Applied Mathematics & Information Sciences
An International Journal

Post-Quantum Secure Hybrid Signcryption from Lattice Assumption

Fenghe Wang[1], *Yupu Hu*[1] *and Chunxiao Wang*[2]

[1] Key Laboratory of Computer Networks and Information Security, Xidian University, Xi'an, China
[2] Department of Mathematics and Physics, Shandong Jianzhu University, Jinan, China

Received: Received April 22, 2011; Revised August 02, 2011, Accepted August 23, 2011
Published online: 1 January 2012

Abstract: Motivated by the demand to have secure signcryption scheme, even in quantum era, the concept of signcryption tag-KEM (key encapsulation machine) is extended to lattice cryptography in this paper. A lattice-based hybrid signcryption scheme is proposed by building a lattice-based signcryption tag-KEM. Based on the hardness of the learning with errors problem and the short integer solution problem, the proposed hybrid signcryption is provable secure in the random oracle model. Furthermore, according to the quantum intractability assumption about lattice problem, the proposed hybrid signcryption scheme is secure even on quantum computers.

Keywords: Tag-KEM, Lattice, Learning with Errors Problem, Short Integer Solution Problem, Pre-sample Function.

1. Introduction

Signcryption concept was first proposed by Zheng [18] in 1997, which provides confidentiality, message authenticity in a single logical step. Since signcryption schemes are more efficient than a direct composition of encryption and signature schemes, following Zhengs pioneering work, a lot of new signcryption schemes have been proposed [3,11,12,10,13]. There are two different ways to construct signcryption schemes, one is public key signcryption and the other is hybrid signcryption. In a public key signcryption scheme both encryption and signature are in public key setting. In a Hybrid signcryption scheme, which consists signcryption KEM (SKEM) and signcryption DEM(SDEM), the signature is in public key setting and encryption is in symmetric key setting. If the message is very large, a hybrid signcryption is more efficient than a public key signcryption. The hybrid signcryption has been studied by Dent[6,7]. Nevertheless, the construction of SKEM+SDEM have complex verification-decryption (unsigncryption) algorithms, which need verify a link between message, key and encapsulation. In 2006, Bjørstad and Dent adapted the tag-KEM + DEM construction to signcryption which achieve simpler scheme descriptions and better generic security reductions[4]

All above schemes are based on number theorem assumption and not secure in quantum era, for, according to the seminal work of Shor [17], that both factorization and discrete logarithms can be solved by quantum algorithm in polynomial time. As a result, it is necessary to construct post-quantum secure signcryption schemes to resist quantum attack in quantum-era. Fortunately, there are still many hard problems can resist quantum attack and those problems are alternatives for the post-quantum era in cryptography, for example, lattice problems. There are no already-existed efficient quantum algorithms for the worst-case lattice problems. Recently, lattice cryptography has gained a lot of attentions and many lattice-based primitives are proposed[9,5,1,16,2,14]. Utilizing hybrid techniques, Peikert constructs an efficient public-key encryption algorithm against chosen-ciphertext-attack over lattice[15]. In the same paper, an efficient technique was proposed to shorten the ciphertext expansion of his encryption algorithm which enhance the efficiency of the encryption algorithm. Another important primitive in lattice-based cryptography is pre-sample function (PSF) which is proposed in 2008 by Gentry et al.[9]. According to PSF, a short basis of a random lattice can be regarded as a trapdoor for lattice-based cryptography systems. As a result, PSF is an important tools for building lattice-

* Corresponding author: e-mail: fenghe2166@163.com

based signature[5,9] and lattice-based ID-based encryption[1,9]. Even those achievements in lattice-based public-key cryptography, many open problems remain unsolved, such as how can we design a lattice-based signcryption scheme? As far as our known, the cryptographic community failed to witness a realization of post-quantum signcryption scheme based on lattice theory.

In this paper, we extend signcryption tag-KEM to lattice cryptography and design a signcryption tag-KEM by lattice tools. In the proposed signcryption tag-KEM, we use the main ideas about how to achieve the CCA security (chosen-ciphertext attack) in literaturec[15] to design the encapsulation algorithm. In our encapsulation algorithm, the PSF is used to authenticate the symmetric key of the signcryption DEM. Furthermore, the embedment of the authentication information about symmetric key is so perfect that the output length of the encapsulation is not enlarged. If we regard hash function as a random oracle, under the lattice problem intractability assumption, we prove that the proposed signcryption tag-KEM is IND-CCA2 and sUF-CMA secure in Bjørstad's security models(see section 2.4). Moreover, we point out that, if we use a relatively small modulus q' to make the outputs of the encapsulation algorithm more "coarse" just as shown by peikert in literature [15], the length of the ciphertext can be more shorter and the efficiency of encapsulation can be more higher.

2. Preliminaries

2.1. Notations

In the following paper, we use bold alphabets like $\mathbf{A}$ or $\mathbf{A}_i$ to denote matrixes. Bold lowercase letters are used to denote vectors. $\omega(f(n))$ denotes a function which grows faster than $cf(n)$ for any constant $c > 0$. $poly(n)$ is used to denote an unspecified function $f(n) = O(n^c)$ for constant c. An arrow $\leftarrow$ is denoted the output of some algorithms. When calling a trapdoor for a lattice, we mean a basis with a shorter Euclidean norm than a normal basis, we refer to [9,15] for the concrete specific definition of the "shorter" basis. We use D_α to denote a Gaussian distribution with Gaussian's parameter α. In this paper, when it comes to vectors, we always consider its Euclide norm, which writes as $||\cdot||$.

2.2. Lattice and Lattice Problems

Definition 1.*(Lattice) Given n linearly independent vectors $\boldsymbol{B} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n, \boldsymbol{b}_i \in R^m\}$, a lattice Λ generated by the vector set $\boldsymbol{B}$ is a vector set which is $\boldsymbol{Bc} = \{c_1\boldsymbol{b}_1 + c_2\boldsymbol{b}_2 + \cdots + c_n\boldsymbol{b}_n\}, c_i \in Z$, denoted by $\Lambda(\boldsymbol{B})$. In this case, $\boldsymbol{B}$ is a basis of the lattice $\Lambda(\boldsymbol{B})$.*

In this paper, we will fucus on a special class of q-ary lattices which can be more easily described by a matrix which functions like a parity check matrix in coding theory. With some integers (q, m, n) and a matrix $\mathbf{A} \in Z_q^{n\times m}$, we define the following m-dimensional q-ary lattice

$$\Lambda_q^{\perp}(\mathbf{A}) = \{\mathbf{e} \in Z_q^m, \mathbf{Ae} = 0(modq)\}.$$

We introduce the shortest integer solution (SIS) problem and learning with errors problem over lattice and we describe those problems in some forms suitable for our applications in this paper.

For an integer n, let $q = q(n), m = m(n)$, $\mathbf{s} \in Z_q^n$ and χ be an error distribution over Z_q^m. Let $A_{(\mathbf{s},\chi)}$ be a distribution obtained by computing $\{\mathbf{A}, \mathbf{y} = \mathbf{A}^\top\mathbf{s} + \mathbf{e}\}$ where $\mathbf{A} \in Z_q^{n\times m}$ is chosen uniformly and randomly and $\mathbf{e}$ is distributed according to χ. Then, we give the definition of the learning with errors problem.

Definition 2.*(LWE Problem).The learning with errors problem is defined as follows:*

The decision variant of the LWE problem is to distinguish $A_{(\boldsymbol{s},\chi)}$ from the uniform distribution over $Z_q^{n\times m} \times Z_q^m$.

The goal of the standard LWE problem is to find $\boldsymbol{s}$ with overwhelming probability by access to any desired $poly(n)$ number of samples from $A_{\boldsymbol{s},\chi}$ for some arbitrary $\boldsymbol{s} \in Z_q^n$.

For $\alpha > 0$, the errors distribution in a standard LWE problem denoted Φ_α is defined by the following process: Samples m numbers $\eta_1, \eta_2, \cdots \eta_m$ according to D_α, computes $e_i = \lfloor q\eta_i \rceil (modq)$. Then let $\mathbf{e} = (e_1, \cdots, e_m)$ be an error vector in the LWE problem. As described in [8], even the distribution of error vector is more "wider" than in above standard setting, the hardness of LWE problem is also satisfied. And we refine this fact as the following lemma which is important for our constructions. Let $LWE_{(m,q,\alpha)}$ be an abbreviation for LWE problem.

Lemma 1. *Hardness of $LWE_{(m,q,\alpha)}$ implies hardness of $LWE_{(m,q,D_{(Z^m,\alpha)})}$*

The proof of Lemma 1 refers to [8]. By Lemma 1, we embed the signature of symmetric key into the encapsulation algorithm without enlarging the encapsulation length.

Based on the LWE problem, we define a trapdoor one-way function $\mathbf{y} = \mathbf{A}^\top\mathbf{s} + \mathbf{e}(modq)$ whose trapdoor is a "short" basis $\mathbf{T}$ of lattice $\Lambda_q^{\perp}(\mathbf{A})$. Precisely, given $\mathbf{y} = \mathbf{A}^\top\mathbf{s} + \mathbf{e}(modq)$, we can find $(\mathbf{s}, \mathbf{e})$ by short basis $\mathbf{T}$ as follows:

1. $\mathbf{Ty} = \mathbf{Te}(modq)$. Due to the fact that both $\mathbf{T}$ and $\mathbf{e}$ with short norm, then, with overwhelming probability, $\mathbf{Te}(modq) = \mathbf{Te}$

2.$\mathbf{e} = \mathbf{T}^{-1}\mathbf{Te}(modq)$

3. By a routine calculation, we find vector $\mathbf{s}$.

Definition 3.*(SIS Problem). The SIS problem is defined as follows: Given a uniformly-distributed random matrix $\boldsymbol{A} \in Z_q^{n\times m}$ for $m = poly(n)$, and a real number β, find a nonzero integral vector$v$ such that $\boldsymbol{Av} = 0(modq)$ and $||\boldsymbol{v}|| < \beta$.*

The following propositions provide two basic tools for our constructions. More details about the following propositions are found in literatures [9,2].

Proposition 1.*[9] Offered a short basis* $\boldsymbol{B}$ *of an n-dimensional lattice* $\Lambda_q^{\perp}(\boldsymbol{A})$*, a parameter* $s > ||\widetilde{\boldsymbol{B}}|| \cdot \omega(\sqrt{logn})$*,* $||\widetilde{\boldsymbol{B}}|| \leq m^{1+\varepsilon}$ *and a vector* $\boldsymbol{y} \in Z_q^n$ *, There is a probabilistic polynomial-time (PPT) algorithm, denoted by* PreSample*, which outputs a vector* $\boldsymbol{e} \in Z_q^m$ *from a distribution that is statistically close to discrete Gaussian distribution. Moreover , This output* $\boldsymbol{e}$*, with high probability, meets requirements* $\boldsymbol{Ae} = \boldsymbol{y}(modq)$ *and* $||\boldsymbol{e}|| \leq s\sqrt{m}$ *.*

Proposition 2.*[2] For any prime* $q = poly(n)$ *and any* $m > 5nlogq$ *, there is a PPT algorithm, called random lattice sample algorithm, which, on input* 1^n*, outputs a matrix* $\boldsymbol{A} \in Z_q^{n \times m}$ *where the distribution of* $\boldsymbol{A}$ *is statistically close to the uniform distribution, and a full-rank set* $\boldsymbol{S} \subset \Lambda_q^{\perp}(\boldsymbol{A})$*, where* $||\boldsymbol{S}|| \leq m^{2.5}$*. And moreover, the set* $\boldsymbol{S}$ *can be efficiently converted to a short basis* $\boldsymbol{T}$ *of the lattice* $\Lambda_q^{\perp}(\boldsymbol{A})$*.*

2.3. Hybrid Signcryption

A hybrid signcryption scheme consists two parts signcryption tag-KEM and signcryption DEM which defined as follows respectively.[4]

Definition 4.*A signcryption tag-KEM is defined as tuple of six algorithms.*

—A probabilistic common parameter generation algorithm, denoted by Gen_c*, takes as input a security parameter* 1^k*, and outputs all the global information* I *needed by users of the scheme, such as choice of groups or hash functions.*

—A probabilistic sender key generation algorithm, Gen_s*, which takes as input the global information* I*, and returns a private/public keypair* (sk_S, pk_S) *that is used to send signcrypted messages.*

—A probabilistic receiver key generation algorithm, Gen_r*, which takes as input the global information* I*, and outputs a private/public keypair* (sk_R, pk_R) *that is used to receive signcrypted messages.*

—A probabilistic symmetric key generation algorithm, Sym*. It takes as input the private key of the sender* sk_S *and the public key of the receiver* pk_R*, and outputs a symmetric key K together with internal state information* ω*.*

—A probabilistic key encapsulation algorithm, Encap*, which takes* ω *and and an arbitrary tag* τ *as inputs, and returns an encapsulation E.*

—A deterministic key decapsulation algorithm , Decap*, which takes as input the public key of the sender* pk_S *, the private key of the receiver* sk_R*, an encapsulation E and an arbitrary tag* τ*. It outputs either a symmetric key K or the unique error symbol.*

Definition 5.*A signcryption DEM consists two polynomial-time algorithms*

—A deterministic encryption algorithm, Enc, which takes as input a message m a symmetric key K of some pre-determined length, and outputs an encryption $C = Enc_K(m)$ *of that message.*

—A deterministic decryption algorithm, Dec, which takes as input a ciphertext C and a symmetric key K of some pre-determined length, and outputs either a message $m = Dec_K(C)$ *or the error symbol.*

Definition 6.*Suppose that (*Gen_c*;* Gen_s*;* Gen_r*;* Encap*;* Sym Decap*) is a signcryption tag-KEM, (*Enc*;* Dec*) is a signcryption DEM, and that, for all security parameters k, the keys produced by the signcryption tag-KEM are of the correct length to be used by the signcryption DEM. We may then construct a hybrid signcryption scheme as following*

—The key generation algorithms of the hybrid signcryption scheme are given by the key generation algorithms ((Gen_c, Gen_r, Gen_s)*) of the signcryption KEM,*

—The signcryption algorithm is given by the following computation:

1. Set $(K, C_1) = encrap(sk_S, pk_R, m)$*; 2. Set* $C_2 = Enc_K(m)$ *;*

3. Outputs (C_1, C_2)

—The designcryption algorithm is given by the next steps:

1. $K = Decap(pk_S, sk_R, C_1)$*(If outputs an errors symbol, stops);*

2. $m = Dec_K(C_2)$*;*

3.If Decap algorithm outputs valid then accepts message. Otherwise, rejects it.

2.4. Security of Hybrid Signcryption

A signcryption scheme should be sure against choose ciphertext attack (IND-CCA2). We define the IND-CCA2 game as a game played between a challenger and a three stages attacker $\mathcal{A} = (\mathcal{A}_1; \mathcal{A}_2, \mathcal{A}_3)$.[4]

Definition 7.*For a given security parameter k, the IND-CCA2 game is played as follows.*

1.The challenger generates some public parameters $I = Gen_c(1^k)$*, a sender key-pair* $(pk_S; sk_S)$ *and a receiver key-pair* $(pk_R; sk_R)$*.*

2. The attacker runs $\mathcal{A}_1$ *with input* (pk_S, pk_R)*. During this execution,* $\mathcal{A}_1$ *can query an symmetric key generation, an encapsulation oracle, a decapsulation oracle.* $\mathcal{A}_1$ *terminates by outputting some state information* $state_1$*.*

3.The challenger computes as follows:

(a) Set $(K_0; \omega) = Sym(sk_S; pk_R)$*.*

(b) Randomly generate a symmetric K_1 *of the same length as* K_0*.*

(c) Randomly generate a bit $b \in \{0, 1\}$*.*

4.The attacker executes $\mathcal{A}_2$ *on the input* $(K_b, state_1)$ *. During its execution* $\mathcal{A}_2$ *can query the same oracle as*

previously. $\mathcal{A}_2$ terminates by outputting some state information $state_2$ and a tag τ.

5.The challenger computes a challenge encapsulation $E = Encap(\omega; \tau)$.

6.The attacker runs $\mathcal{A}_3$ with input $E, state_2$. During its execution, $\mathcal{A}_3$ may access the same oracles as previously with the exception that $\mathcal{A}_3$ cannot query the decapsulation oracle on the input$E, state_2$. $\mathcal{A}_2$ terminates by outputting a guess b' for b.

The attacker wins the game if $b = b'$. $\mathcal{A}$'s advantage in winning the IND-CCA game is defined to be: $pr(b = b') - 1/2$.

Definition 8.*A signcryption tag-KEM is said to be IND-CCA2 secure, if, for any adversary $\mathcal{A}$, the advantage of $\mathcal{A}$ in the IND-CCA2 game (definition7) is negligible with respect to the security parameter 1^k.*

In order to achieve the authentication, a secure signcryption tag-KEM should provide strong existential unforgeability against choose message attack (sUF-CMA)which is specified by a sUF-CMA game between the challenger and the adversary.

Definition 9.*The sUF-CMA game is defined as follows:*

1. The challenger generates a set of global parameters $I = Com(1^k)$, a sender keypair $(sk_S; pk_S) = Key_S(I)$ and a receiver keypair $(sk_R; pk_R) = Key_R(I)$.

2. The adversary $\mathcal{A}$ is run on the input $(I; pk_S; sk_R; pk_R)$. During its execution, $\mathcal{A}$ may access the symmetric key generation and encapsulation oracles as were defined in the previous game. $\mathcal{A}$ terminates by returning an encapsulation E and a tag τ .

The adversary wins the games if E and τ can be decapsulation validly, and τ never be queried in the encapsulation oracles. The advantage of $\mathcal{A}$ is defined as the probability of he wins in the games.

Definition 10.*A signcryption tag-KEM is said to be sUF-CMA, if, for any adversary $\mathcal{A}$, the advantage of $\mathcal{A}$ in the sUF-CMA game(definition9) is negligible with respect to the security parameter 1^k.*

Definition 11.*A signcryption tag-KEM is said to be secure if it is IND-CCA2 and sUF-CMA secure.*

Lemma 2.*[4]A hybrid signcryption scheme constructed from a signcryption tag-KEM and a signcrypt DEM. If the signcryption tag-KEM is IND-CCA2 secure and the DEM is IND-PA secure, then hybrid signcryption is IND-CCA2 secure.*

Lemma 3.*[4] A hybrid signcryption scheme constructed from a signcryption tag-KEM and a DEM. If the signcryption tag-KEM is sUF-CMA secure, then the hybrid signcryption scheme is also sUF-CMA secure.*

3. The Proposed Lattice-based Hybrid Signcryption Scheme

3.1. Lattice-based Signcryption tag-KEM

—Gen_c Let n be a main secure parameter. $q \geq 2$ and $m = (1+\delta)nlgq$ for some constant $\delta > 0$. Let $l = l(n) \geq 1$ be integers and bounded by $poly(n)$. A Gaussian parameters $s = \widetilde{L}\omega(\sqrt[2]{logn})$ where $\widetilde{L} = O(\sqrt{nlgq})$. In our construction, we need two secure hash functions as follows: $h_1 : \{0,1\}^* \to Z_q^n$, $h_2 : Z_q^m \to \{0,1\}^l$

—Gen_R To generate the receiver's public/private key, by proposition 2, the receiver generates two matrix $\mathbf{B}_{10} \in Z_q^{n\times m}$ and $\mathbf{B}_{11} \in Z_q^{n\times m}$ with its trapdoor $\mathbf{T}_{10} \in Z_q^{m\times m}$, $\mathbf{T}_{11} \in Z_q^{m\times m}$, respectively. Randomly chooses $2(l-1)$ matrixes $\mathbf{B}_{ib} \in Z_q^{n\times m}$ where $2 \leq i \leq l$ and $b \in \{0,1\}$. Then, for $i \in [l]$ and $b \in \{0,1\}$, $pk_R = \{\mathbf{B}_{ib} | i \in [l], b \in \{0,1\}\}$ and $sk_R = (\mathbf{T}_{10}, \mathbf{T}_{11})$.

—Gen_S Sender uses the random lattice sample algorithm in proposition 2 to generate a matrix $\mathbf{A} \in Z_q^{n\times m}$ with its trapdoor $\mathbf{T} \in Z_q^{m\times m}$, let $pk_S = \mathbf{A}$ and $sk_S = \mathbf{T}$.

—*Sym*

1. Randomly chooses $\mathbf{s} \in Z_q^n$; 2. Computes $K = h_2(\mathbf{s})$;

3. Set $\omega = (\mathbf{s}, pk_R, sk_S)$.

—*Encap*

1. Randomly chooses $\tau \in \{0,1\}^l$ and Computes $h_1(\mathbf{s}, \tau)$. Utilizing the PreSample algorithm in proposition 1, sender computes $\mathbf{e}_1 \leftarrow PreSample(\mathbf{A}, \mathbf{T}, h_1(\mathbf{s}, \tau))$.

2. Chooses $l-1$ random errors vectors $\mathbf{e}_i$ according to Φ_α for $i \geq 2$. For $\tau = (\tau_1, \tau_2, \cdots, \tau_l)$, computes $\mathbf{b}_i = \mathbf{B}_{i\tau_i}^\top \mathbf{s} + \mathbf{e}_i (mod q)$. Let $\mathbf{b} = (\mathbf{b}_1, \cdots, \mathbf{b}_l)$.

Then, the outputs of the *Encap* algorithm are $(\mathbf{b}, \tau)$

—*Decap*:

The receiver performs the following steps:

1. Parses $\mathbf{b} = (\mathbf{b}_1, \mathbf{b}_2 \cdots, \mathbf{b}_l)$ where $\mathbf{b}_i \in Z_q^m$ for $1 \leq i \leq l$. If $\mathbf{b}$ can not be parsed in this way, rejects it.

2. Computes $\mathbf{s}$ and $\mathbf{e}_1$ from $\mathbf{b}_1$ with the help of the trapdoor $\mathbf{T}_{1\tau_1}$ which can be fixed by the first bit of τ.

3. Computes $h_1(\mathbf{s}, \tau)$, checks $\mathbf{A}\mathbf{e}_1 = h_1(\mathbf{s}, \tau)$ and $||\mathbf{e}_1|| \leq s\sqrt{m}$. Otherwise, rejects it.

4. Computes $(\mathbf{b}_i - \mathbf{B}_{i\tau_i}^\top \mathbf{s})(mod q)$ for every i, and checks its norm is less than $s\sqrt{m}$. Otherwise, rejects it.

5. Computes $K = h_2(\mathbf{s})$ as a symmetric key of the signcryption DEM.

The correctness of the proposed signcryption KEM is easily verified and we overlap it.

3.2. Signcryption DEM

—*Enc* Let $M \in \{0,1\}^l$ be a message, Computes $K \oplus M = c(mod 2)$.

—*Dec* Computes $K \oplus c = M(mod 2)$.

And then, by definition 6, we can build a lattice-based hybrid signcryption scheme from the proposed signcryption tag-KEM and signcryption DEM.

*Remark.*According to literature [15], we can use a relatively small modulus $q'(< q)$ to make the vector **b** more "coarse" to shorten the encapsulation length and enhance the efficiency of the proposed signcryption tag-KEM. In this case, the encapsulation length will be $lmlogq'$ which is far shorter than $lmlogq$ of the original length. For simplicity, we refrain us to give more details about this techniques. Readers can find the details about this technique in literature [15].

4. Analysis of The Proposed Scheme

Theorem 1.*The proposed signcryption tag-KEM is IND-CCA2 secure under the hardness of the learning with errors problem.*

Proof. The proof proceeds in a sequence of games where the first game is identical to the IND-CCA2 game in Definition 7. In the last game of the sequence, the adversary has advantage zero. We show that a PPT adversary cannot distinguish between the games which will prove that the adversary has negligible advantage in winning the original IND-CCA2 game. The hardness of the LWE problem is used in proving that Game 2 and Game 3 are indistinguishable. Let X_i be the events that $b = b'$ in game *i*

Game 0 This is the original IND-CCA2 game of the tag-KEM signcryption between the adversary and the challenger.

Game 1 In this Game, if the adversary queries the decapsulation oracle for a tag τ^*(in step 2,4,6), then outputs an errors symbol and aborts. The remainder of the Game is unchanged.

Except τ^* is queried for decapsulation, Game 0 and Game 1 are unchanged. τ^* is queried with probability $1/2^l$. Then $P(X_0) - P(X_1) \leq 1/2^l = nelg(n)$.

Game 2 In this game, the Gen_R algorithm is modified as follows. Let $\mathbf{B}_{i\tau_i^*}$ be random matrixes, and other $\mathbf{B}_{i(1-\tau_i^*)}$ with is trapdoor $\mathbf{T}_{i(1-\tau_i^*)}$. The remainder steps in this game are just like game 1.

By Proposition 2, all $\mathbf{B}_{ib}$ can be seemed as random. So, $P(X_2) - P(X_1) = negl(n)$. Then Game 1 and Game 2 is undistinguished.

Game 3 It is as same as Game 3 but, challenger outputs the challenge encapsulation $E = (\mathbf{b}, \tau^*)$, where vector **b** are chosen uniformly in Z_q^{lm}.

Then, by using the hardness of the learning with errors problem, the game 2 and game 3 are undistinguished. Otherwise, if there is a adversary can distinguish the Game 2 and Game 3, then we construct a algorithm $\mathcal{B}$ to solve the decision variant of the LWE problem. The algorithm $\mathcal{B}$ performs as follows:

1. Runs the $Gen_c(1^k)$, Gen_S and Gen_R as in Game 2.

2. Responds adversary's query for an symmetric key generation oracle, an encapsulation oracle, a decapsulation oracle as follows:

—*sym oracle* Chooses an random vectors **s**, randomly choose a $K \in \{0,1\}^l$. Let $\omega = (sk_S, pk_R)$, stores $(\mathbf{s}, \omega, K)$. Gives K to the adversary as an answer.

—*encap oracle* If $\tau = \tau^*$, then let **b** be uniform (Game 3), Otherwise, chooses l short vectors randomly as the vectors $\mathbf{e}_1, \cdots, \mathbf{e}_l$ in the *encap* algorithm and uses the encap algorithm to compute the encapsulation.

—*decap oracle* If $\tau = \tau^*$, then outputs an errors symbol. Otherwise, by using the receiver's private key sk_R and sender's public key pk_S to finish the decapsulation. And sends the results to adversary. Because $\tau \neq \tau^*$, the algorithm always can make a decapsulation for adversary's query by some private key $\mathbf{T}_{i(1-\tau_i^*)}$ where $\tau_i \neq \tau_i^*$.

3. The next phases are as same as the Game 2 except the output challenge encapsulation $(\mathbf{b}, \tau^*)$ in step 5. In game 2, **b** is an LWE instance, but in Game 3, **b** is uniform.

Though above simulate phases, we already argued that when **b** is an LWE instance the adversary's view is as in Game 2. When **b** is uniform the adversary's view is as in Game 3. As a results, algorithm $\mathcal{B}$ with the same advantage to distinguish between a random vector and an LWE instance as the adversary to distinguish the Game 2 and Game 3. So we draw a conclusion that the Game 2 and Game 3 are undistinguished under the intractability assumption of the LWE problem.

Put four games together, we have $|P(X_0) - 1/2| = negl(n)$. And then the theorem has been proven.

Theorem 2.*In the random oracle model, the proposed signcryption tag-KEM is sUF-CMA secure under the short integer solution problem intractability assumption.*

Proof. If there is a adversary $\mathcal{A}$ who can attack the sUF-CMA security of proposed signcryption tag-KEM with advantage ε, and by asking q_1 h_1 queries, q_2 the symmetric key generation queries, q_3 encapsulation queries , then, there is challenger $\mathcal{B}$ can solve the shortest integer solution problem with probability ε by running Theoreom 2 the forger as a subroutine.

Supposing $\mathcal{B}$ receives an SIS problem instance $(\mathbf{A}, 2s)$ where $\mathbf{A} \in Z_q^{n\times m}$ and s be a parameters, he wants to find a short vector $\mathbf{e} \in Z_q^m$ satisfies the following properties: $\mathbf{Ae} = 0(modq)$ and $||\mathbf{e}|| \leq 2s\sqrt{m}$. To achieve this arm, $\mathcal{B}$ simulates the signcryption's generation, and begin the attack game with adversary.

Firstly, $\mathcal{B}$ generates global parameters I and $pk_R = \{\mathbf{B}_{ib} | \mathbf{B}_{ib} \in Z_q^{n\times m}, i \in [l], b \in \{0,1\}\}$, $sk_R = (\mathbf{T}_{10}, \mathbf{T}_{11})$ by running Gen_c and Gen_r. Let $pk_S = \mathbf{A}$ and challenger does't know the private key of the sender. $\mathcal{B}$ sends (pk_R, sk_R, pk_S, I) to $\mathcal{A}$ and runs the sUF-CMA game with $\mathcal{A}$. Challenger keeps 3 lists L_i for $i = 1, 2, 3$ to store the answers to the the symmetric key generation and encapsulation oracles and a h_1 oracle. The process is shown as follows.

—*symmetric key generation query* For a fresh *sym* query, $\mathcal{B}$ randomly chooses an vectors **s** and a bit stream $K \in \{0,1\}^l$. Let $\omega = (sk_S, pk_R, \mathbf{s})$, stores (ω, K) to list L_1. Send K as an answer.(For a repeat query, returns the same answers)

—*encapsulation oracles* For a non-repeat encapsulation query (ω, τ), $\mathcal{B}$ preforms as follows:

1.$\mathcal{B}$ checks list L_1 to find $(\mathbf{s}, \omega, K)$. If there without entry accords to ω, then simulate the symmetric key generation query to generate $\mathbf{s}$ and K. At same time, store (ω, K) to list L_1

2. $\mathcal{B}$ Chooses l random vectors $\mathbf{e}_i$, which satisfied $\mathbf{e}_i \leq s\sqrt{m}$.

3. Computes $\mathbf{b}_i = \mathbf{B}_{i\tau_i}^{\top}\mathbf{s} + \mathbf{e}_i$. Parses $\mathbf{b} = (\mathbf{b}_1, \cdots, \mathbf{b}_l)$.

Then sends $(\mathbf{b}, \tau)$ as answer and stores $(\mathbf{b}, \tau, \mathbf{e}_1)$ to list L_2. (For a repeat query, returns the same answers)

Then adversary runs *decap* algorithm to check the legality of the encapsulation by sk_R. During this process he is allowed to query the h_1 oracle.

—h_1 *query* On the i-th non-repeat query $(\mathbf{s}, \tau)$, $\mathcal{B}$ checks list L_2 to find $(\mathbf{b}, \tau, \mathbf{e}_1)$, and computes $\mathbf{h}_{1i} = \mathbf{A}\mathbf{e}_i(modq)$. Sends $\mathbf{h}_{1i}$ as the answer. At last, $\mathcal{B}$ stores $(\mathbf{h}_{1i}, \mathbf{s}, \tau, \mathbf{e}_i)$ to L_3.

After all above queries finished, adversary give a forged signcryption tag-KEM $(\mathbf{b}^*, \tau^*)$ with advantage ε.

Then, challenger can solve the SIS problem as follows:

Firstly, $\mathcal{B}$ parses $\mathbf{b}^* = (\mathbf{b}_1^*, \cdots)$, and runs *decap* algorithm to get message $\mathbf{s}^*$ and $\mathbf{e}_1^*$ from $\mathbf{b}_1^*$ which satisfies $\mathbf{A}\mathbf{e}_1^* = h_{1i^*}$ and $||\mathbf{e}_1^*|| \leq s\sqrt{m}$.

Secondly, checks L_2 to find vector $\mathbf{e}_1'$ which also satisfies $\mathbf{A}\mathbf{e}_1' = h_{1i^*}$ and $||\mathbf{e}_1'|| \leq s\sqrt{m}$.

If $\mathbf{e}_1^* \neq \mathbf{e}_1'$, then $\mathbf{A}(\mathbf{e}_1^* - \mathbf{e}_1') = 0(modq)$ and $||\mathbf{e}_1^* - \mathbf{e}_1'|| \leq 2s\sqrt{m}$. As a result, $\mathcal{B}$ gets a solution of SIS problem. By literature [9], $\mathbf{e}_1^* \neq \mathbf{e}_1'$ happened with probability $1 - 2^{-\omega(logn)}$

Thus, $\mathcal{B}$ solves the SIS problem with advantage $(1 - 2^{-\omega(logn)})\varepsilon$.

Now, we have finished the proof of theorem 2.

By theorem 1 and theorem 2, the proposed signcryption tag-KEM is secure.

If we see hash function h_2 as a random oracle, according to lemma 2 and lemma 3, we conclude that the proposed hybrid signcryption which constructed by signcryption KEM + signcryption DEM is secure.

5. Conclusions

In this paper, we build a lattice-based signcryption scheme using the hybrid technique. We prove that the proposed scheme is secure in the random oracle model. As a try in lattice-based signcryption field, there still many open problems need to be studied, for example, construct a lattice-based hybrid signcryption in the standard model.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 60803149,60970119) and National Basic Research Program of China2007CB311201.

References

[1] Agrawal S, Boneh D, Boyen X. Efficient Lattice (H)IBE in the Standard Model. In Eurocrypt, H. Gilbert (Ed.) LNCS 6110, 2010, 553-572.

[2] Alwen J, Peikert C. Generating Shorter Bases for Hard Random Lattices [C]. In: STACS, volume 09001, 2009, 75-86.

[3] Boyen, X., Multipurpose Identity-Based Signcryption. In: CRYPTO '03, Boneh D. (ed.), Berlin: Springer, LNCS 2729, 2003, 383-399.

[4] Bjøstad T. E.,Dent A.W. Building better signcryption schemes with tag-KEMs. In Public Key Cryptography- PKC 2006, LNCS 3958, Springer-Verlag, 2006, 491-507.

[5] Cash D, Hofheinz D, Kiltz E, peikert C. Bonsai Trees, or How to Delegate a Lattice Basis. In Eurocrypt, H. Gilbert (Ed.), LNCS 6110, 2010, 523-552.

[6] Dent A. W., Hybrid signcryption schemes with insider security. In Information Security and Privacy - ACISP 2005, LNCS 3574, Springer, 2005, 253-266.

[7] Dent A. W. Hybrid signcryption schemes with outsider security. In Proceedings of ISC 2005, LNCS 3650, Springer, 2005, 203-217.

[8] Gordon S.D., Katz J.J, Vaikuntanathan V., A group signature scheme from lattice assumptions, To appear in ASIACRYPT 2010.

[9] Gentry C, Peikert C, Vaikuntanathan V. Trapdoors for Hard Lattices and New Cryptographic Constructions. In: STOC, Victoria, British Columbia, 2008, 197-206.

[10] John M. L., Mao W. B., Two birds one stone: Signcryption using rsa. In Topics in Cryptology - CT-RSA 2003, LNCS 2612, 2003, 211-225.

[11] Libert B., Quisquater J.J., Efficient signcryption with key privacy from gap diffie-hellman groups. In Public Key Cryptography, 2004,187-200.

[12] Libert B., and Quisquater J.J., Improved signcryption from q-diffie-hellman problems. In International Conference on Security in Communication Networks, SCN, LNCS 3352, volume 4, 2004, 220-234.

[13] Li F, Shirase M., Takagi T, Certificateless Hybrid Signcryption,The 5th Information Security Practice and Experience Conference (ISPEC 2009), LNCS 5451, Springer-Verlag, 2009, 112-123.

[14] Lyubashevsky V. Lattice-Based Identification Schemes Secure Under Active Attacks.PKC 2008, LNCS 4939, 2008, 162-179.

[15] Peikert C. Public-Key Cryptosystems from the Worst-Case Shortest Vector Problem. STOC, 2009, 333-342.

[16] Regev O. On Lattice, Learning with Errors, Random Linear Codes, and Cryptography. In: STOC, Baltimore, 2005, 84-93.

[17] Shor P W. Polynomial-time Algorithm for Prime Factorizeation and Discrete Logarithm on a Quantum Computer. SIAM Journal on Computing, 1997,26(5): 1484-1509.

[18] Zheng Y. L., Digital signcryption or how to achieve cost(signature and encryption) << cost(signature) + cost(encryption). In CRYPTO,1997, 165-179.