p_{t-1}}$, which is quite different from that of A as $Z_{k \cdot p_{n-t+2} \cdot p_{n-t+3} \cdot \ldots \cdot p_n, k \cdot p_1 \cdot p_2 \cdot \ldots \cdot p_t}$. Consequently, fewer than t shareholders cannot reconstruct the secret directly by using the GCRT. However, this does not mean fewer than t shareholders cannot obtain the real secret A from the recovered $A^{'}$. The use of GCRT implies that A and $A^{'}$ have a relation as $A = A^{'} + \varphi \cdot p_1 \cdot p_2 \cdot \ldots \cdot p_{t-1}$. Thus, A can be computed from $A^{'}$ if φ can be determined by fewer than t shareholders. Unfortunately, it is very hard to determine the correct φ. This is because $(p_1 \cdot p_2 \cdot \ldots \cdot p_t - p_{n-t+2} \cdot p_{n-t+3} \cdot \ldots \cdot p_n)/p_1 \cdot p_2 \cdot \ldots \cdot p_{t-1} > p_0$ values of φ can make $A^{'} + \varphi \cdot p_1 \cdot p_2 \cdot \ldots \cdot p_{t-1}$ in the range of $Z_{k \cdot p_{n-t+2} \cdot p_{n-t+3} \cdot \ldots \cdot p_n, k \cdot p_1 \cdot p_2 \cdot \ldots \cdot p_t}$, but only one value of $A^{'} + \varphi \cdot p_1 \cdot p_2 \cdot \ldots \cdot p_{t-1}$ is equal to A. Therefore, the probability of finding the exact value of φ is not greater than the probability of guessing the secret A. Based on this security analysis, our proposed VSS ensures perfect security based on the fact that fewer than t shareholders cannot obtain the real secret A from the recovered $A^{'}$.

4.2 Performance Analysis

In this section, we provide performance analysis of our proposed VSS and compare it with the other two related VSSs [12, 21] in terms of computational and communication costs.

Now we analyze the computational cost. In the setup phase, Harn et al.'s VSS creates and distributes shares of the real secret and r additional verification secrets to n shareholders. In contrast, the VSS by Liu et al. and our proposed VSS do not need to generate verification secrets. The difference between our VSS and Liu et al.'s VSS is that our VSS generates an additional polynomial of degree t-1 according to Shamir's SS. In the verification phase of the other two VSSs, n shareholders need to first cooperate to recover one or several integers related to the secret, and then t out of n shareholders recover the real secret, both by using the CRT. [21] analyzes that the time complexity of the VSSs proposed by Harn et al. and Liu et al. are $O(rn^2d^2)$ and $O(n^2d^2)$, respectively, where d is the number of bits of p_i and r is the number of verification secrets.

In comparison, in the verification phase of our proposed VSS, t out of n shareholders first recover the secret k in Shamir's (t, n) SS and then reconstruct the real secret with the recovered k by the GCRT. The time complexity of recovering k is $O(t\log^2 t)$ [25] and the time complexity of recovering the real secret is analyzed as follows. Secret A is computed as $A = \sum_{j=1}^{t} N_{lj} \cdot N^{'}_{lj} \cdot B_{lj}(\mathrm{mod} k \cdot P^{'})$ by the GCRT, where $P^{'} = \prod_{j=1}^{t} p_{lj}$, $N_{lj} = k \cdot \frac{P^{'}}{p_{lj}}$, $N_{lj} \cdot N^{'}_{lj} \equiv k(\mathrm{mod} k \cdot p_{lj})$, and $B_{lj} = \lceil \frac{s_{lj} \cdot p_{lj}}{k} \rceil$. Assume that d is the number of bits of both the operands, p_{lj} and k. Considering that $k \cdot P^{'}$ and $N_{lj} \cdot N^{'}_{lj}$ can be computed offline, this computational process totally contains (t-1) additions, t divisions, $2t$ multiplications, and one modular operation. Therefore, the number of bit operations required is $(t-1) \times d + t \times d^2 + 2t \times d^2 + ((t+1) \times d)^2$, and the time complexity is $O(t^2d^2)$.

Tables 1 and 2 summarize the communication and computational costs of our proposed VSS and the other two VSSs [12, 21]. From the comparisons among these VSSs, we can imply that our proposed VSS has better compu-

Table 1: Comparison of communication cost in setup phase

Scheme	Dealer sends messages	Each u_i sends messages	Each u_i receives messages
VSS in [12]	$n(r+1)$	-	$r+1$
VSS in [21]	n	-	1
Our VSS	$2n$	-	2

tational efficiency than the other two VSSs and our communication efficiency is also satisfactory.

Table 2: Comparison of computational cost

Scheme	Setup phase	Verification phase
VSS in [12]	$O(rn)$	$O(rn^2d^2)$
VSS in [21]	$O(n)$	$O(n^2d^2)$
Our VSS	$O(n)$	$O(t^2d^2)$

Note: n is the number of shares; d is the number of bits of operands; and r is the number of verification secrets.

5 Conclusions

In the paper, we propose a novel integratable VSS mechanism that integrates the concepts of the generalized Chinese remainder theorem (GCRT), Shamir's SS and Asmuth-Bloom's SS. Our proposed VSS improves Harn et al.'s VSS and Liu et al.'s VSS by using a one-way hash function to verify the correctness of the secret. While maintaining the advantages of the other two related VSSs, our proposed VSS is more efficient. In addition, we proved that our proposed VSS is perfectly secure.

Acknowledgments

This research was supported in part by the National Nature Science Foundation of China (grant number: 61202228) and the College Natural Science Key Project of Anhui Province of China (grant number: KJ2012A008). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] C. Asmuth and J. Bloom, "A modular approach to key safeguarding," *IEEE Transactions on Information Theory*, vol. IT-29, no. 2, pp. 208–210, 1983.

[2] J. C. Benaloh, "Secret sharing homomorphisms: Keeping shares of a secret," in *Advances in Cryptology (Crypto'86)*, pp. 251–260, Santa Barbara, USA, August 1986.

[3] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of American Federation of Information Processing Societies National Computer Conference*, pp. 313–317, New York, USA, June 1976.

[4] C. C. Chang and Y. P. Lai, "A fast modular square computing method based on the generalized Chinese remainder theorem for prime module," *Applied Mathematics and Computation*, vol. 161, no. 1, pp. 181–194, 2005.

[5] C. C. Chang, J. S. Yeh, and J. H. Yang, "Generalized Aryabhata remainder theorem," *International Journal of Innovative Computing, Information and Control*, vol. 6, no. 4, pp. 1865–1871, 2010.

[6] S. Chen and M. Wu, "Anonymous multipath routing protocol based on secret sharing in mobile ad hoc networks," *Journal of Systems Engineering and Electronics*, vol. 22, no. 3, pp. 519–527, 2011.

[7] 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 IEEE Symposium on the Foundations of Computer Science*, pp. 383–395, Portland, USA, Oct. 1985.

[8] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *Proceedings of 28th IEEE Symposium on Foundations of Computer Science*, pp. 427–437, Los Angeles, USA, Oct. 1987.

[9] C. Guo and C. C. Chang, "An authenticated group key distribution protocol based on the generalized Chinese remainder theorem," *International Journal of Communication Systems*, vol. 27, no. 1, pp. 126–134, 2014.

[10] L. Harn, "Group authentication," *IEEE Transactions on Computers*, vol. 62, no. 9, pp. 1893–1898, 2013.

[11] L. Harn and C. Lin, "Strong (n, t, n) verifiable secret sharing scheme," *Information Sciences*, vol. 180, no. 16, pp. 3059–3064, 2010.

[12] L. Harn, F. Miao, and C. C. Chang, "Verifiable secret sharing based on the Chinese remainder theorem," *Security and Communication Networks*, vol. 7, no. 6, pp. 950–957, 2014.

[13] S. Iftene, *Secret Sharing Schemes with Applications in Security Protocols*, Technical Report TR 07-01, Oct. 2006.

[14] K. Kaya and A. A. Selcuk, "A verifiable secret sharing scheme based on the Chinese remainder theorem," in *Advances in Cryptology (INDOCRYPT'08)*, pp. 414–425, Kharagpur, India, Dec. 2008.

[15] Y. P. Lai and C. C. Chang, "Parallel computational algorithms for generalized Chinese remainder theorem," *Computers and Electrical Engineering*, vol. 29, no. 8, pp. 801–811, 2003.

[16] C. H. Lin, C. C. Chang, and R. C. T. Lee, "A record-oriented cryptosystem for database sharing," *The Computer Journal*, vol. 35, no. 6, pp. 658–660, 1992.

[17] Y. Liu, C. C. Chang, and S. C. Chang, "An efficient oblivious transfer protocol using residue number system," *International Journal of Network Security*, vol. 15, no. 3, pp. 212–218, 2013.

[18] Y. Liu, C. C. Chang, and S. C. Chang, "A residual number system oriented group key distribution mechanism," *International Journal of Information Processing and Management*, vol. 4, no. 3, pp. 146–155, 2013.

[19] Y. Liu, C. C. Chang, and S. C. Chang, "An access control mechanism based on the generalized Aryabhata remainder theorem," *International Journal of Network Security*, vol. 16, no. 1, pp. 58–64, 2014.

[20] Y. Liu, L. Harn, and C. C. Chang, "An authenticated group key distribution mechanism using theory of numbers," *International Journal of Communication Systems*, vol. 27, no. 11, pp. 3502–3512, Nov. 2014.

[21] Y. Liu, L. Harn, and C. C. Chang, "A novel verifiable secret sharing mechanism using theory of numbers and a method for sharing secrets," *International Journal of Communication Systems*, vol. 28, no. 7, pp. 1282–1292, May 2015.

[22] M. Mignotte, "How to share a secret," in *Proceedings of the Workshop on Cryptography*, pp. 371–375, 1983.

[23] A. Parakh and S. Kak, "Space efficient secret sharing for implicit data security," *Information Sciences*, vol. 181, no. 2, pp. 335–341, 2011.

[24] L. Qiong, W. Zhifang, N. Xiamu, and S. Shenghe, "A non-interactive modular verifiable secret sharing scheme," in *Proceedings of International Conference on Communications, Circuits and Systems (ICCCAS 2005)*, pp. 84–87, Hong Kong, May 2005.

[25] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[26] H. Zhu, T. Liu, D. Zhu, and H. Li, "Robust and simple n-party entangled authentication cloud storage protocol based on secret sharing scheme," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 4, no. 2, pp. 110–117, 2013.

Yanjun Liu received her B.S. degree in 2005, in School of Computer Science and Technology from Anhui University, Hefei, China. She received her Ph.D. degree in 2010, in School of Computer Science and Technology from University of Science and Technology of China, Hefei, China. She is currently serving in Anhui University. Meanwhile, she is a postdoctor at Asia University, Taichung, Taiwan. Her current research interests include information security and computer cryptography.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His first degree is Bachelor of Science in Applied Mathematics and master degree is Master of Science in computer and decision sciences. Both were awarded in National Tsing Hua University. Dr. Chang served in National Chung Cheng University from 1989 to 2005. His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from Feb. 2005. Prior to joining Feng Chia University, Professor Chang was an associate professor in Chiao Tung University, professor in National Chung Hsing University, chair professor in National Chung Cheng University. He had also been Visiting Researcher and Visiting Scientist to Tokyo University and Kyoto University, Japan. During his service in Chung Cheng, Professor Chang served as Chairman of the Institute of Computer Science and Information Engineering, Dean of College of Engineering, Provost and then Acting President of Chung Cheng University and Director of Advisory Office in Ministry of Education, Taiwan. Professor Chang has won many research awards and honorary positions by and in prestigious organizations both nationally and internationally. He is currently a Fellow of IEEE and a Fellow of IEE, UK. And since his early years of career development, he consecutively won Outstanding Talent in Information Sciences of the R. O. C., AceR Dragon Award of the Ten Most Outstanding Talents, Outstanding Scholar Award of the R. O. C., Outstanding Engineering Professor Award of the R. O. C., Distinguished Research Awards of National Science Council of the R. O. C., Top Fifteen Scholars in Systems and Software Engineering of the Journal of Systems and Software, and so on. On numerous occasions, he was invited to serve as Visiting Professor, Chair Professor, Honorary Professor, Honorary Director, Honorary Chairman, Distinguished Alumnus, Distinguished Researcher, Research Fellow by universities and research institutes. His current research interests include database design, computer cryptography, image compression and data structures.