A (n,n)-Threshold Secret Sharing Scheme Based on Ggh Cryptosystem

Saeidi A1, Zahedi MM2, Nakhaei Amroodi A3
1Department of Mathematics, Islamic Azad University, Kerman Branch, Kerman, Iran
2Department of Mathematics, Kerman Graduate University of Technology (KGUT), Kerman, Iran
3Department of Mathematics and Cryptography, Malek Ashtar University of Technology, Isfahan, Iran

*Corresponding Author Email: alisaeidi2010@gmail.com

Abstract

In this paper, we consider the problem of secret sharing where the secret is encrypted using Goldreich-Goldwasser-Halevi (GGH) public-key cryptosystem and the ciphertext is publicly available. We present a (n,n)-threshold secret sharing scheme whose security can be reduced to the GGH cryptosystem. This is a strong property since the GGH is secure for large enough dimensions. Also, the scheme has certain technical advantage: we do not use the discrete logarithm problem for verification.

Keywords: GGH cryptosystem, Secret sharing, Verifiable, Encryption, Decryption.

Introduction

During the past three decades, security and cryptographic researches have been developed in many various fields (Moussavi Khalkhali et al., 2004; Naccache, 2012; Kandul et al., 2015). In cryptography formal notions of security have evolved in two main directions: 1- reduction based proofs (from the initial work of Goldwasser and Micali (1984), 2- simulation-based proofs (from the initial work of Goldreich et al (1987). In this paper, we explore a combination of public-key encryption (PKE) with an (n,n)-threshold secret sharing scheme. A (t,n)-threshold secret-sharing scheme is a fundamental cryptographic scheme, which allows a dealer owning a secret to distribute this secret among a group of n shareholders in such a way that any t shareholders can reconstruct the secret, but no subset of less than t shareholders can obtain information about the secret (Shamir, 1979; Mignotte, 1983). Secret sharing schemes were independently introduced by both Shamir (1979) and Blakley (1979), with the scope of safeguarding encryption keys.

Later other methods for key safeguarding were introduced (Harn et al., 2010). The original secret sharing scheme of Shamir (1979) scheme was based on polynomial interpolation while Blakley (1979) based his scheme on intersection of affine hyperplanes. Some applications of secret sharing schemes are access control systems, e-voting (Iftene, 2007), authentication protocols, information dispersal (Krawczyk, 1994) etc. In fact classical constructions for (t,n) secret-sharing schemes include the polynomial-based Shamir (1979) and the integer-based Chinese Remainder Theorem (CRT) scheme (Iftene, 2007). Chor et al (1985) proposed the notion of verifiable secret sharing (VSS). Using VSS, shareholders are able to verify that their shares are valid without revealing their shares. There are vast research papers on VSS (Naccache, 2012; Harn et al., 2010) in the literature. However, VSS is a complicated process, which requires additional information and processing time. In this paper we construct a (n,n)-threshold secret sharing based on complexity and difficulty of lattice problems. The (n,n)-threshold schemes are called unanimous consent schemes in the literature. Definition 1. Let B=\{b_1,\ldots,b_n\} be a set of n linearly independent vectors in \mathbb{R}^m.

\[ L(B) = \left\{ \sum x_i b_i \mid x_i \in \mathbb{R} \right\} = \left\{ Bx \mid x \in \mathbb{R}^n \right\} \] (1)

The lattice generated by B is the set of all integer linear combinations of the columns vectors in matrix B=[b_1,\ldots,b_n] (Ajtai, 1996). The results on the complexity of lattices (Micciancio & Goldwasser, 2002; Wei et al., 2014) have drawn considerable attention to lattice problems as potential candidates to design cryptographic primitives, and encryption schemes in particular qua even the probability of generating a lattice has been investigated (Fontein & Wocjan, 2014). Recently the lattice based cryptosystems have widely investigated (Wang et al., 2014; Xiao-yuan et al., 2013). The connection between worst-case and average-case hardness of certain lattice problems, as shown by Ajtai and Dwork
(1997), remains a strong reason to create cryptosystems based on these problems. Ideally this would result in a cryptosystem such that breaking the system is provably as hard as solving the problem. In 1994, Shor invented an algorithm for quantum computers that is able to efficiently factorize numbers and to solve the discrete logarithm problem (Shor, 1994). As the research into quantum computers continues, Shor’s algorithm threatens the security of systems such as RSA, ElGamal and other cryptosystems based on the discrete logarithm problem.

This has prompted a search for so-called “post-quantum” alternatives to cryptosystems based on these problems. Lattice-based cryptography is one of several candidates that are possibly resistant to attacks from quantum computers. Thus, lattice problems are attractive options as bases for public-key cryptography. The famous cryptosystems based on lattices are AD (Ajtai & Dwork, 1997), GGH (Goldreich et al., 1997) and NTRU (Hoffstein et al., 1998). Recently some other cryptosystems based on ideal lattices have introduced(liao-yuan et al., 2013). While the AD cryptosystem is a theoretical interest, the GGH cryptosystem was suggested as a practical alternative to number theory-based schemes currently in use e.g. the RSA cryptosystem (Rivest et al., 1987). The GGH cryptosystem is secure even against chosen-ciphertext attacks (CCA-secure) (Nguyen, 1999).

Two other related cryptosystems are McEliece’s cryptosystem (McEliece, 1978) and NTRU (Hoffstein et al., 1998). Neither of them is a lattice based cryptosystem in the strict meaning of the term, but they uses ideas from other areas of mathematics (polynomial ring and finite field arithmetic respectively). The Goldreich-Goldwasser-Halevi (GGH) cryptosystem relies on the difficulty of the closest vector problem (CVP) in a lattice. In CVP, one considers a vector $x \in \mathbb{R}^n$ (not necessarily in the lattice) and wants to find a point $u \in L$ minimizing the distance between x and u (Ajtai, 1996). The outline of this article is organized as follows: in Section 2, we present GGH cryptosystem. Section 3 is devoted to describe the verifiable secret sharing. Finally, we give future works in Section 4.

**Ggh Cryptosystem**

Inspired by the seminal work of Ajtai (1997), Goldreich et al (1997) (GGH) proposed at Crypto ’97 a lattice analogue of the coding-theory-based public-key cryptosystem of McEliece (1978). While encryption is superficially the same for the McEliece and GGH cryptosystems, there are significant differences between the security analysis of these schemes. An advantage of the lattice approach is that the error vector is required to have less structure: it is only required to be short, compared with McEliece where the error vector must have low Hamming weight. Both schemes have ciphertexts larger than the messages but an advantage of McEliece is that ciphertexts have a fixed size whereas for GGH the coefficients are integers whose size can vary significantly.

The security of GGH is related to the hardness of approximating the closest vector problem (CVP) in a lattice. The Goldreich et al. (1997) focused on encryption. Five encryption challenges of GGH were issued on the Internet (Goldreich et al., 1997). Two years later, Nguyen (1999) found a flaw in the original GGH encryption scheme, which allowed solving four out of the five GGH challenges, and obtaining partial information on the last one. However, GGH can still be secure with an appropriate choice of the parameters (Nguyen & Regev, 2006). There are two methods for description of GGH cryptosystem: primitive idea (Goldreich et al., 1997) and improved idea (Micciancio, 2001). We make a brief description of them. More details can be found by Goldreich et al. (1997), and Micciancio (2001).

**Primitive Idea**

The private key consists of a secret matrix R, whose columns form a basis for the lattice L. In particular, R is chosen in such a way that the quantity $\rho = \frac{1}{2} \min_i \|r_i\|_2$ (where $r_i$ is the i-th vector of Gram-Schmidt orthogonalization process over R) is relatively large, so that all errors of length less than $\rho$ can be efficiency corrected using R. This basis consists of reasonably short integral vectors. In the GGH cryptosystem, R has chosen as a perturbation of a multiple of the identity matrix, so that its vectors are almost orthogonal: more precisely, $R = kI_n + E$ where $k = \lfloor \sqrt{n} + 1 \rfloor + 1$ and each entry of the $n \times n$ matrix E is chosen uniformly at random in $\{-1, \ldots, +1\}$ where 1 is commonly selected four. The function $[\cdot, \cdot]:\mathbb{R}^n \rightarrow \mathbb{R}^n$ denotes the round function of Babai, which is the function that sends each real number to the closest integer (Babai, 1986). Babai (1986) published rounding method to approximate CVP thanks to the LLL algorithm. Also B is the multiplication of R by sufficiently many small unimodular matrices. Therefore B= UR for a random matrix U. The encryption scheme takes the public basis B, an integer vector v and an error vector r of length at most $\rho$ such that $\frac{1}{2} \min_i \|b_i\|_2 < \|e\| \leq \rho$, and returns the ciphertext vector $c = Bv + r$. In other words to encrypt a message, we do the following procedure:

- encode it as an integer vector $V \in \mathbb{Z}^n$,
- generate a random error vector $r$. 

107
compute the ciphertext $c= Bv+r$ where the error vector $r$ is taken randomly from fixed points $\{-\sigma, +\sigma\}$ (In original paper: $\sigma \approx 3$).

We do the following procedure for decryption:

- find the closest vector $z \in \mathbf{L}$ from $c$ such that $\forall w \in \mathbf{L}, \|z - c\| \leq \|w - c\|$.
- recover an error vector by $r = c - z$.
- decode message from the error vector $r$.

In other words to decrypt a ciphertext $c$, we compute $v = B^{-1}RD$ to retrieve the message $v$ where $D = [R^{-1}c]$. The ciphertext in GGH encryption is considerably larger than the message as mentioned above. A precise analysis of this depends on the sizes of entries in $R$.

**Improved Idea**

Micciancio (2001) proposed a variant of the GGH cryptosystem. The first idea is encoding the message in the error vector rather than in the lattice point and reducing it to the orthogonal parallelepiped. This method is the same as original method in GGH cryptosystem. The second method is choosing $B$ as the Hermite normal form (HNF) of $R$ such that the ciphertext is $C = v - Bx \equiv v \mod B$ where $x_j = \frac{m_i - \sum_{j \neq i} b_{ij} x_j}{b_{ii}}$ and the function $\lfloor \cdot \rfloor : \mathbb{R} \rightarrow \mathbb{Z}$ denotes integer part function. In this case, $v$ is recovered by finding the lattice point $x$ closest to the target ciphertext $C = (v \mod B) = x + v$ (e.g., by using Babai’s rounding procedure (Babai, 1986), which gives $x = B_1^{-1}B^{-1}(x + v)$), and the associated plaintext vector $v = C - x$. We refer to Micciancio (2001) for further details.

**Verifiable Secret Sharing**

We denote the shareholders of the scheme by $P_i$, the secret by $m \in \mathbb{R}$, and the dealer of the scheme by $D$. His role is to compute the shares of the secret and distribute them to all the shareholders in the scheme. Our scheme is a $(n,n)$ threshold scheme, so in order to recover the secret, $n$ parties have to participate with their shares. In order to share the secret $m \in \mathbb{R}$ between $n$ participants, the dealer proceeds in the following manner: (in the same manner as in the section 2) he

- selects the private matrix $R$ and the error vector $e$,
- distributes $r_i$ to $P_i$ for $1 \leq i \leq n$ where $r_i$ is the $i$th column of $R$,
- computes the public matrix $B$ from $R$ via one of the methods in section 2,
- encrypts the secret $m$ using the encryption methods as mentioned in section 2,
- publishes $c$ and $B$ to $P_i$ ($1 \leq i \leq n$).

To recover the secret, all the shares are needed. In fact only $n$ participants can recover the secret. The recovery with verification is made as follows:

- To verify the scheme, we should show that the matrix $R$ is retrieved uniquely. Since $R$ has $n$ columns therefore the shareholders do not need to the verification. In fact, if one share (which is a column of $R$) does not be valid then the matrix $R$ (which has $n$ columns) must be retrieved by $n-1$ vectors but this is impossible. In other words, if at least one share do not be valid then $R$ will not obtain and therefore $m$ will not retrieve.
- If all the shares are gathered, they recover the matrix $R$, because any share is a column of the matrix $R$.
- The secret vector $m$ can retrieve by the decryption of $c$ using the decryption methods as mentioned in section 2.

Notice that if $k$ number of $P_i$s want to guess the secret $m$, then they should do $l^{n-k}$ searches to recover $R$. Thus this secret sharing scheme will be secure against brute-force attack when $n$ is selected large enough.

**Proposition 1.** Suppose that the elements of an $n \times n$ matrix $R$ belong to finite set $X$, where $X$ has $l$ elements, then the possible number of entries matrix $R$ in which $k$ columns are given is $l^{n(n-k)}$.

**Proof.** Any column has $n$ entries and any entry can take $l$ possible values. Therefore any column can take $l^n$ values using the basic counting principle. Since $k$ columns of an $n \times n$ matrix $R$ are given thus $n-k$ columns of matrix $R$ are unknown. Consequently, $n-k$ columns of matrix $R$ can take $l^{n-k}$ values.

$\Box$
Notice that GGH cryptosystem is secure for large enough dimensions. Therefore $n$ should be large enough in a (n,n)-threshold secret sharing scheme. To construct (n′,n′)-threshold secret sharing scheme in which n′ is small, the dealer constructs the secure (n,n)-threshold secret sharing scheme based on GGH cryptosystem so that n is large enough and n/n′ is an integer number. Then he considers n′ participants and distributes n/n′ shares to any participant. Consequently, if n′ participants pool their shares then they pool \( n \cdot \frac{n}{n′} = n \) shares and can retrieve the secret.

**Concluding remarks and future works**

We constructed a simple, efficient unanimous consent secret sharing scheme based on the famous GGH-cryptosystem. We showed that the scheme does not allow recovering the secret if at least one participant is missing. In addition, breaking the ciphertext reduces to the CVP problem which is believed to be hard. The scheme offers the possibility for the participants to check if all the shares distributed by the dealer are valid. There is still a lot of work to be done in order to improve the capabilities of the scheme: it would be desirable to find a (t,n) variant of the scheme where t ≠ n and to discover a way to make it multi-secret (to allow several secrets are shared instead of one secret is shared on each round).

**Acknowledgment**

This research was financed from the budget of Islamic Azad University Kerman Branch-Iran in the form of research design of “Post Quantum cryptography based on lattices”. Therefore, the authors are highly grateful to the Department of Mathematics, Islamic Azad University, Kerman Branch, Kerman, Iran for providing an excellent research environment in which to conduct this research.

**Reference**


