

# Cryptanalysis of Unidirectional Proxy Re-Encryption Scheme

Kunwar Singh<sup>1</sup>, C. Pandu Rangan<sup>2</sup>, and A.K. Banerjee<sup>3</sup>

<sup>1</sup> Computer Science and Engineering Department  
NIT Trichy, Tiruchirappalli, India  
kunwar@nitt.edu

<sup>2</sup> Computer Science and Engineering Department  
IIT, Madras  
rangan@cse.iitm.ac.in

<sup>3</sup> Mathematics Department  
NIT Trichy, Tiruchirappalli, India  
banerjee@nitt.edu

**Abstract.** At Eurocrypt 1998, Blaze, Bleumer and Strauss [7] presented a new primitive called Proxy Re-Encryption (*PRE*). This new primitive allows semi trusted proxy to transform a ciphertext for Alice (delegator) into a ciphertext for Bob (delegatee) without knowing the message. Ateniese et al [6] introduced *master secret security* as another security requirement for unidirectional *PRE*. *Master secret security* demands that no coalition of dishonest proxy and malicious delegates can compute the master secret key (private key) of the delegator. In this paper, first we have shown that Aono et al's scheme [4] is not secure under *master secret security* model. In other words if proxy and delegatee collude they can compute the private key of the delegator. Second, based on Aono et al's paper [4] we have constructed unidirectional *PRE* which is also secure under *master secret security* model. Like [4], our scheme is also multi-use.

**Keywords:** Lattice, Proxy Re-encryption, Learning With Error (LWE).

## 1 Introduction

At Eurocrypt 1998, Blaze, Bleumer and Strauss [7] presented a new primitive called Proxy Re-Encryption *PRE*. This new primitive allows semi trusted proxy to transform a ciphertext for Alice (delegator) into a ciphertext for Bob (delegatee) without knowing the message. A natural application of *PRE* is to forward encrypted e-mail to others. For example, Director (delegator) can authorize his secretary (proxy) to convert encrypted mail for Director into encrypted mail for Dean (delegatee) whenever he is on leave. Then Dean can decrypt the encrypted mail using his secret key. Blaze et al gave first *PRE* scheme which was bidirectional and multi-use. Bidirectional means proxy can transform a ciphertext for Alice to a ciphertext for Bob and vice-versa without knowing the message. In multi use, proxy can transform a ciphertext from Alice to Bob, then from Bob to Carol and so on. Ateniese et al [6] presented a first unidirectional *PRE* scheme. In unidirectional, proxy can transform a ciphertext for Alice to a ciphertext for Bob but does not allow vice-versa.

Lattice based cryptography have bloomed in recent years because of the following advantages.

- Number-theoretic hard problems like prime factorization and discrete logarithm problem can be solved in polynomial time by Shor's algorithm [11]. But till now there is no polynomial time quantum algorithm for lattice hard problems.
- Ajtai [2] in his seminal result on the average case / worst case has shown that lattice based cryptosystem in the average case is as hard as solving some lattice based hard problems in the worst case. So lattice problems give strong hardness guarantee in the average case. Lattice based cryptosystems are also efficient and parallelizable.

Recently Regev [10] defined the Learning With Error (LWE) problem and showed that it also enjoys similar average case / worst case equivalence hardness properties through a quantum reduction.

Combining these two concepts Xagawa [14] presented bidirectional lattice based proxy re-encryption scheme under LWE assumption. Singh et al [13] gave bidirectional identity based lattice based proxy re-encryption scheme. Recently Aono et al [4] presented first unidirectional lattice based proxy re-encryption scheme.

**Our Contribution:** Ateniese et al [6] introduced *master secret security* as another security requirement for unidirectional *PRE*. *Master secret security* demands that no coalition of dishonest proxy and malicious delegates can compute the master secret key (private key) of the delegator. Ateniese et al [6] gave following motivation for *master secret security*.

1. Some PRE may define two or more type of encryption schemes. In one encryption scheme ciphertext may be decrypted by only master secret (private key) of the delegator. Other encryption scheme re-encrypted ciphertext may be decrypted by private key of the delegatee.
2. Delegator may want to delegate decryption rights to delegatee but may not want to delegate signing rights to delegatee. With this security it is possible.

In this paper, first we have shown that Aono et al's scheme [4] is not secure under *master secret security* model. In other words if proxy and delegatee collude they can compute the private key of the delegator. Second, based on Aono et al's paper [4] we have constructed unidirectional *PRE* which is also secure under *master secret security* model. Like [4], our scheme is also multi-use.

## 2 Preliminaries

### 2.1 Notation

We denote  $[j] = \{0, 1, \dots, j\}$ . We assume vectors to be in column form and are written using bold letters, e.g.  $\mathbf{x}$ . Matrices are written as bold capital letters, e.g.  $\mathbf{X}$ . The norm  $\|\cdot\|$  here is the standard Euclidean norm in  $R^n$ . We denote probabilistic polynomial time as *PPT*.

## 2.2 Unidirectional Proxy Re-Encryption Scheme(PRE)

*PRE* consists of seven algorithms.

**PublicParameters( $n$ ):** On input a security parameter  $n$ , this algorithm outputs public parameters.

**KeyGeneration( $n$ ):** On input a security parameter  $n$ , this algorithm outputs a secret key  $sk$  and the corresponding public key  $pk$  of the user.

**Encrypt( $pk, M$ ):** This algorithm takes input as a public parameters, a public key and a message, and outputs ciphertext  $C$ .

**Re-Encryption Key( $sk_i, pk_i, pk_j$ ):** This algorithm takes input as a secret key  $sk_i$ , a public key  $pk_i$  and a public key  $pk_j$ , and outputs unidirectional reencryption key  $rk_{i,j}$ .

**Re-Encryption( $rk_{i,j}, C_i$ ):** On input a ciphertext  $C_i$  and re-encryption key  $rk_{i,j}$ , this algorithm outputs a re-encrypted ciphertext  $C_j$ .

**Decrypt( $sk_j, C_j$ ):** This algorithm takes input as public parameters  $PP$ , a private key  $sk_j$  and a ciphertext  $C_j$ , and outputs message  $m$ .

**Correctness.** Unidirectional Proxy Re-encryption is correct if suppose  $C_i \leftarrow \text{Encrypt}(pk_i, m)$ ,  $rk_{i,j} \leftarrow \text{Re-Encryption Key}(sk_i, pk_i, pk_j)$  and  $C_j \leftarrow \text{Re-Encryption}(rk_{i,j}, C_i)$ , following equation holds.

- $\text{Decrypt}(sk_i, C_i) = m$ .
- $\text{Decrypt}(sk_j, C_j) = m$ .

## 2.3 Security Model for Unidirectional Proxy Re-Encryption Scheme

Here security model is adapted from [6]. Security of *PRE* is defined using two properties: semantic security (IND-p-CPA) and master secret security.

### 2.3.1 Semantic Security (IND-p-CPA)

Following security model captures the idea that when a group of polynomially bounded adversarial users and proxy collude against target delegator  $B$ , they can not get any bit of information with the condition that target delegator  $B$  never gives delegation rights to any adversarial users (including delegatee). We define security model using the following game that is played between the challenger and adversary.

**Setup:** The challenger  $C$  runs  $\text{Setup}(1^k)$  and gives the public parameters  $PP$  to the adversary. Challenger  $C$  runs the KeyGeneration algorithm  $n_u$  times to obtain a list of public/private keys  $PK_{good}, SK_{good}$ , and runs the KeyGeneration algorithm for  $n_c$  times to obtain a list of corrupted private/public keys  $PK_{corr}, SK_{corr}$ . Adversary gets  $PP, SK_{corr}$ , and  $PK = (PK_{good} \cup PK_{corr})$ .

**Phase 1:** The adversary can make following queries.

- The adversary can issue re-encryption key query  $rk_{i,j}$  corresponding to the public keys  $pk_i$  and  $pk_j$  such that either  $pk_i, pk_j \in PK_{good}$  or  $pk_i, pk_j \in PK_{corr}$  or  $pk_i \in PK_{corr}$  and  $pk_j \in PK_{good}$ . Adversary can repeat this query polynomial times for different pair of public keys adaptively.
- The adversary can issue re-encryption query  $rk_{i,j}$  corresponding to public keys  $pk_i$  and  $pk_j$  such that either  $pk_i, pk_j \in PK_{good}$  or  $pk_i, pk_j \in PK_{corr}$  or  $pk_i \in PK_{corr}$  and  $pk_j \in PK_{good}$ . Challenger runs  $RKGen$  algorithm to obtain  $rk_{i,j}$  corresponding to public keys  $pk_i$  and  $pk_j$  then challenger generates ciphertext  $C_2$  by running  $Re-encryption$  algorithm.

**Challenge:** The adversary submits target public key  $pk_{i^*}$  and message  $m$  with the conditions that  $pk_{i^*}$  should belong to  $PK_{good}$ . Challenger randomly choose a bit  $r \in \{0, 1\}$  and a random string  $C$  with the size of valid ciphertext. If  $r = 0$  it sets the challenge ciphertext to  $C^* := \text{Encrypt}(PP, pk_{i^*}, m)$ . If  $r = 1$  it assigns the challenge ciphertext  $C^* := C$ . It sends challenge ciphertext  $C^*$  to the adversary.

**Phase 2:** Phase 1 procedure is repeated.

**Guess:** Adversary finally outputs a answer  $r' \in \{0, 1\}$  and wins the game if  $r = r'$ .

Adversary  $A$  is referred as an IND-p-CPA adversary. The advantage of the adversary  $A$  in attacking a PRE scheme  $\xi$  is defined as

$$Adv_{\xi, A}(n) = |Pr[r = r'] - 1/2|$$

**Definition 1.** PRE scheme is IND-p-CPA if for all PPT algorithm  $A$  and negligible function  $\epsilon$ ,  $Adv_{\xi, A}(n) \leq \epsilon$ .

### 2.3.2 Master Secret Security

Security model captures the idea that no coalition of dishonest proxy and malicious delegates can compute the master secret key (private key) of the delegator. We define security model using a game that is played between the challenger and adversary. The game proceeds as follows.

**Setup:** The challenger  $C$  runs  $\text{Setup}(1^k)$  and gives the public parameters  $PP$  to adversary.

**Challenge:** The adversary submits target delegator  $B$ .

#### Query Phase

1. The adversary can issue re-encryption key query  $rk_{i,j}$  corresponding to any public keys  $pk_i$  and  $pk_j$ .
2. The adversary can issue re-encryption query  $rk_{i,j}$  corresponding to any public keys  $pk_i$  and  $pk_j$ .

**Guess:** Adversary finally outputs a guess  $x$  for private key  $sk_B$  of target delegator  $B$  and wins if  $x = sk_B$ .

We define the adversary’s advantage in winning this game as  $AdvMSS_{\xi,A}(n) = |Pr[x = sk_B]|$

**Definition 2.** *PRE scheme is secure if for all PPT algorithm  $A$  and negligible function  $\epsilon$ ,  $Adv_{\xi,A}(n) \leq \epsilon$  and  $AdvMSS_{\xi,A}(n) \leq \epsilon$ .*

### 2.4 Integer Lattices ([8])

A lattice is the set of all integer combinations

$$L(b_1, \dots, b_n) = \left\{ \sum_{i=1}^n x_i b_i : x_i \in \mathbb{Z} \text{ for } 1 \leq i \leq n \right\}$$

of  $n$  linearly independent vectors  $\{b_1, \dots, b_n\} \in \mathbb{R}^n$ . The set of vectors  $\{b_1, \dots, b_n\}$  is called a lattice basis.

**Definition 3.** *For  $q$  prime,  $A \in \mathbb{Z}_q^{n \times m}$  and  $u \in \mathbb{Z}_q^n$ , define:*

$$\Lambda_q(A) := \{e \in \mathbb{Z}^m \text{ s.t. } \exists s \in \mathbb{Z}_q^n \text{ where } A^T s = e \pmod{q}\}$$

$$\Lambda_q^\perp(A) := \{e \in \mathbb{Z}^m \text{ s.t. } Ae = 0 \pmod{q}\}$$

$$\Lambda_q^u(A) := \{e \in \mathbb{Z}^m \text{ s.t. } Ae = u \pmod{q}\}$$

**Theorem 1.** ([2,3]) *Let  $q$  be prime and  $m := \lceil 6n \log q \rceil$ .*

*There is PPT algorithm  $TrapGen(q, n)$  that outputs a pair  $(A \in \mathbb{Z}_q^{n \times m}, T \in \mathbb{Z}^{n \times m})$  such that statistically distance between matrix  $A$  and a uniform matrix in  $\mathbb{Z}_q^{n \times m}$  is negligible and  $T$  is a basis for  $\Lambda_q^\perp(A)$  satisfying*

$$\|\tilde{T}\| \leq O(\sqrt{n \log q}) \text{ and } \|T\| \leq O(n \log q)$$

*with overwhelming probability in  $n$ .*

### 2.5 The LWE Hardness Assumption ([10,1])

Regev [10] proposed the LWE (learning with error) assumption.

**Definition 4. LWE:** *Consider a prime number  $q$ , a positive integer  $n$ , and a Gaussian distribution  $\chi^m$  over  $\mathbb{Z}_q^m$ . Given  $(A, As + x)$  where matrix  $A \in \mathbb{Z}_q^{m \times n}$  is uniformly random and  $x \in \chi^m$ .*

*LWE hard problem is to find  $s$  with non-negligible probability.*

**Definition 5. Decision LWE:** *Consider a prime number  $q$ , a positive integer  $n$ , and a Gaussian distribution  $\chi^m$  over  $\mathbb{Z}_q^m$ . The input is a pair  $(A, v)$  from an unspecified challenge oracle  $O$ , where  $A \in \mathbb{Z}_q^{m \times n}$  is chosen uniformly. An unspecified challenge oracle  $O$  is either a noisy pseudo-random sampler  $O_s$  or a truly random sampler  $O_\$$ . It is based on how  $v$  is chosen.*

1. When  $v$  is chosen to be  $As + e$  for a uniformly chosen  $s \in \mathbb{Z}_q^n$  and a vector  $e \in \mathcal{X}^m$ , an unspecified challenge oracle  $O$  is a noisy pseudo-random sampler  $O_s$ .
2. When  $v$  is chosen uniformly from  $\mathbb{Z}_q^m$ , an unspecified challenge oracle  $O$  is a truly random sampler  $O_\$$ .

Goal of the adversary is to distinguish between the above two cases with non-negligible probability.

Or we say that an algorithm  $A$  decides the  $(\mathbb{Z}_q, n, \chi)$ -LWE problem if  $|\Pr[A^{O_s} = 1] - \Pr[A^{O_\$} = 1]|$  is non-negligible for a random  $s \in \mathbb{Z}_q^n$ .

Above decision LWE is also hard even if  $s$  is chosen from the Gaussian distribution rather than the uniform distribution [5,9].

### 2.6 Small Integer Solution (SIS) Assumption ([2])

SIS and ISIS hard problems were proposed by Ajtai [2] in 1996.

**Definition 6.** Given an integer  $q$ , a matrix  $A \in \mathbb{Z}_q^{n \times m}$  and real  $\beta$ , find a short nonzero integer vector  $x \in \mathbb{Z}_q^m$  such that  $Ax = 0 \pmod q$  and  $\|x\| \leq \beta$ .

OR find a nonzero integer vector  $x \in \mathbb{Z}_2^m$  such that  $Ax = 0 \pmod q$ .

### 2.7 Inhomogeneous Small Integer Solution (ISIS) Assumption

**Definition 7.** Given an integer  $q$ , a matrix  $A \in \mathbb{Z}_q^{n \times m}$ , a syndrome  $u \in \mathbb{Z}_q^n$  and real  $\beta$ , find a short nonzero integer vector  $x \in \mathbb{Z}_q^m$  such that  $Ax = u \pmod q$  and  $\|x\| \leq \beta$ .

OR find a nonzero integer vector  $x \in \mathbb{Z}_2^m$  such that  $Ax = u \pmod q$ .

## 3 Cryptanalysis of the Aono et al’s Unidirectional Proxy Re-Encryption Scheme

### 3.1 Aono et al’s Unidirectional Proxy Re-Encryption Scheme

In Indocrypt 2013, Aono et al [4] presented key private unidirectional proxy re-encryption scheme. First, we describe Aono et al’s scheme [4]. Before that we describe functions

**Bits()** and **Power2()** used in [4].

Let  $v = (v_1, \dots, v_n) \in \mathbb{Z}_q^n$ ,  $k = \lceil \lg q \rceil$  and  $(b_{i,1}, \dots, b_{i,k})$  be the bit representation of  $v_i$  such that  $v_i = \sum_{j=0}^k 2^j b_{i,j}$ . Then **Bits()** is defined as

$$\text{Bits}(v) = [b_{1,1} \dots b_{n,1} | b_{1,2} \dots b_{n,2} | \dots | b_{1,k} \dots b_{n,k}] \in \{0, 1\}^{1 \times nk}$$

(First  $n$  bits are first bit of  $v_1, \dots, v_n$  and next  $n$  bits are second bit of  $v_1, \dots, v_n$  and so on).

Let  $X = [X_1 | \dots | X_l] \in \mathbb{Z}_q^{n \times l}$  where  $X_i$  are columns. Then

$$\text{Power2}(X) = \begin{bmatrix} X_1 \dots X_l \\ 2X_1 \dots 2X_l \\ \vdots \\ 2^{k-1}X_1 \dots 2^{k-1}X_l \end{bmatrix} \in \mathbb{Z}_k^{nk \times l}$$

It can be shown that

$$\text{Bits}(v)\text{Power2}(X) = vX \in \mathbb{Z}_q^{1 \times l}$$

**Setup** ( $n$ ): On input a security parameter  $n$ , set the parameter  $q = \text{poly}(n)$  and randomly choose matrix  $A \in \mathbb{Z}_q^{n \times n}$ .

**KeyGeneration** ( $n$ ): Let  $s = \alpha q$  for  $0 < \alpha < 1$ . Choose Gaussian noise matrices  $R, S \in \psi_s^{n \times l}$  and  $E \in \psi_s^{nk \times l}$  where  $l$  is message length. Compute  $P = R - AS$ . So private key is  $S$  and public key is  $P$ .

**Proxy Key Gen** ( $PP, S_A, P_B$ ): On input of Alice's private key  $S_A$  and Bob's public key  $P_B$ , do the following.

1. Bob chooses matrices  $X \in \psi_s^{nk \times l}$  ( $k = \lceil \lg q \rceil$ ) randomly and noise Matrix  $E \in \psi_s^{nk \times l}$  where  $\psi_s$  is a gaussian distribution. Bob computes  $-XS_B + E$  and sends  $X, -XS_B + E$  secretly to the Alice.
2. Alice compute proxy re-encryption key  $rk_{A,B} = (P_B, Q)$  where

$$Q = \begin{bmatrix} X & -XS_B + E + \text{Power2}(S_A) \\ 0_{l \times n} & I_{l \times l} \end{bmatrix}$$

Above three algorithm is enough for our cryptanalysis. Complete scheme is given in [4].

### 3.2 Attack on Aono et al's Unidirectional Proxy Re-Encryption Scheme

In Aono et al's scheme, if proxy and delegatee collude they can compute delegator's private key. It works as follows.

Let  $S = [S_1 | \dots | S_l] \in \mathbb{Z}_q^{n \times l}$  where  $S_i$  are columns. Then  $\text{Power2}(S)$  is defined as

$$\text{Power2}(S) = \begin{bmatrix} S_1 \dots S_l \\ 2S_1 \dots 2S_l \\ \vdots \\ 2^{k-1}S_1 \dots 2^{k-1}S_l \end{bmatrix} \in \mathbb{Z}_q^{nk \times l}$$

Here first  $n$  rows are  $S$ . So if we know  $\text{Power2}(S)$  then we can find  $S$ . (Here  $k$  is number of bits required to represent  $q$ ).

Now let us see the expression of proxy key  $Q$

$$Q = \begin{bmatrix} X & -XS_B + E + \text{Power2}(S_A) \\ 0_{l \times n} & I_{l \times l} \end{bmatrix},$$

where  $S_B$  is private key of Bob (delegatee). Bob (delegatee) creates  $X, E$  and securely sends  $X, -XS_B + E$  to Alice. Basically Bob knows  $X, -XS_B + E$ . Both Bob (delegatee) and proxy know  $Q, X$  and  $-XS_B + E$  and they can compute  $\text{Power2}(S_A)$ . So they can compute private key of Alice (delegator)  $S_A$  which is first  $n$  rows of the  $\text{Power2}(S_A)$ .

### 4 Lattice Based Unidirectional Proxy Re-Encryption Scheme

We describe our scheme to avoid the above attack. Our scheme is variant of Aono et al [4].

**Setup** ( $n$ ): On input a security parameter  $n$ , we set the parameters  $q = poly(n)$  and  $m = O(nlg n)$  accordingly. We choose a matrix  $A \in Z_q^{n \times n}$  and matrix  $X \in Z_q^{nk \times n}$  randomly, where  $k = \lceil lg q \rceil$ . Public parameters ( $PP$ ) are matrix  $A$  and matrix  $X$ .

**KeyGeneration** ( $n$ ): Let  $s = \alpha q$  for  $0 < \alpha < 1$ . We choose noise matrices  $R, S \in \psi_s^{n \times l}$  and  $E \in \psi_s^{nk \times l}$  where  $l$  is message length. We compute  $P_1 = R - AS$  and  $P_2 = -XS + E$ . So private key is  $S$  and public key  $P = (P_1, P_2) \in (Z_q^{n \times l}, Z_q^{nk \times l})$ .

**Encrypt** ( $PP, m, P_1, P_2$ ): To encrypt a message  $m \in \{0, 1\}^l$ , we do the following.

1. We choose noise vectors  $e_1, e_2 \in \psi_s^{1 \times n}$  and  $e_3 \in \psi_s^{1 \times l}$  where  $\psi_s$  is a gaussian distribution.
2. Compute  $c_1 = e_1A + e_2 \in Z_q^{1 \times n}$ ,  $c_2 = e_1P_1 + e_3 + m \lfloor \frac{q}{2} \rfloor$ .
3. Output the ciphertext  $C = (c_1, c_2) \in Z_q^{1 \times (n+l)}$ .

**RKGen** ( $PP, S_A, P_B$ ): On input of Alice’s private key  $S_A$  and Bob’s public key  $P_B$ , we do the following.

1. We choose noise vectors  $e_4 \in \psi_s^{nk \times nk}$  and  $e_5 \in \psi_s^{nk \times l}$  where  $\psi_s$  is a gaussian distribution.
2. We compute proxy re-encryption key  $rk_{A,B} = Q$  where

$$Q = \begin{bmatrix} e_4X & e_4P_2 + e_5 + Power2(S_A) \\ 0_{l \times n} & I_{l \times l} \end{bmatrix}$$

**Re-Encrypt** ( $PP, rk_{A,B}, C_A$ ): On input of re-encryption key  $rk_{A,B}$ , proxy transforms Alice’s ciphertext  $C_A$  to Bob’s ciphertext  $C_B$  by the following equation.

$$C_B = (c_{1B}, c_{2B}) = [Bits(c_1) | c_2] \cdot rk_{A,B} \in Z_q^{1 \times (n+l)}$$

**Decrypt** ( $PP, S_B, C_B$ ): To decrypt  $C_B = (c_1, c_2)$ , we do the following.

1. We compute

$$m = \begin{bmatrix} c_1 & c_2 \end{bmatrix} \begin{bmatrix} S_B \\ I_{l \times l} \end{bmatrix}$$

2. Let  $m = (m_1, \dots, m_l)$ . If  $m_i$  is less than  $\lfloor \frac{q}{4} \rfloor \bmod q$  than  $m_i = 0$  otherwise  $m_i = 1$ .



**Correctness:** First we decrypt the normal ciphertext

$$c_1 S_A + c_2 = e_2 S_A + e_3 + m \lfloor \frac{q}{2} \rfloor,$$

which will yield  $m$  if  $e_2 S_A + e_3$  is less than  $\lfloor \frac{q}{4} \rfloor$ . Now we decrypt the re-encrypted ciphertext

$$\begin{aligned} [\text{Bits}(c_1)|c_2].rk_{A,B}. \begin{bmatrix} S_B \\ I_{l \times l} \end{bmatrix} &= [\text{Bits}(c_1)|c_2]. \begin{bmatrix} e_4 X & e_4 P_2 + e_5 + \text{Power2}(S_A) \\ 0_{l \times n} & I_{l \times l} \end{bmatrix} \begin{bmatrix} S_B \\ I_{l \times l} \end{bmatrix} \\ &= [\text{Bits}(c_1)|c_2]. \begin{bmatrix} e_4 E + e_5 + \text{Power2}(S_A) \\ I_{l \times l} \end{bmatrix} \\ &= \text{Bits}(c_1)e_4 E + \text{Bits}(c_1)e_5 + \text{Bits}(c_1)\text{Power2}(S_A) + c_2 \\ &= \text{Bits}(c_1)e_4 E + \text{Bits}(c_1)e_5 + c_1 S_A + c_2 \\ &= \text{Bits}(c_1)e_4 E + \text{Bits}(c_1)e_5 + e_2 S_A + e_3 + m \lfloor \frac{q}{2} \rfloor \end{aligned}$$

which will yield  $m$  if  $\text{Bits}(c_1)e_4 E + \text{Bits}(c_1)e_5 + e_2 S_A + e_3$  is less than  $\lfloor \frac{q}{4} \rfloor$ .

Since  $e_2, e_3, e_4, e_5, S_A$  are from Gaussian distribution  $\psi_s$  so with some  $s = \alpha q$  it is possible that  $\text{Bits}(c_1)e_4 E + \text{Bits}(c_1)e_5 + e_2 S_A + e_3$  is less than  $\lfloor \frac{q}{4} \rfloor$ .

**Theorem 2.** *Lattice based unidirectional PRE scheme is IND-p-CPA (semantic) secure assuming the  $\text{LWE}_{q,\chi}$  is hard or  $\text{Adv}_{B,\text{LWE}_{q,\chi}}(n) = \text{Adv}_{\chi,A}(n)$ .*

**Proof:** Here proof has similar structure as in the proof of [4,14,12]. Now we show semantic security (IND-p-CPA) of *PRE*. Suppose there is a PPT adversary  $\mathcal{A}$  with non-negligible probability breaks *PRE* scheme. Then we construct PPT algorithm  $\mathcal{B}$  (challenger) that solves LWE hard problem with non-negligible probability. Here  $CU$  denotes set of corrupted users and  $HU$  denotes set of honest users.

Challenger  $\mathcal{B}$  obtains the  $n + l$  LWE samples from LWE oracle, which is parsed as  $(A, c_1 = e_1 A + e_2)$  and  $(P_1, c_2 = e_1 P_1 + e_3)$ . Now challenger  $\mathcal{B}$  sets the master public key  $mpk = A$  and public key of target delegator  $PK^* = P_1$ .

**Re-Encryption Queries:** Challenger  $\mathcal{B}$  answers re-encryption key queries and re-encryption queries of the adversary  $\mathcal{A}$  in following way.

- Whenever  $\mathcal{A}$  submits a re-encryption key query for the the identities  $u_j$  and  $u_k$  such that  $u_j, u_k \in HU$ , challenger  $\mathcal{B}$  randomly choose matrices  $X_1, X_2 \in \mathbb{Z}_q^{n_k \times l}$  and returns

$$Q = \begin{bmatrix} X_1 & X_2 \\ 0_{l \times n} & I_{l \times l} \end{bmatrix}$$

to the challenger  $\mathcal{B}$ .

- Whenever  $\mathcal{A}$  submits a re-encryption query for the the public keys  $u_j$  and  $u_k$  such that  $u_j, u_k \in HU$ , Challenger  $\mathcal{B}$  returns a random vector in  $\mathbb{Z}_q^{1 \times (n+l)}$ .
- Whenever  $\mathcal{A}$  submits a re-encryption key query or a re-encryption query for the the public keys  $u_j$  and  $u_k$  such that  $u_j, u_k \in CU$ . Since private key is known to corrupted users so adversary himself can compute re-encryption key or re-encrypted ciphertext. (This query may not be required)

**Challenge Ciphertext:** Now adversary  $\mathcal{A}$  submits a message  $m$ . Now challenger  $\mathcal{B}$  computes  $c_1^* = c_1$  and  $c_2^* = c_2 + m \lfloor \frac{q}{2} \rfloor$  and sends  $C^* = (c_1^*, c_2^*)$  to adversary  $\mathcal{A}$ .

**Phase 2:** Adversary can ask query with some restriction same as in phase one.

Now adversary  $\mathcal{A}$  outputs that challenged ciphertext is a valid ciphertext, then challenger will output that oracle  $O$  as pseudo-random LWE oracle. If adversary  $\mathcal{A}$  outputs random ciphertext then adversary will output random LWE oracle. In other words if adversary  $\mathcal{A}$  terminates with some output then challenger  $\mathcal{B}$  outputs the same. So if adversary  $\mathcal{A}$  breaks the scheme then one can construct challenger  $\mathcal{B}$  which solves LWE.

$Adv_{B,LWE_{q,\chi}}(n) = Adv_{\mathcal{A},A}(n)$ . Hence our scheme is semantically secure.

**Theorem 3.** *Lattice based unidirectional PRE scheme is master secret security assuming the  $LWE_{q,\chi}$  is hard or  $Adv_{B,LWE_{q,\chi}}(n) = Adv_{MSS_{\mathcal{A},A}}(n)$ .*

**Proof:** Here proof has similar structure as in the proof of [4,14]. We now show master secret security of *PRE*. Suppose there is a PPT adversary  $\mathcal{A}$  that can compute private key of the delegator  $D$  in our *PRE* scheme with non-negligible probability then we construct a PPT algorithm (challenger  $\mathcal{B}$ ) that solves LWE hard problem with non-negligible probability. Here  $CU$  denotes set of corrupted users and  $HU$  denotes set of honest users.

For  $i = 1$  to  $i = l$ ,

- Challenger  $\mathcal{B}$  obtains the  $nk + n$  LWE samples from LWE oracle, which is parsed as  $(-A, P_{1,i} = -AS_i + R_i)$  and  $(-X, P_{2,i} = -XS_i + E_i)$ .

Now challenger  $\mathcal{B}$  sets the master public key  $mpk = A$  and public key of the target delegator  $P = (P_1, P_2)$ , where

$$P_1 = (P_{1,1}, \dots, P_{1,l}) \text{ and } P_2 = (P_{2,1}, \dots, P_{2,l}).$$

$\mathcal{B}$  does not know about private key  $S = (S_1, \dots, S_l)$ .

**Re-encryption Queries:** Challenger  $\mathcal{B}$  answers re-encryption key queries and re-encryption queries of the adversary  $\mathcal{A}$  in following way.

- Whenever  $\mathcal{A}$  submits a re-encryption key query for the the public keys  $u_j$  and  $u_x$  such that  $u_j \in HU, u_k \in CU$ , challenger  $\mathcal{B}$  randomly choose matrices  $X_1, X_2 \in \mathbb{Z}_q^{nk \times l}$  and returns

$$Q = \begin{bmatrix} X_1 & X_2 \\ 0_{l \times n} & I_{l \times l} \end{bmatrix}$$

to the challenger  $\mathcal{B}$ . Here  $X_1$  is random because  $X_1 = e_4 X$ , where  $e_4$  is random and  $X$  is public key of corrupt user. But in [4]  $X_1 = X$ , so  $\mathcal{B}$  can not return random  $X_1$  as one part of  $Q$  in [4].

- Whenever  $\mathcal{A}$  submits a re-encryption query for the the public keys  $u_j$  and  $u_x$  such that  $u_j \in HU, u_k \in CU$ , challenger  $\mathcal{B}$  returns a random vector in  $\mathbb{Z}_q^{1 \times (n+l)}$ .

Now adversary  $\mathcal{A}$  outputs private key of the target delegator, challenger  $\mathcal{B}$  outputs the same as the solution for *LWE* problem. So if adversary  $\mathcal{A}$  can compute private key of the delegator  $D$  in our *PRE* scheme then one can construct challenger  $\mathcal{B}$  which solves *LWE*.

$Adv_{\mathcal{B},LWE_{q,\chi}}(n) = Adv_{\chi,A}(n)$ . Hence our scheme is secure under *master secret security*.

## 5 Conclusion

We have shown that Aono et al's [4] scheme is not secure under *master secret security*. We have also shown that our scheme is not only semantically secure but also secure under *master secret security* model. Lattice based *PRE* in identity based setting is an open problem.

## References

1. Agrawal, S., Boneh, D., Boyen, X.: Efficient Lattice (H)IBE in the Standard Model. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 553–572. Springer, Heidelberg (2010)
2. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: STOC, pp. 99–108. ACM (1996)
3. Alwen, J., Peikert, C.: Generating Shorter Bases for Hard Random Lattices. In: International Symposium on Theoretical Aspects of Computer Science, STACS 2009, pp. 75–86. IBFI Schloss Dagstuhl (2009)
4. Aono, Y., Boyen, X., Phong, T.L., Wang, L.: Key-private proxy re-encryption under *LWE*. In: Paul, G., Vaudenay, S. (eds.) INDOCRYPT 2013. LNCS, vol. 8250, pp. 1–18. Springer, Heidelberg (2013)
5. Applebaum, B., Cash, D., Peikert, C., Sahai, A.: Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 595–618. Springer, Heidelberg (2009)
6. Ateniese, G., Fu, K., Green, M., Hohenberger, S.: Improved Proxy Re-encryption Schemes with Applications to Secure Distributed Storage. In: 12th Annual Network and Distributed System Security Symposium. LNCS, pp. 29–35. Springer (2005)
7. Blaze, M., Bleumer, G., Strauss, M.: Divertible protocols and atomic proxy cryptography. In: Nyberg, K. (ed.) EUROCRYPT 1998. LNCS, vol. 1403, pp. 127–144. Springer, Heidelberg (1998)
8. Micciancio, D., Goldwasser, S.: Complexity of Lattice Problems: A Cryptographic Perspective, vol. 671. Kluwer Academic Publishers (2002)
9. Lindner, R., Peikert, C.: Better key sizes (and attacks) for *LWE*-based encryption. In: Kiayias, A. (ed.) CT-RSA 2011. LNCS, vol. 6558, pp. 319–339. Springer, Heidelberg (2011)
10. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: STOC, pp. 84–93. ACM (2005)
11. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal on Computing, 1484–1509 (1997)
12. Singh, K., Pandu Rangan, C., Banerjee, A.K.: Lattice based efficient threshold public key encryption scheme. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA) 4(4), 93–107 (2013)

13. Singh, K., Pandu Rangan, C., Banerjee, A.K.: Lattice based identity based proxy re-encryption scheme. *Journal of Internet Services and Information Security (JISIS)* 3(3/4), 38–51 (2013)
14. Xagawa, K.: *Cryptography with Lattices*. PhD Thesis. Department of Mathematical and Computing Sciences Tokyo Institute of Technology (2010)