A Key Exchange Based on the Short Integer Solution Problem and the Learning with Errors Problem

Jintai Ding, Kevin Schmitt, and Zheng Zhang(✉)

Department of Mathematical Science, University of Cincinnati, Cincinnati, USA
zhang2zh@mail.uc.edu

Abstract. Short integer solution (SIS) and learning with errors (LWE) are two hard lattice problems. These two problems are believed having huge potential in application of cryptography. In 2012, Ding et al. [5] introduced the first provably secure key exchange based on LWE problem. On the other hand, we believe that it is very difficult to do key exchange on SIS problem only. In 2014, Wang et al. [6] did an attempt, but it was not successful. Mao et al. [7] broke the protocol by an attack based on CBi-SIS problem in 2016. However, their attack is not efficient. In this paper, we present a extremely straightforward and simple attack to Wang's key exchange and then we will construct a key exchange based on SIS and LWE problems.

Keywords: Key exchange · SIS · LWE · Attack · Lattice

1 Introduction

1.1 Background

Key exchange protocol makes it possible for two parties to exchange keys over untrusted channels. The first revolutionary key exchange protocol was presented by Diffie and Hellman [2], which is called Diffie-Hellman key exchange protocol. The security of Diffie-Hellman key exchange is based on a hard number theory problem called discrete logarithm problem. However, in 1994, Peter Shor [3] theoretically proved that these hard number theory problems can hardly resist the attack from a quantum computer. Therefore, a post-quantum key exchange is urgently needed. Key exchange based on hard lattice problems is considered to be one of the candidates of post-quantum key exchanges.

1.2 Key Exchange Based on SIS Problem

A well-know hard lattice problem is the SIS problem introduced by Ajtai [1]. Some efforts have been made to construct a key exchange based on SIS problem. Although there are other attempts of key exchange on SIS problem, the basic structure is the following.

C. Carlet et al. (Eds.): C2SI 2019, LNCS 11445, pp. 105–117, 2019.
https://doi.org/10.1007/978-3-030-16458-4_8

(1) Assume that Alice and Bob agree to do a key exchange. The system generates a random matrix $\mathbf{M} \in \mathbb{Z}_q^{n \times m}$.
(2) Alice chooses a secret key $\mathbf{s}_A \in \mathbf{Z}_q^m$ with norm $||\mathbf{s}_A|| \leq \beta$. She computes $\mathbf{P}_A = \mathbf{M}\mathbf{s}_A$ and send $\mathbf{P}_A$ to Bob.
(3) Bob chooses a secret key $\mathbf{s}_B \in \mathbf{Z}_q^n$ with norm $||\mathbf{s}_B|| \leq \beta$. He computes $\mathbf{P}_B = \mathbf{s}_B^T\mathbf{M}$, and sends $\mathbf{P}_B$ to Alice.
(4) Receiving $\mathbf{P}_B$, Alice computes $\mathbf{K}_A = \mathbf{s}_A^T\mathbf{P}_B^T = \mathbf{s}_A^T\mathbf{M}^T\mathbf{s}_B$.
(5) Receiving $\mathbf{P}_A$, Bob computes $\mathbf{K}_B = \mathbf{P}_A^T\mathbf{s}_B = \mathbf{s}_A^T\mathbf{M}^T\mathbf{s}_B$.

Note that in order to apply the SIS problem to ensure the security of Alice's secret key, we need the condition that $n \ll m$. On the other hand, we also need the condition that $m \gg n$ to apply the SIS problem to guarantee the security of Bob's secret key. Therefore both parties have to get much more numbers of variables than equations, which makes it impossible to do key exchange on SIS problem.

1.3 Key Exchange Based on LWE Problem

Another building block of lattice-based problem is the LWE problem introduced by Regev [4]. The LWE problem is attractive due to its security and efficiency. A lot of attempts have been made to build a key exchange on LWE problem, but not until 2012, the first provably secure key exchange based on LWE problem was published by Ding [5]. The scheme is very efficient in computation, and can be extended to Ring-LWE. A new invention in his protocol is to extract a shared secret from the two values which are very close by rounding with signal functions.

1.4 Our Contributions

We first present an attack to Wang's protocol [6] based on an elementary linear algebra problem: solving linear equations. We observe that any solution to the system of linear equations can be used to recover the shared key. Therefore we claim that SIS problem is irrelevant to Wang's key exchange and there is no need for Mao et al. [7] to solve any SIS related problem at all.

Next we present a key exchange based on both SIS problem and LWE problem. In other words, Alice will use LWE problem to ensure the security on what she sends to Bob and Bob will use SIS problem to ensure the security on what he sends to Alice. It is obvious that our system is not symmetric. After the switch, we can extract a shared key from the two values which are very close by signal function proposed by Ding [5] in key exchange based on LWE problem.

2 Attack to Wang's Protocol

2.1 Preliminary

Let us first recall the definition of SIS problem and its derivatives introduced in Wang et al.'s paper [6].

Definition 1 *(SIS problem). Given a random matrix* $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$*, the goal of SIS problem is to find a nonzero vector* $\mathbf{z} \in \mathbb{Z}^m$ *that satisfies* $\mathbf{Az} = \mathbf{0}$ *with* $||\mathbf{z}|| \leq \beta$*.*

Note that a solution to the equation $\mathbf{Az} = \mathbf{0}$ is easy to obtain without the requirement on the length ($||\mathbf{z}|| \leq \beta$) by Gaussian elimination, however it is hard to find a solution of short length.

Next, Wang et al. extend this problem to Bi-ISIS* Problem.

Definition 2 *(Bi-ISIS* Problem). Given integers* n*,* m*,* q *(*$m > n \log q$*), a real* β *as in SIS, and a matrix* $\mathbf{A} \in \mathbb{Z}_q^{m\times m}$ *with rank* n*,* $\mathbf{e}_1$ *is linearly independent with column vectors of* $\mathbf{A}$*,* $\mathbf{e}_2$ *is linearly independent with row vectors of* $\mathbf{A}$*, given vectors* $\mathbf{b}_1 \in \{\mathbf{Az} + \mathbf{e}_1 : \mathbf{z} \in \mathbb{Z}^m, \langle \mathbf{e}_2, \mathbf{z}\rangle = 0 \mod q\}$*, and* $\mathbf{b}_2^t \in \{\mathbf{z}^t\mathbf{A} + \mathbf{e}_2^t : \mathbf{z} \in \mathbb{Z}^m, \langle \mathbf{e}_1, \mathbf{z}\rangle = 0 \mod q\}$*, the goal is to find a vector* $\mathbf{x} \in \mathbb{Z}^m$ *and a vector* $\mathbf{y} \in \mathbb{Z}^m$ *such that*

$$\begin{cases} \mathbf{Ax} + \mathbf{e}_1 = \mathbf{b}_1 \mod q \text{ and } ||x|| \leq \beta \\ \mathbf{y}^t\mathbf{A} + \mathbf{e}_2^t = \mathbf{b}_2^t \mod q \text{ and } ||y|| \leq \beta \end{cases} \tag{1}$$

Finally they define the CBi-ISIS problem.

Given the parameters n, m, q and $m > n \log q$ as in ISIS problem, a matrix $\mathbf{A} \in \mathbb{Z}_q^{m\times m}$ with rank equals to n. For any vectors $\mathbf{x} \in \mathbb{Z}$ with $||\mathbf{x}|| \leq \beta$, and $\mathbf{y} \in \mathbf{Z}$ with $||y|| \leq \beta$, there exists two vector sets $\{\mathbf{v}_1, ..., \mathbf{v}_n\}$ which is linear independent with rows vectors of $\mathbf{A}$, and $\{\mathbf{u}_1, ..., \mathbf{u}_n\}$ which is linear independent with column vectors of$\mathbf{A}$, such that $\langle \mathbf{v}_i, \mathbf{x}\rangle = 0 \mod q$ and $\langle \mathbf{u}_i, \mathbf{y}\rangle = 0 \mod q$. The CBi-ISIS problem is defined as follows:

Definition 3 *(CBi-ISIS problem). Given* $\mathbf{Ax} + \mathbf{e}_1$ *and* $\mathbf{y}^t\mathbf{A} + \mathbf{e}_2^t$*, the goal is to compute* $\mathbf{y}^t\mathbf{Ax} \mod q$*, where* $\mathbf{e}_1 = \sum_{i\in S} \mathbf{u}_i$*. and* $\mathbf{e}_2^t = \sum_{i\in S'} \mathbf{v}_i^t$*. S and S′ are random subset of* $\{1, \cdots n\}$*.*

If there is an algorithm that solves the Bi-ISIS* problem, we can use this algorithm to solve CBi-ISIS problem.

Remark 1. *Given any poly-bounded* m*,* $\beta = poly(n)$*, as well as any prime* $q \geq \beta\sqrt{\omega(n \log n)}$*, the* $SIS_{q,m,\beta}$ *and* $ISIS_{q,m,\beta}$ *problems in the average case are as hard as approximating the problems* $SIVP_\gamma$ *and* $GapSVP_\gamma$ *in the worst case to within certain* $\gamma = \beta \cdot \tilde{O}(\sqrt{n})$ *factors.*

2.2 Notation

We will use the same notation in Wang et al.'s paper [6]: Let $\mathbb{Z}$ denote the ring of integers; $\mathbb{Z}_q$ is the finite field module q; ${\mathbb{Z}_q}^{m\times m}$ is the set of all $m \times m$ matrices with entries in $\mathbb{Z}_q$. We define the norm on $\mathbb{Z}^m$ to be the l_2 norm. We can view $\mathbb{Z}_q \subset \mathbb{Z}$ and use the l_2 norm on it. Furthermore, if t is a positive integer with $t \leq q$, we can view $\mathbb{Z}_t \subset \mathbb{Z}_q$.

Moreover, the operator $*$ is defined by $\mathbf{A}*\mathbf{x} = \mathbf{A}*\mathbf{x} = \mathbf{Ax} + \sum_{i\in S} \mathbf{u}_i \mod q$, in which S is a random subset of $\{1, .., n\}$. and $\mathbf{y}^t * \mathbf{A} = \mathbf{y}^t\mathbf{A} + \sum_{i\in S'} \mathbf{v}_i^t \mod q$, in which S' is a random subset of $\{1, .., n\}$.

2.3 Description of the Protocol

We now briefly describe the protocol [6].

1. Alice and Bob agree to use a random matrix $\mathbf{A} \in {\mathbb{Z}_q}^{m \times m}$ with rank n and a real number β.
2. Alice picks a random $\mathbf{x} \in \mathbb{Z}^m$ such that $||\mathbf{x}|| \leq \beta$, then generates the set $\mathbf{V} = \{\mathbf{v}_1^t, \ldots, \mathbf{v}_n^t\}$, which is linear independent with row vectors of $\mathbf{A}$, and $\langle \mathbf{v}_i, \mathbf{x} \rangle = 0 \mod q$. Alice keeps $\mathbf{x}$ private and publishes $\mathbf{V}$. Now Bob picks a random vector $\mathbf{y} \in \mathbb{Z}^m$ such that $||\mathbf{y}|| \leq \beta$, then generates $\mathbf{U} = \{\mathbf{u}_1 \ldots \mathbf{u}_n\}$ which is linear independent with column vectors of $\mathbf{A}$, and $\langle \mathbf{u}_i, \mathbf{y} \rangle = 0 \mod q$. Bob keeps $\mathbf{y}$ private and makes $\mathbf{U}$ public.
3. Alice uses $\mathbf{U}$ to compute $\mathbf{a} = \mathbf{A} * \mathbf{x} = \mathbf{A}\mathbf{x} + \sum_{i \in S} \mathbf{u}_i \mod q$, in which S is a random subset of $\{1, .., n\}$, and sends $\mathbf{a}$ to Bob.
4. Bob uses $\mathbf{V}$ to compute $\mathbf{b}^t = \mathbf{y}^t * \mathbf{A} = \mathbf{y}^t\mathbf{A} + \sum_{i \in S'} \mathbf{v}_i^t \mod q$, in which S' is a random subset of $\{1, .., n\}$, and sends $\mathbf{b}^t$ to Alice.
5. Alice computes $K_1 = \mathbf{b}^t \cdot \mathbf{x} = \mathbf{y}^t\mathbf{A}\mathbf{x} \mod q$.
6. Bob computes $K_2 = \mathbf{y}^t \cdot \mathbf{a} = \mathbf{y}^t\mathbf{A}\mathbf{x} \mod q$.

Therefore the shared secret key is $K = K_1 = K_2 = \mathbf{y}^t\mathbf{A}\mathbf{x} \mod q$.

2.4 Mao's Attack [7]

Mao et al. assume that the protocol was based on the Bi-ISIS* problem, their goal is to solve the CBi-ISIS problem. They try to keep the original $\mathbf{x}$ and $\mathbf{y}$ during the attack so that they will match the shared key. However, according to their experiments results [7], the decomposition of the matrix $\mathbf{A}$ and solving the matrix $\mathbf{T}_1$ such that $\mathbf{T}_1\mathbf{A} = \mathbf{0}$ are very slow.

2.5 Our Attack

Our attack to this protocol is based on solving linear equations. An eavesdropper can obtain the information $\{\mathbf{a}, \mathbf{b}\}$. Since $\mathbf{A}$ and $\mathbf{U}$ are public, the eavesdropper has the linear equations

$$\begin{cases} \mathbf{A}\bar{\mathbf{x}} + \sum_{i \in \{1,..,n\}} \alpha_i \mathbf{u}_i = \mathbf{a} \mod q \\ \mathbf{v}_i^t \cdot \bar{\mathbf{x}} = 0 \mod q, \text{ for } i \in \{1, ..n\} \end{cases} \tag{2}$$

The linear independence of $\mathbf{U}$ with columns of $\mathbf{A}$ does not make any obstacle for the eavesdropper to solving the linear equations. Since $\mathbf{a}$ is of this form, the linear equations must contain at least one solution. Assume $\mathbf{A}$ has entires $[a_{ij}]$, in which

$1 \leq i, j \leq m$, $\mathbf{a} = (a_1, \cdots, a_m)^t$, $\mathbf{u}_i = (u_{i1}, \cdots, u_{im})^t$, and $\mathbf{v}_i = (v_{i1}, \cdots, v_{im})^t$. The equations have the following matrix form:

$$\begin{bmatrix} a_1 \\ \vdots \\ a_m \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1m} & u_{11} & \cdots & u_{n1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mm} & u_{1m} & \cdots & u_{nm} \\ v_{11} & \cdots & v_{1m} & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ v_{n1} & \cdots & v_{nm} & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \bar{x_1} \\ \vdots \\ \bar{x_m} \\ \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} \tag{3}$$

The eavesdropper can solve the equations and get solutions $\bar{\mathbf{x}} = (\bar{x}_1, \cdots, \bar{x}_m)^t \in \mathbb{Z}_q^m$, and $\alpha_i \in \mathbb{Z}_q$. Although $\bar{\mathbf{x}}$ is not necessary equal to the original $\mathbf{x}$ and of course not necessarily short, the eavesdropper can still use it to recover the secret key. Once the eavesdropper obtains $\bar{\mathbf{x}}$, he computes

$$\begin{aligned} \mathbf{b}^t \cdot \bar{\mathbf{x}} &= \left(\mathbf{y}^t \mathbf{A} + \sum_{i \in S'} \mathbf{v}_i^t \right) \cdot \bar{\mathbf{x}} \mod q \\ &= \mathbf{y}^t \mathbf{A} \bar{\mathbf{x}} \mod q \\ &= \mathbf{y}^t \left(\mathbf{a} - \sum_{i \in \{1,..,n\}} \alpha_i \mathbf{u}_i \right) \mod q \\ &= \mathbf{y}^t \left(\mathbf{A}\mathbf{x} + \sum_{i \in S} \mathbf{u}_i - \sum_{i \in \{1,..,n\}} \alpha_i \mathbf{u}_i \right) \mod q \\ &= \mathbf{y}^t \mathbf{A} \mathbf{x} \mod q. \end{aligned}$$

Therefore, the eavesdropper successfully recovers the secret key. Similarly, one can do it on $\mathbf{b}^t = \mathbf{y}^t * \mathbf{A}$.

One can see that the process of our attack is very straightforward, which contains only two steps: (1) solve the linear equations. (2) compute the dot product $\mathbf{b}^t \cdot \bar{\mathbf{x}}$.

2.6 Experimental Results

We did the experiments with the same parameters in Mao et al.'s paper [7].

(q, m, n)	time_1	time_2
10007, 3854, 128	3.430 s	27842.89 s
6421, 3240, 80	2.250 s	8201.06 s
4099, 1536, 64	0.29 s	1638.64 s

Remark: time_1 is the time spent in our attack, and time_2 is the time Mao et al. spent in their attack [7]. We used the software of Magma student version on an Intel core i7 with CPU 3.2 GHz, 8 GB storage memory. Mao et al.'s plateform is an Intel Dual-Core2, CPU 2.6 Ghz, Windows 7 operating system with 4 G storage memory, they use the MATLAB version 7.9.

2.7 Toy Example

We show a toy example of our attack with parameters: $(q = 7, m = 5, n = 2, \beta = 3)$. We did this example on the software called Magma in our computer lab.

Alice and Bob agree on a random matrix $\mathbf{A}$ equal to

$$\begin{bmatrix} 1 & 3 & 6 & 4 & 1 \\ 3 & 5 & 1 & 1 & 3 \\ 6 & 2 & 3 & 1 & 6 \\ 3 & 4 & 2 & 0 & 3 \\ 2 & 6 & 5 & 1 & 2 \end{bmatrix}$$

Alice picks a random vector $\mathbf{x} = (1, 0, 2, 0, 1)^t$, then she generates the set $\mathbf{V}$whose elements are:

$$\mathbf{v}_1 = (1, 3, 5, 6, 3)^t$$
$$\mathbf{v}_2 = (5, 6, 6, 6, 4)^t$$

Each $\mathbf{v}_i$ is orthogonal to $\mathbf{x}$, and neither of them is in the row space of $\mathbf{A}$. Alice keeps $\mathbf{x}$ as a secret and publishes the set $\mathbf{V}$.

Bob picks a random vector $\mathbf{y} = (1, 2, 3, 0, 0)^t$, then he generates the set $\mathbf{U}$ whose elements are:

$$\mathbf{u}_1 = (5, 2, 4, 2, 6)$$
$$\mathbf{u}_2 = (1, 1, 6, 6, 3)$$

Bob keeps $\mathbf{y}$ private and makes $\mathbf{U}$ public.
Alice now computes $\mathbf{a} = \mathbf{A} * \mathbf{x} = \mathbf{A}\mathbf{x} + \mathbf{u}_1 = (6, 1, 0, 2, 0)^t$.
Bob computes $\mathbf{b}^t = \mathbf{y}^t * \mathbf{A} = \mathbf{y}^t\mathbf{A} + \mathbf{v}_1^t + \mathbf{v}_2 = (3, 4, 2, 1, 1)$.
Alice computes $K_1 = \mathbf{b}^t \cdot \mathbf{x} = 1$. Bob computes $K_2 = \mathbf{y}^t \cdot \mathbf{a} = 1$. Hence the secret shared key is 1.

Now let Eve be the eavesdropper. He can get $\{\mathbf{a}, \mathbf{b}\}$. He now sets the equation of (3). In the matrix form:

$$\begin{bmatrix} 6 \\ 1 \\ 0 \\ 2 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 3 & 6 & 4 & 1 & 1 & 5 \\ 3 & 5 & 1 & 1 & 3 & 3 & 6 \\ 6 & 2 & 3 & 1 & 6 & 5 & 6 \\ 3 & 4 & 2 & 0 & 3 & 6 & 6 \\ 2 & 6 & 5 & 1 & 2 & 3 & 4 \\ 5 & 2 & 4 & 2 & 6 & 0 & 0 \\ 1 & 1 & 6 & 6 & 3 & 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{x_1} \\ \bar{x_2} \\ \bar{x_3} \\ \bar{x_4} \\ \bar{x_5} \\ \alpha_1 \\ \alpha_2 \end{bmatrix} \tag{4}$$

By solving the above linear equations, he can get the solution $(\bar{x_1}, \bar{x_2}, \bar{x_3}, \bar{x_4}, \bar{x_5}, \alpha_1, \alpha_2,) = (2, 0, 6, 4, 0, 1, 0)^t$. It follows that $\bar{\mathbf{x}} = (2, 0, 6, 4, 0)^t$. Next he computes that $\mathbf{b}^t \cdot \bar{\mathbf{x}} = (3, 4, 2, 1, 1) \cdot (2, 0, 6, 4, 0)^t = 1 \mod 7$, which is exactly the secret shared key.

We see that even $\bar{\mathbf{x}}$ is not equal to the private key $\mathbf{x}$ that Alice keeps and has norm larger than β, $\bar{\mathbf{x}}$ still works to break the protocol.

3 Key Exchange on SIS and LWE

3.1 Preliminary

Now let us recall the learning with error (LWE) problem, the short integer solution problem, and the shortest independent vectors problem. For a finite set X, let $U(X)$ denote the uniform distribution on X.

Definition 4. *A function family is a probability distribution over a set of functions with common domain and range. For a function family $\mathcal{F}$ with a finite range and probability distribution χ over the common domain of $\mathcal{F}$, we say that $(\mathcal{F}, \chi)$ is pseudorandom if the distribution obtained from sampling $f \leftarrow \mathcal{F}$ and $x \leftarrow \chi$ and outputting $(f, f(x))$ and the distribution that samples $f \leftarrow \mathcal{F}$ and $y \leftarrow U(Y)$ and outputs (f, y) are indistinguishable. See [10] for more details.*

Definition 5. *The Learning With Errors (LWE) function family is the set of all functions $g_{\mathbf{A}}$ indexed by $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ with domain $\mathbb{Z}_q^n \times \mathbb{Z}_q^m$ and range $\mathbb{Z}_q^m$ defined by $g_A(s, e) = As + e$. The LWE function family is endowed with the uniform distribution over $\mathbb{Z}_q^{n \times m}$ to choose $g_{\mathbf{A}}$. For probability distributions χ on $\mathbb{Z}_q^n$ and Ψ on $\mathbb{Z}_q^m$, we denote by LWE(m, n, q, χ, Ψ) the distribution obtained by sampling a function $g_{\mathbf{A}}$ from the LWE function family, $s \leftarrow \chi$, $e \leftarrow \Psi$, and outputting $g_{\mathbf{A}}(s, e) = \mathbf{A}s + e$.*

Definition 6. *The Short Integer Solution (SIS) function family is the set of all functions $f_{\mathbf{A}}$ indexed by $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ with domain $\mathbb{Z}_q^n$ and range $\mathbb{Z}_q^m$ endowed with the uniform distribution over $\mathbb{Z}_q^{m \times n}$. For a probability distribution χ on $\mathbb{Z}_q^n$, we denote by SIS(m, n, q, χ) the distribution obtained by sampling a function $f_{\mathbf{A}}$ from the SIS function family and sampling $x \leftarrow \chi$ and outputting $f_{\mathbf{A}}(x) = \mathbf{A}x$.*

Proposition 1 [8]**.** *For any $n, m \geq n + \omega(\log n), q$, and distribution χ over $\mathbb{Z}^m$, the LWE(m, n, q) function family is one-way (resp. pseudorandom, or uninvertible) with respect to input distribution $U(\mathbb{Z}_q^n) \times \chi$ if and only if the SIS$(m, m-n, q)$ function family is one-way (resp. pseudorandom, or uninvertible) with respect to the input distribution χ.*

Definition 7. *For $k \in \mathbb{N}$ and $\gamma > 0$, we denote by SIVP(k, γ) the shortest independent vectors problem in dimension k with approximation factor γ.*

Definition 8. *For $x \in \mathbb{R}^n$ and $s > 0$, let $\rho_s(x) = exp(-\pi \|x/s\|^2)$. ρ_s can be normalized into a gaussian probability measure on $\mathbb{R}^n$, and is denoted by $D_s(x) = \rho_s(x)/s^n$. For a lattice $\Lambda \subset \mathbb{R}^n$, let $\mathcal{D}_{\Lambda,s}(x) = \rho_s(x)/\rho_s(\Lambda)$, where $\rho_s(\Lambda) = \sum_{y \in \Lambda} \rho_s(y)$. Then $\mathcal{D}_{\Lambda,s}$ is a probability distribution on Λ and is called the discrete Gaussian distribution on Λ.*

For n, m positive integers and $s > 0$, we denote by $\mathcal{D}_{\mathbb{Z}_q^{m\times n},s}$ the distribution obtained by sampling from $\mathcal{D}_{\mathbb{Z}^m,s}$ n times and outputting $A \mod q \in \mathbb{Z}_q^{m\times n}$. We denote by $\mathcal{D}_{\mathbb{Z}_q^m,s}$ the distribution $\mathcal{D}_{\mathbb{Z}_q^{m\times 1}}$.

Lemma 2 [10]. *For any $s \geq \omega(\sqrt{\log n})$, then we have*

$$\mathbb{P}_{x\leftarrow\mathcal{D}_{\mathbb{Z}^n,s}}[||x|| \geq s\sqrt{n}] \leq 2^{-n}.$$

Lemma 3 LWE Assumption [4]. *It has been shown that as long as $\alpha q > 2\sqrt{n}$, then LWE$(m, n, q, U(\mathbb{Z}_q^n), \mathcal{D}_{\mathbb{Z}_q^m,\alpha q})$ is pseudorandom. It has been shown that the LWE distribution remains pseudorandom when the input distribution on $\mathbb{Z}_q^n$ is given by $\mathcal{D}_{\mathbb{Z}_q^n,\alpha q}$, this is called the HNF-LWE assumption.*

Lemma 4 U-LWE Assumption [9]. *Let $n = 8k$ for some $k \in \mathbb{N}$, $0 \leq a \leq n^{O(1)}$, $m = 2n+a$, $t = \lceil (Cm)^{9/7+(8a)/(7n)} \rceil$ for a large enough universal constant $C \geq 1$, and $16t^2 \leq q \leq n^{O(1)}$. Then LWE$(m, n, q, U(\mathbb{Z}_q^n), U(\mathbb{Z}_{t-1}^m))$ is pseudorandom under the assumption that SIVP$(k, \tilde{O}(\sqrt{k}q)$ is hard in the worst case. When k is assumed to be large enough so that SIVP$(k, \tilde{O}(\sqrt{k}q)$ is hard we call this the U-LWE assumption.*

Remark 2. *For $l \in \mathbb{N}$ and probability distributions χ over $\mathbb{Z}_q^{n\times l}$ and Ψ over $\mathbb{Z}_q^{m\times l}$, we can define the distribution LWE(m, n, l, q, χ, Ψ) to be given by sampling $A \leftarrow U(\mathbb{Z}_q^{m\times n})$, $\mathbf{s} \leftarrow \chi$, $\mathbf{e} \leftarrow \Psi$ and outputting $\mathbf{As} + \mathbf{e} \in \mathbb{Z}_q^{m\times l}$. We can also define SIS$(m, n, l, q, \chi)$ similarly. Notice that the LWE, HNF-LWE, and U-LWE assumptions hold with the added dimension l for these new distributions, i.e. they are pseudorandom under certain choices of distributions for $\mathbb{Z}_q^{n\times l}$ and $\mathbb{Z}_q^{m\times l}$.*

Remark 3. *Observe that we can also define the transpose function familys LWE^T and SIS^T which outputs $\mathbf{s}^T\mathbf{A}^T + \mathbf{e}^T$ and $\mathbf{x}^T\mathbf{A}^T$ respectively. We have that LWE^T is pseudorandom, and hence SIS^T is pseudorandom, under the LWE, HNF-LWE, or U-LWE assumptions respectively.*

Claim 1. *For n, m, k, and q positive integers, and matrices $\mathbf{A} \leftarrow U(\mathbb{Z}_q^{n\times m})$ and $\mathbf{B} \leftarrow U(\mathbb{Z}_q^{m\times k})$ we have that as nm k, and q are fixed, there exists an m computable in polynomial time such that AB is indistinguishable from uniform.*

One can see this claim by fixing q and taking a positive integer n. Then choosing $\mathbf{A}, \mathbf{B} \leftarrow U(\mathbb{Z}_q^{n\times n})$ and verifying that the distribution of $\mathbf{AB}$ approaches the uniform distribution on $\mathbb{Z}_q^{n\times n}$ for $n = \text{poly}(q)$. From this we can deduce the claim for non-square matrices.

3.2 Desciption of the Protocol

The diffie-Hellman key exchange is based on the fact that exponential map is commutative.

$$g^{ab} = (g^a)^b = (g^b)^a.$$

over some multiplicative group G with large order p. Ding's key exchange on LWE [5] uses the associativity of the bilinear form, namely

$$\mathbf{x}^T\mathbf{M}\mathbf{y} = \left(\mathbf{x}^T\mathbf{M}\right)\mathbf{y} = \mathbf{x}^T\left(\mathbf{M}\mathbf{y}\right).$$

for some vectors $\mathbf{x}$ and $\mathbf{y}$ in $\mathbb{Z}_q^n$ and a matrix $\mathbf{M} \in \mathbf{Z}_q^{n\times n}$. These two key exchange protocols are both symmetric. In other words, two parties do the same thing in the process of key exchange because the security that both parties rely on is from the same difficult problem. However, our key exchange protocol is different. It is not symmetric since one party will apply SIS and one party will apply LWE.

We give two similar key exchange protocols.

3.2.1 Normal Construction

Two parties Alice and Bob decide to do a key exchange over an open channel.

(1) The system first generates the public parameters q, n, m and α with $n \ll m$. Then generates the matrix $M \in \mathbb{Z}_q^{n\times m}$ uniformly at random. Let l and k be positive integers.
(2) Alice choose a secret matrix $s_A \leftarrow U(\mathbb{Z}_q^{l\times n})$ and an error matrix $e_A \leftarrow \mathcal{D}_{\mathbb{Z}_q^{l\times m},\alpha q}$, then computes $\mathbf{P}_A = s_A\mathbf{M} + 2e_A$. She sends $\mathbf{P}_A$ to Bob.
(3) Upon receiving $\mathbf{P}_A$, Bob chooses a secret matrix $s_B \leftarrow \mathcal{D}_{\mathbb{Z}_q^{m\times k},\alpha q}$ and computes $\mathbf{P}_B = \mathbf{M}s_B$ and sends $\mathbf{P}_B$ to Alice. Next he computes

$$\mathbf{K}_B = \mathbf{P}_A\mathbf{s}_B = \left(\mathbf{s}_A\mathbf{M} + 2\mathbf{e}_A\right)\mathbf{s}_B = \mathbf{s}_A\mathbf{M}\mathbf{s}_B + 2\mathbf{e}_A\mathbf{s}_B.$$

(4) Upon receiving $\mathbf{P}_B$, Alice computes

$$\mathbf{K}_A = \mathbf{s}_A\mathbf{P}_B = \mathbf{s}_A\mathbf{M}\mathbf{s}_B.$$

3.2.2 Uniform Construction

Two parties Alice and Bob decide to do a key exchange over an open channel.

(1) The system first generates the public parameters q, n, m, α, and t. Let $r \in \mathbb{N}$ and $m = 8r$, $0 \le a \le m^{O(1)}$, $n = 2m + a$, and $t = \lceil (Cn)^{9/7+(8a)/(7m)} \rceil$ for a large enough constant C ([9] Lemma 4). We have the additional constraint on q that $16t^2 \le q \le m^{O(1)}$. The system then generates the matrix $M \in \mathbb{Z}_q^{n\times m}$ uniformly at random. Let k be a positive integer.
(2) Alice chooses a secret $\mathbf{s}_A \leftarrow U(\mathbb{Z}_q^{n\times n})$, then she computes $\mathbf{P}_A = \mathbf{s}_A\mathbf{M} + 2\mathbf{e}_A$, where $\mathbf{e}_A \leftarrow \mathcal{D}_{\mathbb{Z}^{n\times m},\alpha q}$. She sends $\mathbf{P}_A$ to Bob.
(3) Receiving $\mathbf{P}_A$, Bob chooses a secret matrix $\mathbf{s}_B \leftarrow U(\mathbb{Z}_{t-1}^{m\times k})$. He computes $\mathbf{P}_B = \mathbf{M}s_B$. Bob sends $\mathbf{P}_B$ to Alice. Next, he computes

$$\mathbf{K}_B = \mathbf{P}_A\mathbf{s}_B = \left(\mathbf{s}_A\mathbf{M} + 2\mathbf{e}_A\right)\mathbf{s}_B = \mathbf{s}_A\mathbf{M}\mathbf{s}_B + 2\mathbf{e}_A\mathbf{s}_B.$$

(4) Receiving $\mathbf{P}_B$, Alice computes

$$\mathbf{K}_A = \mathbf{s}_A\mathbf{P}_B = \mathbf{s}_A\mathbf{M}\mathbf{s}_B.$$

3.3 Remove the Approximation

We imitate the way to remove the approximation that Ding [5] presents in his key exchange on LWE. We need the help of a robust extractor which allows two parties to extract identical information from two close elements with signal functions.

3.3.1 Robust Extractor

An algorithm E is a robust extractor on $\mathbb{Z}_q$ with error tolerance δ with respect to a hint function S if the following holds:

(1) The deterministic algorithm E takes as input an $x \in \mathbb{Z}_q$ and a signal $\sigma \in \{0,1\}$, outputs $k = E(x,\sigma) \in \{0,1\}$.
(2) The hint algorithm S takes as input a $y \in \mathbb{Z}_q$ and outputs a signal $\sigma \leftarrow S(y) \in \{0,1\}$.
(3) For any $x, y \in \mathbb{Z}_q$ such that $x - y$ is even and $|x - y| \leq \delta$, then it holds that $E(x,\sigma) = E(y,\sigma)$ where $\sigma \leftarrow S(y)$.

Signal function: For prime $q > 2$, we define $\sigma_0(x)$, $\sigma_1(x)$ from $\mathbb{Z}_q$ to $\{0,1\}$ as follows.

$$\sigma_0 = \begin{cases} 0 \text{ if } x \in \left[-\left\lfloor \frac{q}{4} \right\rfloor, \left\lfloor \frac{q}{4} \right\rfloor\right] \\ 1 \text{ otherwise} \end{cases} \qquad \sigma_1 = \begin{cases} 0 \text{ if } x \in \left[-\left\lfloor \frac{q}{4} \right\rfloor + 1, \left\lfloor \frac{q}{4} \right\rfloor + 1\right] \\ 1 \text{ otherwise} \end{cases}$$

In our robust extractor, we define the hint algorithm S as: for any $y \in \mathbb{Z}_q$, $S(y) = \sigma_b(y)$, where $b \xleftarrow{\$} \{0,1\}$. The robust extractor is defined as: $E(x,\sigma) = (x + \sigma \cdot \frac{q-1}{2} \mod q) \mod 2$.
By the construction of the robust extractor, Ding [5] proved that:

Lemma 5 [5]. *Let $q > 8$ be an odd integer, the function E defined above is a robust extractor with respect to S with error tolerance $\frac{q}{4} - 2$.*

Since our key exchange is of multiple bits, we need to extract the shared key from matrices. So we define a robust extractor over the space of matrices.

Definition 9. *Now for $i = 1, ..., l$ and $j = 1, ..., k$, given the robust extractor $E(x, \sigma_{i,j})$ on $\mathbf{Z}_q$ defined above, we define a robust extractor E' on $\mathbb{Z}_q^{l \times k}$:*

$$E'(\mathbf{A}, \sigma') = [E(a_{ij}, \sigma_{ij})] = \begin{bmatrix} a_{11} + \sigma_{11} \cdot \frac{q-1}{2} & \cdots & a_{1n} + \sigma_{1n} \cdot \frac{q-1}{2} \\ \vdots & \ddots & \vdots \\ a_{l1} + \sigma_{l1} \cdot \frac{q-1}{2} & \cdots & a_{ln} + \sigma_{l1} \cdot \frac{q-1}{2} \end{bmatrix} \mod q \mod 2.$$

where a_{ij} are the entires of $\mathbf{A}$ and σ' is a $l \times k$ matrix whose entries are σ_{ij}.

3.3.2 Extract the Shared Key

Alice has $\mathbf{s}_A$, Bob has $\mathbf{s}_B$.

Bob computes $\mathbf{P}_B$ as above and send it to Alice.

Receiving $\mathbf{P}_B$, Alice computes $\mathbf{K}_A$ as above and then she computes $\sigma' \leftarrow S(\mathbf{K}_A)$, then she obtains the shared key $\mathbf{SK}_A = E'(\mathbf{K}_A, \sigma')$. She also computes $\mathbf{P}_A$ as above and sends $(\mathbf{P}_A, \sigma')$ to Bob.

Bob receives $(\mathbf{P}_A, \sigma')$, and Bob computes $\mathbf{K}_B$ as above and computes $\mathbf{SK}_B = E'(\mathbf{K}_B, \sigma')$.

3.4 Correctness

We see that $\mathbf{K}_A - \mathbf{K}_B = -2\mathbf{e}_A\mathbf{s}_B$, and the entries of $\mathbf{K}_A - \mathbf{K}_B$ are even. We need to show that if each entry of the approximation $|2\mathbf{e}_A\mathbf{s}_B|$ is less than the error tolerance, then we obtain that $E'(\mathbf{K}_B, \sigma') = E'(\mathbf{K}_A, \sigma')$.

To complete the proof, we imitate a result from Ding's key exchange on LWE [5]:

Lemma 6. *If the uniform key exchange (Sect. 3.2.2) is run and* $2\alpha q(t-1)\sqrt{n} \leq \frac{q}{4} - 2$, *then* $\mathbf{SK}_A = \mathbf{SK}_B$ *with overwhelming probability. If the normal key exchange is run (Sect. 3.2.1) and* $2(\alpha q)^2\sqrt{lm} \leq \frac{q}{4} - 2$, *then* $\mathbf{SK}_A = \mathbf{SK}_B$ *with overwhelming probability.*

Proof. Let k_{ij} be an entry of $\mathbf{K}_A - \mathbf{K}_B$, so it can be expressed as $k_{ij} = -2\mathbf{v}_i^T\mathbf{u}_j$, where $\mathbf{v}_i$ is the i-th column vector of $\mathbf{e}_A^T$, $\mathbf{u}_j$ is the j-th column vector of $\mathbf{s}_B$. According to Lemma 2, if the Uniform key exchange is run, it is easy to see that

$$|k_{ij}| = |2\mathbf{v}_i^T\mathbf{u}_j| \leq 2\alpha q\sqrt{n}|u_j| \leq 2\alpha q(t-1)\sqrt{n}.$$

with overwhelming probability. According to Lemma 2 again, if the normal key exchange is run, it is easy to see that

$$|k_{ij}| = 2|\mathbf{v}_i^T\mathbf{u}_j| \leq 2\alpha^2 q^2\sqrt{lm}.$$

with overwhelming probability.

With such a choice of the parameters, we will have each entry of $|\mathbf{K}_A - \mathbf{K}_B|$ less than the error tolerance. By Lemma 5 and our definition of E', we have that

$$E'(\mathbf{K}_A, \sigma') = [E(x_{ij}, \sigma_{ij})] = [E(y_{ij}, \sigma_{ij})] = E'(\mathbf{K}_B, \sigma').$$

where x_{ij} is the entry of $\mathbf{K}_A$ and y_{ij} is the entry of $\mathbf{K}_B$.

Moreover we show that shared key is $E'(\mathbf{K}_A, \sigma') = E'(\mathbf{K}_B, \sigma') = \mathbf{s}_A\mathbf{M}\mathbf{s}_B + \frac{q-1}{2}\sigma' \mod q \mod 2$. It is clear that $\mathbf{s}_A\mathbf{M}\mathbf{s}_B + \frac{q-1}{2}\sigma' = \mathbf{K}_B + \frac{q-1}{2}\sigma'(\mathbf{K}_A) - 2\mathbf{e}_A\mathbf{s}_B$. Moreover we can observe that each entry of the matrix $|\mathbf{K}_B + \frac{q-1}{2}\sigma'(\mathbf{K}_A)|$ is less than $\frac{q}{4} + 1$. It follows that $\mathbf{s}_A\mathbf{M}\mathbf{s}_B + \frac{q-1}{2}\sigma' = \mathbf{K}_B + \frac{q-1}{2}\sigma'(\mathbf{K}_A) \mod q - 2\mathbf{e}_A\mathbf{s}_B$ because each entry of $|\mathbf{K}_B + \frac{q-1}{2}\sigma'(\mathbf{K}_A) \mod q - 2\mathbf{e}_A\mathbf{s}_B|$ is less than or equal to $\frac{q}{4} + 1 + \frac{q}{4} - 2 \leq \frac{q-1}{2}$. This implies that $\mathbf{SK}_B = E'(\mathbf{K}_B, \sigma') = \mathbf{s}_A\mathbf{M}\mathbf{s}_B + \frac{q-1}{2}\sigma'$. A similar proof shows that $\mathbf{SK}_A = E'(\mathbf{K}_A, \sigma') = \mathbf{s}_A\mathbf{M}\mathbf{s}_B + \frac{q-1}{2}\sigma'$. □

3.5 Security

Theorem 7. *If either protocol described above is run honestly by both parties Alice and Bob and the LWE (and resp. U-LWE) assumption hold, then* $\mathbf{SK}_A$ *and* $\mathbf{SK}_B$ *are indistinguishable from uniformly chosen elements of* $\mathbb{Z}_q^{l\times k}$ *given* $\mathbf{M}$, $\mathbf{P}_B$, *and* $\mathbf{P}_A$. *Thus the protocol is secure against passive adverseries.*

Proof. We only prove the theorem for protocol Sect. 3.2.1, the proof is similar for Sect. 3.2.2. Assuming the protocol is run honestly, the distribution of $\mathbf{P}_A$ is computationally indistinguishable from the uniform distribution on $\mathbb{Z}_q^{l\times m}$ due to the LWE assumption that

$$\mathrm{LWE}^T\left(m, n, l, q, U(\mathbb{Z}_q^{n\times l}), \mathcal{D}_{\mathbb{Z}_q^{m\times l}, \alpha q}\right) \text{ is pseudorandom.}$$

Now by the LWE assumption we have that

$$\mathrm{LWE}(m, n, k, q, U(\mathbb{Z}_q^{n\times k}), \mathcal{D}_{\mathbb{Z}_q^{m\times k}, \alpha q}) \text{ is pseudorandom.}$$

Thus by Proposition 1 we conclude that $\mathrm{SIS}(n, m, k, q, \mathcal{D}_{\mathbb{Z}_q^{m\times k}, \alpha q})$ is pseudorandom. Hence, as $\mathbf{P}_A$ is indistinguishable from uniform, it follows that $\mathbf{K}_B = \mathbf{P}_A s_B$ is computationally indistinguishable from the uniform distribution $\mathbb{Z}_q^{l\times k}$. Since $\mathbf{K}_B$ is indistinguishable from uniform, it follows that $\mathbf{SK}_B$ is indistinguishable from uniform by [4] (Lemma 3).

Now we focus on $\mathbf{SK}_A$. We have that $\mathbf{K}_A = s_A \mathbf{M} s_B$, where $\mathbf{M}$ and s_A are chosen uniformly at random. We invoke Claim 1, that for sufficiently large l and m, the distribution of $s_A M$ is indistinguishable from the uniform distribution over $\mathbb{Z}_q^{l\times m}$. Again, by the LWE assumption, we have that

$$\mathrm{LWE}(l, m, k, q, U(\mathbb{Z}_q^{l\times k}), \mathcal{D}_{\mathbb{Z}_q^{m\times k}, \alpha q}) \text{ is pseudorandom.}$$

Hence by Proposition 1 we deduce that $\mathrm{SIS}(l, m, k, q, \mathcal{D}_{\mathbb{Z}_q^{m\times k}, \alpha q})$ is pseudorandom. Thus, as $s_A M$ is indistinguishable from uniform and $s_B \leftarrow \mathcal{D}_{\mathbb{Z}_q^{m\times k}, \alpha q}$, we conclude that $K_A = (s_A M) s_B$ is indistinguishable from the uniform distribution on $\mathbb{Z}_q^{l\times k}$. Therefore $\mathbf{SK}_B$ is indistinguishable from uniform on $\mathbb{Z}_q^{l\times k}$ by [4] (Lemma 3) and the proof is complete. □

Acknowledgement. This study is partially supported by U.S Air force.

References

1. Ajtai, M.: Generating hard instances of lattice problems. Quaderni di Matematica **13**, 1–32 (2004). Preliminary version in STOC (1996)
2. Diffie, W., Hellman, M.: New directions in cryptography. Inf. Theory **22**(6), 644–654 (1976)
3. Shor, P.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Rev. **41**(2), 303–332 (1999)

4. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: STOC, pp. 84–93. ACM (2005)
5. Ding, J., Xiang, X., Lin, X.: A simple provably secure key exchange scheme based on the learning with errors problem. Cryptology ePrint Archive, Report 2012/688 (2012). https://eprint.iacr.org
6. Wang, S., Zhu, Y., Ma, D., Feng, R.: Lattice-based key exchange on small integer solution problem. Sci. China Inf. Sci. **57**(11), 1–12 (2014)
7. Mao, S., Zhang, P., Wang, H.: Cryptanalysis of a lattice based key exchange protocol. Sci. China Inf. Sci. **60**, 028101 (2016)
8. Micciancio, D., Peikert, C.: Hardness of SIS and LWE with small parameters. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. LNCS, vol. 8042, pp. 21–39. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40041-4_2
9. Cabarcas, D., Florian, G., Patrick, W.: Provably secure LWE encryption with smallish uniform noise and secret. Cryptology ePrint Archive, Report 2013/164 (2013). https://eprint.iacr.org
10. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. SIAM J. Comput. **37**(1), 267 (2007)