A Simple Provably Secure Key Exchange Scheme Based on the Learning with Errors Problem

Jintai Ding [1], Xiaodong Lin [2]

[1] Chongqing Universuty , University of Cincinnati
[2] Rutgers University

Abstract. We use a variant of learning with errors (LWE) problem, a simple and direct extension of the original LWE problem to the case of a small secret, which we call a small LWE problem (SLWE), to build a new simple and provably secure key exchange scheme. The basic idea behind the construction can be viewed as certain type of bilinear pairing with errors (PE). We build a more efficient implementation of our scheme using a similar LWE problem but solely based on matrices, and we extend our construction further using the ring LWE problem, where the provable security is based on the hardness of the ring LWE problem.

1 Introduction

The learning with errors (LWE) problem, introduced by Regev in 2005 [18], and its extension, the ring learning with errors (RLWE) problem [20] have attracted a lot of attentions in theory and applications due to their usage in cryptographic constructions with some good provable secure properties. The main claim is that they are as hard as certain worst-case lattice problems and hence the related cryptographic constructions. Recently they have been used to construct homomorphic encryption schemes [23].

A LWE problem can be described as follows.

First, we have a parameter n, a prime modulus q , and an "error" probability distribution κ on the finite field F_q with q elements.

Definition 1. *Let $\Pi_{S,\kappa}$ on F_q be the probability distribution obtained by selecting an element A in F_q^n randomly and uniformly, choosing $e \in F_q$ according to κ, and outputting $(A, < A, S > +e)$, where $+$ is the addition that is performed in F_q.*

An algorithm that solves the LWE problem with modulus q and error distribution κ, if, for any S in F_q^n , with an arbitrary number of independent samples from $\Pi_{S,\kappa}$, it outputs S (with high probability).

To achieve the provable security of the related cryptographic constructions based on the LWE problem, one chooses q to be specific polynomial functions of n, that is q is replaced by a polynomial functions of n, which we will denote as $q(n)$, κ to be certain discrete version of normal distribution centered around 0 with the standard deviation $\sigma = \alpha q \geq \sqrt{n}$, and elements of F_q are represented by integers in the range $[-(q-1)/2, (q-1)/2)]$.

In the original encryption scheme based on the LWE problem, one can only encrypt one bit a time and therefore the system is rather inefficient and it has a large key size. To further improve the efficiency of the cryptosystems based on the LWE problem, a new problem, which is a LWE problem based on a quotient ring of the polynomial ring $F_q[x]$ [20], was proposed. This is called the ring LWE (RLWE) problem. The RLWE problem is further used in homomorphic encryption schemes. In the cryptosystems based on the RLWE problem, their security is reduced to hard problems on a subclass of lattices, the class of ideal lattices, instead of general lattices.

What motivates the work in this paper is to try to build a simple key exchange protocol using the basic idea of from the LWE problem. There are already related works in [8, 11, 14, 15], but there is not yet any provably secure key exchange protocols based on the LWE problem as a direct generalization of the Diffie-Hellman key exchange protocol, which is elegant in terms of its simplicity. To achieve this goal, we use a new variant of the LWE problem suggested in [3] and proposal a new provably secure key exchange protocol.

The key idea behind our new construction can be viewed as a way to share a secret given by the value of pairing of two vectors X and Y in F_q^n via the bilinear form:

$$Q(X,Y) = X^t AY,$$

where A is a $n \times n$ square matrix. Surely in order to make the system provably secure, we need to introduce the small errors to achieve our goal. The main contribution of this paper is to use this simple idea to build a simple and provably secure key exchange scheme. Furthermore, we build a more efficient implementation of our scheme using a similar LWE problem but solely based on matrices, and we extend our construction further based on the ring LWE problem, where the provable security is based on hardness of the Ring LWE problem.

The paper is organized as follows. In Section 2, we will present the basic construction and the security proofs. In Section 3, we will present some more efficient implementations and a construction based on the Ring-LWE problem. In the last section, we will present the conclusion and the discussion.

2 An variant of the LWE problem and new key exchange protocol

We will first present a variant of the LWE problem, which was first presented in [3].

2.1 A variant of the LWE problem

Again, we assume that F_q is represented by integers in the range $[-(q-1)/2, (q-1)/2]$.

This variant of the LWE problem is based on the LWE problem.

We will replace a vector A with a matrix A of size $m \times n$, and S also with a matrix of size $n \times 1$, such that they are compatible to perform matrix multiplication $A \times S$. We also replace e with a compatible matrix of size $m \times 1$. We will work on the same finite field with q elements.

To simplify the exposition, we will only present, in detail, for the case where A is a square matrices of the size $n \times n$ and, S and e of the size $n \times 1$.

Definition 2. *Let $\Pi_{S,\kappa}$ over F_q be the probability distribution obtained by selecting an $n \times n$ matrix A, whose each entry are chosen in F_q uniformly and independently, choosing e as a $n \times 1$ vector over F_q with each entry chosen according to an error distribution κ independently, and outputting $(A, A \times S + e)$, where $+$ is the addition that is performed in F_q^n.*

An algorithm that solves a LWE with modulus q and error distribution κ, if, for any vector S in F_q^n, with any number of independent sample(s) from $\Pi_{S,\kappa}$, it outputs S (with high probability).

For the case that we choose a small S, namely each entry of S is chosen independently according to the error distribution κ, we call this problem a small LWE problem (SLWE). If we further impose the condition A to be symmetric, we call it a small symmetric LWE problem (SSLWE). If we choose the secret S randomly and independently from the set $-z, .., 0, 1.., z$ with z a fixed small positive integer, we call such a problem uniformly small LWE problem (USLWE).

Here we would like to point out, for this problem, we can also modify the SLWE problem to allow only **one** sample.

Let

$$A = \begin{pmatrix} A_1 \\ A_2 \\ . \\ . \\ . \\ A_n \end{pmatrix},$$

where A_i is the i-th row of the matrix A. Let $S = (s_1, ..., s_n)^t$, where s_j is the j-th element of S. Then we have that

$$A \times S + e = \begin{pmatrix} A_1 \\ A_2 \\ . \\ . \\ . \\ A_n \end{pmatrix} (s_1, ..., s_n)^t + \begin{pmatrix} e_1 \\ e_2 \\ . \\ \vdots \\ e_n \end{pmatrix}.$$

If we look at each entry of $(A, A \times S + e)$, say the j-th entry, then each entry at the j-th position actually is an output in the form

$$(A_i, < A_i, S^t > +e_i).$$

Therefore it is nothing but n independent outputs of a LWE problem. A SLWE problem above is nothing by putting n different independent samples of a LWE problem together but with further restrictions on S (small). Therefore, if we can solve any LWE problem efficiently, we should be able to solve the corresponding SLWE problem efficiently.

However, a SLWE has its advantages due to the following observation.

Proposition 3. *Assume that we have a LWE problem with n independent samples, then we can find many "fake" solutions efficiently. Here by a "fake" solution, we mean that we find a vector S' such that $(< A_i, S > +e_i)- < A_i, S' >$ follows the error distribution.*

The algorithm is rather simple. For the samples $(A_i, < A_i, S^t > +e_i)$, we choose randomly n error samples e'_i following the designated error distribution. Let $b_i =< A_i, S^t > +e_i$. We will then try to solve the set of n linear equation

$$< A_i, X >= b_i - e'_i,$$

for $i = 1, ..., n$. Due to the random property of A, e_i, e'_i etc, we can assume that this is a set of random linear equations with n equations and n variables, we know that we have a high probability to find a solution. Therefore, we can find a solution with a few tries, where we select different e'_i. We call this solution S'. Clear we have that $b_i- < A_i, S' >= e'_i$, which follows the error distribution. Therefore S' is a "fake" solution.

Surely we know that if the sample size is much large than n, like in the case of the encryption scheme based on the LWE problem, where the sample size is $2nlog(q)$, the fake solutions disappear. However if the sample size is slightly bigger, we can still find "fake" solutions easily following the same argument, but the probability to find a fake solution drops by a factor of q, when we add an additional sample.

However, for the SLWE problem, it is not at all easy to find "fake" solutions due to the restriction of small solutions, since the above algorithm should find solutions which should follow closely an uniform random distribution.

Due to the results in [3], we know

Theorem 4. *If the secret S's coordinates are sampled independently from the LWE error distribution, the corresponding LWE problem is as hard as LWE with a uniformly random secret S.*

This shows that the SLWE problem is as hard as the corresponding LWE problem.

The same is true for the case of the RLWE problem that if one can solve the Ring LWE problem with a small secret namely the element S being small, then one can solve it with an uniform secret [21]. We will then use the SLWE with a small secret to build our key exchange protocol.

As for the USLWE problem, the situation is slightly different, in this case, by [13], it is proven that if the secret has enough entropy, then the USLWE form of LWE problem is as hard as a LWE but with a (much) smaller dimension and (much) smaller error.

Remark 5. For our construction, we will use a version of LWE problem slightly different from the original LWE problem, namely the output will be in the form: $(A, A \times S + te)$, where t is a fixed small positive integer like 2, or 3. From the previous works, we know this is just as hard a problem as the original LWE problem, since the hardness reduction works here as well. The idea of introducing t appeared first in the RLWE constructions. [20]

2.2 A simple and provably secure key exchange protocol

Key exchange protocols are very important cryptographic protocols. The original Diffien-Hellman key exchange protocol [10] is built on the fact that the exponential maps are commutative, namely

$$(x^a)^b = (x^b)^a,$$

over F_q for a large q.

If we look carefully why the key exchange above works, one realize we may do the same thing using the associativity and commutativity of computing the value of bilinear form, namely,

$$X^t \times A \times Y = (X^t \times A) \times Y = X^t \times (A \times Y),$$

where A is a $n \times n$ matrix and X, Y are vectors of size n. Here this computation can be viewed a pairing of the two vector X and Y via the corresponding bilinear form.

Surely we need to introduce small errors, namely, the idea of LWE problem, to make the scheme secure. Our basic idea is that we can use the SLWE problem to build a key exchange protocol like the Diffie-Hellmann key exchange protocol.

The protocol can be set up as follows.

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

1. Alice and Bob will first publicly select F_q, n, a small prime integer $t(<< n)$, and a $n \times n$ matrix M randomly and uniformly over $F_q^{n^2}$, where $q \approx n^3$, and the error distribution over F_q^n to be a distribution such that each component are independent, and each component follow the same discrete distribution κ as in the case of LWE, namely a discrete normal distribution over F_q center around 0 with standard deviation approximately $\sqrt{n}$. All the information above is public.
2. Then each party chooses its own secret S_i a $n \times 1$ vector for $i = A, B$ according to the error distribution, namely entries are chosen independently according to the discrete normal distribution κ over F_q center around 0 with standard deviation approximately $\sqrt{n}$, and e_i an error vector chosen according to the same error distribution.
   For Alice, she computes
   $$M_A = MS_A + te_A,$$
   where t is the chosen small integer $(t << n)$.
   For Bob, he computes
   $$M_B = M^t S_B + te_b.$$

3. Both parties exchange M_i. This means both M_i are public, but certainly keep S_i and e_i secret.
4. Alice computes:
$$K_A = S_A^t \times M_B = S_A^t M^t S_B + t < e_B, S_A > .$$
Bob computes:
$$K_B = M_A^t \times S_B = S_A^t M^t S_B + t < e_A, S_B > .$$
5. Then Bob and Alice will derive a shared secret in the following way:
   (a) if K_B is out of the range $[-(q-1)/4, (q-1)/4]$, Bob will send a message of "No" to Alice. Then Alice will compute $K_A + (q-1)/2$ modular q (very likely into the range of $[-(q-1)/4, (q-1)/4]$) and then modular t and; Bob will compute $K_B + (q-1)/2$ modular q (into the range of $[-(q-1)/4, (q-1)/4]$) and then modular t, and they derive a shared secret;
   (b) if K_B is in the range of $[-(q-1)/4, (q-1)/4]$, Bob will send a message of "Yes" to Alice. Then Alice will compute K_A modular t and Bob will compute K_B modular t, and they derive a shared secret.

   Then we can repeat the process selecting independently different A, S_i, e_i or different A, e_i but the same S_i each time to derive as large a size of share secret as they wish.

The last step of the computation can be viewed as certain rounding technique, namely how to derive a shared information using numbers that are close to each other.

The reason that Alice and Bob can derive from K_A and K_B a shared secret to be the exchanged key using such a rounding technique is exactly due to the fact e_i and S_i are small, therefore we expect that $t < e_B, S_A >$ and $t < e_A, S_B >$ to be small.

Theorem 6. *Asymptotically, the probability that, for a single exchange round, Bob and Alice will derive different results is less than* e^{-n}.

The proof of the theorem follows from the following simple lemma in statistics [24].

Lemma 7. *Let x_i, y_i, $i = 1, ...n$, be $2n$ independent randomly variables over real numbers following the Normal distribution $\mathcal{N}(0,1)$, then, for $\epsilon > 1$,*

$$P(|\sum_1^n x_i y_i| \geq n\epsilon) \leq \exp\left(-\frac{n}{2}\left(\sqrt{1+4\epsilon^2} - 1 + \log(\sqrt{1+4\epsilon^2} - 1) - \log(2\epsilon^2)\right)\right)$$

This implies with large n, we have

Proposition 8.
$$P(|< \frac{1}{\sqrt{n}} e_B, \frac{1}{\sqrt{n}} S_A > | \geq 2n) \leq e^{-n};$$
$$P(|< \frac{1}{\sqrt{n}} e_A, \frac{1}{\sqrt{n}} S_B > | \geq 2n) \leq e^{-n}.$$

This means that
$$P(t| < e_B, S_A > | \geq 2n^2 t) \leq e^{-n};$$
$$P(t| < e_A, S_B > | \geq 2n^2 t) \leq e^{-n}.$$

Therefore, we can easily see that $t < e_B, S_A >$ and $t < e_A, S_B >$ are small with very high probability. The reason we chosen only K_B (or $K_B + (q-1)/2$) in the range $[-(q-1)/4, (q-1)/4]$ is that we do not want any modular q operation to occur when we add $t < e_B, S_A >$ to $S_A^t M S_B$ (or $S_A^t M S_B + (q-1)/2$) together, or when we add $t < e_A, S_B >$ to $S_A^t M S_B$ (or $S_A^t M S_B + (q-1)/2$)

together, and since $q \approx n^3$, we therefore have very high probability that we derive a shared secret. Therefore, **Theorem 5** is correct.

Theorem 5 also promises that if we repeat any $poly(n)$ times the key exchange protocol to get a shared secret of size $poly(n)$, we have an very high probability that all the secrets will match, and if we repeat the process e^n times, the probability that we have mismatched secret to be $1/e$.

It is straightforward to show that the secret Bob and Alice share follows a uniform and random distribution over $\mathcal{Z}/t$.

2.3 Security analysis

Now let us take a careful look at the security of such a scheme. Here, as usual, we will consider only the security against passive adversaries.

An attacker can monitor the whole communication process. Therefore to break the system, the attacker needs to solve the following mathematical problem.

Definition 9. *Assume that we are given*

- *a matrix M and prime integers t, q and the error distribution κ as above;*
- *$M_A = MS_A + te_A$ and $M_B = M^t S_B + te_B$, where e_i follows the error distribution described above and the entries of S_i also follows the same error distribution;*
- *and the fact that $K_B = M_A^t \times S_B = S_A^t M^t S_B + t < e_A, S_B >$ is in the range of $[-(q-1)/4, (q-1)/4]$ or not;*

the problem is to find an algorithm to derive K_A modular t if K_B is in the range of $[-(q-1)/4, (q-1)/4]$, and $K_A + (q-1)/2$ first modular q then modular t with high probability.

We call such a problem a pairing with error problem (PE).

To simplify the matter, we would like to look at the case, where t is set to be 2. In this case, it is clear that we would like to find an algorithm that can find K_B modular 2 if it is in the range of $[-(q-1)/4, (q-1)/4]$, otherwise $K_B + (q-1)/2$ modular q then modular 2, with a success probability to be at least $1/2 + \epsilon$, where ϵ is a fix nonzero positive number.

To solve such a problem, we need a oracle as follows.

Definition 10. *Assume that we are given*

- *a $n \times n$matrix M and prime integers $2 << n$, $q > n^3$,*
- *two vectors M_A and M_B,*
- *a one bit information of "yes" and "no".*

The oracle would return a number modular t.

If $M_A = MS_A + te_A$ and $M_B = M^t S_B + te_B$, where e_i follows the error distribution described above and S_i also follows the same error distribution; the signal "yes" is given if $K_B = M_A^t \times S_B = S_A^t M^t S_B + t < e_A, S_B >$ is in the range of $[-(q-1)/4, (q-1)/4]$ or otherwise a signal "no" is given, then the oracle would return $S_A^t M^t S_B$ modular 2 if K_B is in the range of $[-(q-1)/4, (q-1)/4]$, and $S_A^t M^t S_B + (q-1)/2$ first modular q then modular 2 with high probability.

We will show that the PE problem is as strong as a decision SLWE problem.

Definition 11. *Let $\Pi_{S,\kappa}$ be the probability distribution obtained by selecting an $n \times n$ matrix A, whose each entry are chosen in F_q uniformly and independently, choosing e as a $n \times 1$ vector over F_q with each entry selected independently according to the distribution κ, and outputting $(A, A \times S + te)$, where + is the addition that is performed in F_q^n, and S a* **small** *$n \times 1$ vector over F_q with each entry selected independently according to the distribution κ.*

An decision SLWE problem is to find an algorithm that, given two series of pairs of arrays $(A_i, A_i \times S + te)$, and (A_i, R_i),, where the $(A_i, A_i \times S + te)$ comes from $\Pi_{S,\kappa}$, and R_i is a vector whose each entry are chosen in F_q uniformly and independently, it can distinguish which one comes from $\Pi_{S,\kappa}$, which one is not, with high probability.

Theorem 12. *A PE problem is at least as hard as the corresponding decision SLWE problem.*

Proof

Given any pair of arrays from the two series we have which we call (A, R').

We randomly select a small vector S_B. Then we assume that (A, R') is from the $\Pi_{S,\kappa}$, namely in the form of $R' = A \times S + te'$, where S and e follows the distribution κ.

Then we use them to perform the part of key exchange protocol but only on the Bob side using (A, R'), where we will treat R' as M_A.

Namely we will compute

$$M_B = AS_B + te_B,$$

where $t = 2$, surely a small integer $(2 << n)$.

Then we compute:

$$K_B = R'^t \times S_B$$

and this should be

$$S^t AS_B + t < e', S_B >,$$

for some e', if the (A, R') is indeed from $\Pi_{S,\kappa}$

If K_B is in the range of $[-(q-1)/4, (q-1)/4]$, we will compute compute K_B modular $t = 2$; or if K_B is in not the range of $[-(q-1)/4, (q-1)/4]$, we will compute compute $K_B + (q-1)/2$ modular q (into the range of $[-(q-1)/4, (q-1)/4]$) and then modular $t = 2$.

Then we will call the oracle that can solve the PE problem, where the input will be A as M, R' as M_A, M_B. If indeed (A, R') is from $\Pi_{S,\kappa}$, we expect the two results to have a probability of $1/2 + \epsilon$ to match.

Now let us look very carefully at what the oracle does. The oracle is promised to do the following: if given a triple M, M_A, M_B, and an additional bit signal of "yes" or "no" for the range of K_B, the oracle returns a number modular 2, and if M_A and M_B are derived following the distribution described above, we have a $1/2+\epsilon$ success rate. This means that what the oracle does for inputs of M_A derived from (S_A, e_A) in the range $\|S_A\|_\infty < \frac{1}{2}n\sqrt{n}$ and $\|e_A\|_\infty < \frac{1}{2}n\sqrt{n}$, which has the probability large than $(1 - e^{n^2/2})^{2n}$ according to the error distribution, really matters, and otherwise outside this range whatever the oracle does does not really matter. In addition, the oracle tells us that if we have two different S_1, e_1 and S_2, e_2, as long as $MS_1 + te_1 = MS_2 + te_2 = M_A$, we would derive the same answer from the oracle, since the input of the oracle will be the same. But $S_1^t M^t S_B$ and $S_2^t M^t S_B$ have only 1/2 probability to be the same, since we can assume the matrix M involved is invertible (otherwise we can simply remove the pairs where the matrix is not invertible), and for a given M and M_A, M_B with its S_B, and the pair M and M_A can be derived from all the elements in the set of all pairs in the form of $(S_A, e_A) = (M^{-1}(M_A - te), e)$ for any e in F_q^n. This implies that the oracle to give the answer essentially randomly once outside the range above. But according to the uniform distribution, the probability to be in that range for an randomly uniformly chosen element occurs with less than $(\frac{1}{n\sqrt{n})})^{2n}$ probability. This implies that if M_A is uniformly chosen (not given by the (S_A, e_A) following the error distribution), the oracle should give independent random answers, therefore we expect that the probability that the result the oracle will produce should match what Bob derives at the probability 1/2.

This gives us a way to distinguish which series is which after only n/ϵ tries with very high probability.

This finishes the proof.

Following the same argument in [18, 19], we also know that

Theorem 13. *The decision SLWE problem is as hard as the corresponding search SLWE problem.*

Here we would like to point out that when we use a decision LWE protocol to solve the SLWE problem, we should test according to the probability distribution since the secret S follows the error distribution.

This implies that

Theorem 14. *The PE problem is as hard as the corresponding SLWE problem.*

Following Theorem 4 in Section 1, we therefore conclude that our key exchange protocol is provably secure, since to break it is as hard as the corresponding LWE problem.

We call our new scheme a SLWE key exchange protocol.

We can also build a provably secure key exchange scheme based on the USLWE problem in a similar way and its provable security relies on the related security assumption of the USLWE problem.

Remark 15. One may ask why we still try to build a key exchange protocol, since there are several public key encryption schemes that can already be used to perform the same task. There are several aspects from which we would like to point out the contributions of our work. First, from theoretical point of view, it is an interesting question to find out if we can do a nice key exchange without using the same mechanism as that of the encryption schemes, as in the case of the original Diffie-Hellmann key exchange Scheme. This is in general a nontrivial question. For example, in the case of multivariate public key cryptography, we still can not build a good key exchange scheme. Here our construction's key point is that it uses the structure of bilinear forms and it is a nice and elegant construction mathematically. Since our construction relies on different mathematical structures from the previous LWE constructions, it could provide new tools for constructing protocols for other applications. In the original LWE encryption scheme, the public key includes a matrix of size $(2n \log q) \times n$, while we use only a matrix of size $n \times n$. This means our scheme can be much more efficient in terms of both communication and computation cost. In some way, our construction is a significant additional step in showing how versatile the LWE assumption can be. The basic idea of our construction also motivated a more general matrix construction below, which is much more efficient than the original LWE construction.

3 More efficient implementations

We can also see that the scheme above is not very efficient in the sense that we can only derive one bit a time like the original LWE encryption scheme, and surely we would like to build something much more efficient for practical applications. If we follow the usual path, we should go directly to a construction of using the Ring-LWE problem, which indeed works. But here we would like to look something slightly different first, namely a construction using solely matrices.

3.1 A matrix implementation

As we discussed earlier that the original Diffien-Hellman key exchange protocol [10] is built on the fact that the exponential maps are commutative, namely $(x^a)^b = (x^b)^a$. We use certain type of paring with error to build a key exchange protocol in the section above. If we look carefully, we realize that we can use further the fact of associativity of matrices multiplications of three matrices A, B and C:

$$A \times B \times C = (A \times B) \times C = A \times (B \times C).$$

In this case, we will just use all matrices for our construction and not vectors anymore.

From this point of view, it is evident that we can use the matrix version of the SLWE problem to build a key exchange protocol.

Definition 16. *Let $\Pi_{S,\kappa}$ over F_q be the probability distribution obtained by selecting an $n \times n$ matrix A, whose each entry are chosen in F_q uniformly and independently, choosing e as a $n \times n$*

matrix over F_q with each entry chosen according to an error distribution κ independently, and outputting $(A, A \times S + e)$, where $+$ is the addition that is performed in $F_q^{n^2}$.

An algorithm that solves a LWE with modulus q and error distribution κ, if, for any $n \times n$ matrix S in F_q^n, with any number of independent sample(s) from $\Pi_{S,\kappa}$, it outputs S (with high probability).

We call this problem matrix LWE problem(MLWE). For the case where we choose a small S, namely each entry of S is chosen independently according to the error distribution κ, we call this problem a small MLWE problem (SMLWE). If we further impose the condition A to be symmetric, we call it a small symmetric MLWE problem (SSMLWE). If we choose the secret S randomly and independently from the set $-z, .., 0, 1.., z$ with z a fixed small positive integer, we call such a problem uniformly small MLWE problem (USMLWE).

It is clear the MLWE problem is nothing but put n LWE problem together but sharing the same matrices. Therefore it is as hard as the corresponding LWE problem.

Theorem 17. *An MLWE problem is as hard as the corresponding LWE problem with the same parameters.*

Proof. It is clear that if we can solve a LWE problem, we can solve the corresponding MLWE problem by breaking the MLWE problem into n separate LWE problems. Therefore we only need to show the other way is also true.

Assume that we have a series of output of a LWE problem, $(A_i, A_i \times S + e_i)$, for i=1,...,l where A_i is a $n \times n$ matrix, the secret S is a $n \times 1$ matrix vector, and e_i is a $n \times 1$ error vector. Since A_i are known, for $j = 1, ..., n-1$, we can independently and randomly select a $n \times 1$ vector S^j, and independently and randomly error vectors e_i^j, for $i = 1, ..., l$, and output the series $(A_i, A_i \times S_j + e_i^j)$. Then we can put them together to build a new MLWE problem with the output series $(A_i, A_i \times \bar{S} + E_i)$ where the matrix $\bar{S}$ and E_i is defined as:

$$\bar{S} = (S, S^1, S^2, ..., S^{m-1}),$$

$$E_i = (e_i, e_i^1, e_i^2, ..., e_i^{m-1}).$$

This gives us a WLWE problem. Then we can use a MLWE solver to solve this problem. The first column of the solution $\bar{S}$ also gives use the solution for the LWE problem: S. This finishes the proof.

The same is true with the SMLWE problem, which is just as hard as a LWE problem.

The protocol can be set up as follows.

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

- Alice and Bob will first publicly select F_q, n and a $n \times n$ matrix M over F_q uniformly and randomly, where $q \approx n^3$ and the error distribution $\bar{\kappa}$ to be a distribution over $n \times n$ matrices such that each component are independent, and each component follow the same discrete error distribution κ as in the case of LWE, namely a discrete normal distribution over F_q center around 0 with standard deviation approximately $\sqrt{n}$. This setting is just like the key distribution case above. All the information above is public.
- Then each party chooses its own secret S_i as a $n \times n$ matrix chosen according to the error distribution $\bar{\kappa}$, e_i also as a $n \times n$ matrix following the error distribution $\bar{\kappa}$, but jointly choose a small prime integer t $(t << n)$

  For Alice, she computes
  $$M_A = MS_A + te_A,$$
  where t is a small integer $(t << n)$.

  For Bob, he computes
  $$M_B = M^t S_b + te_B.$$

– Both parties exchange M_i. This means both M_i are public, but certainly keep S_i and e_i secret.
– Alice computes:

$$K_A = S_A^t \times M_B = S_A^t M^t S_B + tS_a e_B.$$

Bob computes:

$$K_B = M_A^t \times S_B = S_A^t M^t S_B + te_A S_B.$$

– Both of them will perform a rounding technique to derive the shared key as follows:
  1. Bob will make a list T_1 of all positions of the entries of K_B such that these entries are in the range of $[-(q-1)/4, (q-1)/4]$ and a list T_2 of all positions which are not in the range of $[-(q-1)/4, (q-1)/4]$. Then Bob will send to Alice the list T_1.
  2. Then each party will compute the residues of these entries modular t in T_1, and for the entries not in T_1, which is in T_2, they will add $(q-1)/2$ to each entry and compute the residue modular q first (into the range of $[-(q-1)/4, (q-1)/4]$) then the residue modular t. That will give a shared key between these two users.

Again the reason that Alice and Bob can derive from K_A and K_B a shared secret to be the exchanged key via certain rounding techniques as in the case above is exactly that e_i and S_i are small, therefore K_A and K_B are close.

We call this scheme a SMLWE key exchange protocol. We can again define a similar matrix version of pairing with error problem (MPE), and this will allow us to derive the provable security of this more efficient scheme.

In term of both communication and computation efficiency, the new system is much better. Since this time they need to exchange n^2 entries in F_q, and each perform $2n^{2.8}$ computations (with Strassen fast matrix multiplication) to derive n^2 bits if $t = 2$, while, for the original one above, they exchange $2n$ entries with $2n$ computations each in F_q and derive 1 bit.

Theorem 18. *If we choose the same system parameters, namely n and q, the matrix SLWE key exchange protocol is as secure as the SLWE key exchange protocol.*

The proof follows from the fact that the SMLWE problem is as hard as the SLWE problem, since the matrix version can be viewed as just assembling multiple SLWE samples into one matrix SLWE sample.

Again, security analysis of this key exchange protocol will be essentially the same as in the case of situation above.

We note here that we can choose also rectangular matrix for the construction as long as we make sure the sizes are matching in terms of matrix multiplications, but parameters need to large enough.

3.2 Constructions based on the Ring LWE problem

Similarly we can build a key exchange scheme based on the ring learning with errors problem (RLWE) of [20]. However, in this paper, we will use a variant of the RLWE problem described in [21].

For the RLWE problem, we consider the rings $\mathcal{R} = \mathcal{Z}[x]/f(x)$, and $\mathcal{R}_q = \mathcal{R}/q\mathcal{R}$, where $f(x)$ is a degree n polynomial in $\mathcal{Z}[x]$, $\mathcal{Z}$ is the ring of integers, and q is a prime number. Here q is an odd prime and elements in $\mathcal{Z}_q = F_q = \mathcal{Z}/q$ are represented by elements: $-(q-1)/2, ..., -1, 0, 1, .., (q-1)/2$, which can be viewed as elements in $\mathcal{Z}$ when we talk about norm of an element. Any element in $\mathcal{R}_q$ is represented by a degree $n-1$ polynomial, which can also be viewed as a vector with its corresponding coefficients as its entries. For an element

$$a(x) = a_0 + a_1 x + ... + a_{n-1} x^{n-1},$$

we define

$$\|a\| = \max|a_i|,$$

the l_∞ norm of the vector $(a_0, a_1, ..., a_{n-1})$ and we treat this vector as an element in $\mathcal{Z}^n$ and a_i an element in $\mathcal{Z}$.

The $\mathrm{RLWE}_{f,q,\chi}$ problem is parameterized by an polynomial $f(x)$ of degree n, a prime number q and an error distribution χ over $\mathcal{R}_q$. It is defined as follows.

Definition 19. *Let the secret s be an element in $\mathcal{R}_q$, a uniformly chosen random ring element. The problem is to find s, given any polynomial number of samples of the pair*

$$(a_i, b_i = a_i \times s + e_i),$$

where a_i is uniformly random in $\mathcal{R}_q$ and e_i is selected following the error distribution χ.

The hardness of such a problem is based on the fact that the b_is are computationally indistinguishable from uniform in $\mathcal{R}_q$. One can show [20] that solving the $\mathrm{RLWE}_{f,q,\chi}$ problem above is known to give us a quantum algorithm that solves short vector problems on ideal lattices with related parameters. We believe (or assume) that the latter problem is exponentially hard.

We will here again use the fact [3, 20] that the $\mathrm{RLWE}_{f,q,\chi}$ problem is equivalent to a variant where the secret s is sampled from the error distribution χ rather than being uniform in $\mathcal{R}_q$ and the error element e_i are multiples of some small integer t that is relatively prime to q.

Here we will consider the RLWE problem with specific choices of the parameters.

- We choose $f(x)$ to be the cyclotomic polynomial $x^n + 1$ for $n = 2^u$, a power of two;
- The error distribution χ is the discrete Gaussian distribution $D_{Z^n,\sigma}$ for some $n >> \sigma > \omega(\sqrt{\log n}) > 1$;
- $q = 1 \pmod{2n}$ and q a polynomial of n and $q \approx n^3$;
- t a small prime and $t << n << q$.

Next we will present two key facts in the $\mathrm{RLWE}_{f,q,\chi}$ setting defined above, which are needed for our key exchange scheme [12, 16, 17, 21].

Lemma 20. *The length of a vector drawn from a discrete Gaussian of with standard deviation σ is bounded by σn, namely,*

$$Pr(\|X\| > \sigma n) \leq 2^{-n+1},$$

for X chosen according to χ.

Lemma 21. *The multiplication in the ring $\mathcal{R}_q$ increases from the norms of the constituent elements in a reasonable scale, that is,*

$$\|X \times Y(mod \;\; f(x))\| \leq n\|X\|\|Y\|,$$

for $X, Y \in \mathcal{R}_q$ and the norm is the l_∞ norm defined above.

Note again, these lemmas are valid with conditions defined by the $\mathrm{RLWE}_{f,q,\chi}$ setting.

With the $\mathrm{RLWE}_{f,q,\chi}$ setting above, we are now ready to have two parties Alice and Bob to do a key exchange over an open channel. It goes as follows.

1. Alice and Bob will first publicly select all the parameters for the $\mathrm{RLWE}_{f,q,\chi}$ including $q(\approx n^3)$, n, $f(x)$ and χ. In addition, they will select a random element M over $\mathcal{R}_q$ uniformly. All the information above is public.

2. Then each party chooses its own secret s_i as an element in $\mathcal{R}_q$ according to to the error distribution χ , and e_i independently also as an element following the error distribution χ, but jointly choose a small prime integer t ($t << n$)
   For Alice, she computes
$$M_A = Ms_A + te_A,$$
   where t is a small integer ($t << n$).
   For Bob, he computes
$$M_B = Ms_b + te_b.$$
3. Both parties exchange M_i. This means both M_i are public, but certainly keep s_i and e_i secret.
4. Alice computes:
$$K_A = s_A \times M_B = s_A M s_B + te_B s_A.$$
   Bob computes:
$$K_B = M_A \times s_B = s_A M s_B + te_A s_B.$$
5. Both of them will perform a rounding technique to derive the shared key as follows:
   (a) Bob will then make a list of size n, and this list consists of pairs in the form of (i, j), where $i = 0, ..., n-1$, and $j = 1$ if the x^i coefficient of K_B is in the range of $[-(q-1)/4, (q-1)/4]$, otherwise $j = 0$.
   (b) Then Bob will send this list to Alice. Then each will compute the residue of the corresponding entries modular t in the following way:
   for an element of the list (i, j),
   1) if $j = 1$, each will compute the i-th entry of K_A and K_B modular t respectively,
   2) if $j = 0$, each will add $(q-1)/2$ to the i-th entry of K_A and K_B modular q back to range of $[-(q-1)/4, (q-1)/4]$, then compute the residues modular t.

That will give a shared key between these two users. We call this scheme a RLWE key exchange protocol.

From Lemma 14 and Lemma 15, we can deduce that there is a very low probability of failure of this key exchange protocol.

We note here that the commutativity and the associativity of the ring $\mathcal{R}_q$ play a key role in this construction.

In terms of security analysis, we can show the provable security of the scheme following the hardness of the RLWE$_{f,q,\chi}$ problem by using a similar PE problem over the ring R_q.

Definition 22. *Assume that we are given*

– *a random element M in R_q, prime integers t, q and the error distribution χ with parameters selected as in the RLWE$_{f,q,\chi}$ above;*
– *$M_A = Ms_A + te_A$ and $M_B = Ms_B + te_B$, where e_i follows the error distribution χ and s_i also follows the error distribution χ;*
– *and the fact that $(K_B)_i$, the coefficients x^i of $K_B = M_A \times s_B = s_A M s_B + te_A s_B$ is in the range of $[-(q-1)/4, (q-1)/4]$ or not;*

the problem is to find an algorithm to derive K_B (or K_A) modular t or $K_B + (q-1)/2$ (or $K_A + (q-1)/2$) modular q (into the range of [-(q-1)/4, (q-1)/4]) and then modular t with high probability.

We call such a problem a pairing with error problem over a ring(RPE).

Since, it is nearly a parallel extension of the proof of the provable security of the case of SLWE key exchange protocol to the RLWE key exchange protocol, we will leave out the details. We conclude that the RLWE key exchange protocol is provable secure based on the hardness of the RLWE$_{f,q,\chi}$ problem.

With the same parameters q and n, this scheme is even more efficient due to the possibility doing fast multiplication over the ring $\mathcal{R}_q$ using FFT type of algorithms.

4 Conclusion and Discussion

In this paper, we use a variant of LWE problem with matrices, a simple and direct extension of the original LWE to the case of matrices, to build a new, simple and provably secure key exchange protocol. We also extend the construction to the RLWE case. A key ingredient of the construction relies on the definition of a so-called pairing with error problem, which provide the bridge to establish the provable security of our key exchange protocol.

The work in this paper can be attributed to the understanding that the basic idea behind the LWE problem itself can be viewed as certain form of inner product with small errors that somehow can be eliminated for certain applications. Our construction can be viewed as an extension of this idea to the case of a bilinear pairing, namely a pairing of bilinear forms with errors. In addition, the reason why the scheme works well actually depends on the associativity and the commutativity of the multiplications in both the commutative rings (the RLWE problem) and the non-commutative rings (the SMLWE problem and the RLWE). We believe that exploring further algebraic properties of the rings could yield even more interesting cryptographic protocols, such as certain homomorphic properties over non-commutative operations over matrices.

Using the same idea, we are now in the process finishing works in building simple and provable secure identity-based encryption systems and scalable key distribution systems for large networks.

The basic design of the work was first finished in Aug. 2011 at Cincinnati and was submitted to University of Cincinnati IP Office. A provisional patent application was filed in April 2012.

5 Acknowledgment

I would like to thank Johannes Buchmann, Adi Shamir, , Xiaoyun Wang, Christopher Peikert, Oded Regev, Michael Schneider, Patrick Schmidt and Chendong Tao,for useful discussions. This work was also presented in a retreat of the research group of Professor Johannes Buchmann in June 2012 and in the Germany-Taiwan workshop on cryptography in National Chung-Hsing University in Taiwan in Nov. 2012. This work was partially supported by NSF China Grant # 60973131.

References

1. Shweta Agrawal, Dan Boneh, Xavier Boyen: Efficient Lattice (H)IBE in the Standard Model. EUROCRYPT 2010: 553-572
2. Shweta Agrawal, Xavier Boyen, Vinod Vaikuntanathan, Panagiotis Voulgaris, Hoeteck Wee: Fuzzy Identity Based Encryption from Lattices. IACR Cryptology ePrint Archive 2011: 414 (2011)
3. Benny Applebaum, David Cash, Chris Peikert, Amit Sahai; Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems. Advances in Cryptology-CRYPTO 2009, LNCS, P595-618, Springer 2009
4. Mihir Bellare, Eike Kiltz, Chris Peikert, Brent Waters: Identity-Based (Lossy) Trapdoor Functions and Applications. EUROCRYPT 2012: 228-245
5. Carlo Blundo, Alfredo De Santis, Amir Herzberg, Shay Kutten, Ugo Vaccaro, Moti Yung: Perfectly-Secure Key Distribution for Dynamic Conferences. CRYPTO 1992: 471-486
6. A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. Journal of the ACM, 50(4):506?19, 2003.
7. Johannes Buchmann, Daniel Cabarcas, Jintai Ding, Mohamed Saied Emam Mohamed: Flexible Partial Enlargement to Accelerate Groener Basis Computation over F2. AFRICACRYPT 2010: 69-81
8. Ran Canetti, Dana Dachman-Soled, Vinod Vaikuntanathan, Hoeteck Wee: Efficient Password Authenticated Key Exchange via Oblivious Transfer. Public Key Cryptography 2012: 449-466
9. Don Coppersmith, Shmuel Winograd, Matrix multiplication via arithmetic progressions, Journal of Symbolic Computation - Special issue on computational algebraic complexity archive Volume 9 Issue 3, March 1990 Pages 251-280
10. W. Diffie and M. E. Hellman, New Directions in Cryptography IEEE Transactions on Information Theory, vol. IT-22, Nov. 1976, pp: 644?54.
11. Atsushi Fujioka, Koutarou Suzuki, Keita Xagawa, Kazuki Yoneyama: Strongly Secure Authenticated Key Exchange from Factoring, Codes, and Lattices. Public Key Cryptography 2012: 467-484
12. Craig Gentry. Fully homomorphic encryption using ideal lattices. In Michael Mitzenmacher, editor, STOC, pages 169C178. ACM, 2009.
13. Shafi Goldwasser, Yael Kalai, Chris Peikert, Vinod Vaikuntanathan; Robustness of the learning with errors assumption Authors. Innovations in Computer Science (ICS), 2010
14. Jonathan Katz, Vinod Vaikuntanathan: Round-Optimal Password-Based Authenticated Key Exchange. TCC 2011: 293-310
15. Jonathan Katz, Vinod Vaikuntanathan: Smooth Projective Hashing and Password-Based Authenticated Key Exchange from Lattices. ASIACRYPT 2009: 636-652
16. Vadim Lyubashevsky and Daniele Micciancio. Generalized compact knapsacks are collision resistant. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, ICALP (2), volume 4052 of Lecture Notes in Computer Science, pages 144C155. Springer, 2006.
17. Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on gaussian measures. SIAM J. Comput., 37(1):267C302, 2007.
18. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM, 56(6):34, 2009. Preliminary version in STOC?5.
19. Oded Regev, The Learning with Errors Problem (Invited Survey), CCC, pp.191-204, 2010 25th Annual IEEE Conference on Computational Complexity, 2010
20. Vadim Lyubashevsky, Chris Peikert, Oded Regev, On ideal lattices and learning with errors over rings In Eurocrypt 2010
21. Kristin Lauter and Michael Naehrig and Vinod Vaikuntanathan, Can Homomorphic Encryption be Practical?, Cryptology ePrint Archive, Report 2011/405, 2011, http://eprint.iacr.org,
22. Zvika Brakerski, Vinod Vaikuntanathan, Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages. CRYPTO 2011: 505-524, LNCS, Springer, 2011
23. Zvika Brakerski, Vinod Vaikuntanathan, Efficient Fully Homomorphic Encryption from (Standard) LWE. FOCS 2011: 97-106
24. R. Vershynin, Introduction to the non-asymptotic analysis of random matrices. Chapter 5 of the book Compressed Sensing, Theory and Applications, ed. Y. Eldar and G. Kutyniok. Cambridge University Press, 2012. pp. 210–268.