Cryptanalysis of An Improved Predicate Encryption Scheme from LWE

Chengbo Xu
School of Mathematical Sciences, University of Jinan
No. 336, Nanxinzhuang West Road, Jinan 250022, Shandong, P. R. China
(Email: cbqysy@163.com)

(Received Apr. 1, 2018; Revised and Accepted Aug. 21, 2018; First Online June 5, 2019)

Abstract

Predicate encryption scheme is a paradigm which provides fine-grained access control and has attractive applications. In 2017, Brakerski, Tsabary, Vaikuntanathan, and Wee (TCC 2017) proposed a new LWE based predicate encryption scheme in order to overcome drawbacks in the scheme proposed by Gorbunov, Vaikuntanathan and Wee (CRYPTO 2015). In this paper, We analyze this scheme and provide two practical attacks to show that the scheme (TCC 2017) is insecure under the full attribute hiding security model. These two attacks mainly exploit several homomorphic and linear properties in the construction. This illustrates that in order to construct full attribute hiding secure predicate encryption scheme these weak properties must be bypassed.

Keywords: Functional Encryption; Lattice with Error (LWE); Predicate Encryption

1 Introduction

With the emergence and development of cloud computing and other complex networks, considerable progress has been witnessed recently in the field of computing on encrypted data. A number of concepts and constructions of cryptographic primitives have turned out, such as Attribute Based Encryption [3,7,8,13,15,19,21,24,25], Fully Homomorphic Encryption [12,14,17,18], Functional Encryption [1,2,4,9,16,23].

Among them, the notion of fully homomorphic encryption permits arbitrary computation on encrypted data, but still restricts decryption to be *all or nothing* as traditional notions of public key encryption. However, *Functional encryption* [9], attribute based encryption [8,19], provides a satisfying solutions to this problem in theory. Two features provided by functional encryption are fine-grained access and computing on encrypted data. The fine-grained access part is formalized as a cryptographic notion, named *predicate encryption* [10, 11, 20, 22]. In predicate encryption system, ciphertext ct is associated with descriptive attribute values a in addition to plaintexts μ while secret key sk_f is associated with a predicate f. A user holding the key sk_f can decrypt ciphertext ct if and only if $f(a) = 0$.

In the literature, The security requirement for predicate encryption scheme can be formalized in two ways. The basic one is the definition of weak attribute-hiding, which enforces privacy of a and the plaintext amidst multiple unauthorized secret key queries: an adversary holding secret keys for different query predicates learns nothing about the attribute x and the plaintext if none of them is individually authorized to decrypt the ciphertext. The second, called full attribute-hiding, requires that a remains hidden given an unbounded number of keys, which may comprise of both authorized and unauthorized keys.

Recently, Gorbunov, Vaikuntanathan and Wee [20] constructed a predicate encryption scheme for all circuits (of an a-priori bounded polynomial depth) from the LWE assumption. But the construction only achieved the weak attribute-hiding security. Two sources of leakage in the scheme prevent its construction from achieving the full attribute-hiding property. Later, Agrawal [2] indeed exploited the two sources of leakage to recover the attribute a under full attribute-hiding attacks. Based on these, Brakerski *etc.* [11] proposed an improved predicate encryption scheme by feat of the new "Dual Use" technique, that is, using the same LWE secret for the FHE [20] and the ABE [8]. In this paper, we cryptanalyze this improved scheme and show that it still does not achieve the full attribute-hiding security.

Our Contributions: We provide two polynomial time attacks to show that the Brakerski *etc.*'s predicate encryption scheme [11] is still not secure under the full attribute-hiding attacks.

1) Our first attack is inspired by the attack method in [2] which is designed to attack the inner product predicate encryption scheme [4] mainly using the inherent property of linearity in the inner product operation. However, the Brakerski *etc.*'s predicate encryption scheme we considered here, is designed for general predicates described by polynomial-size circuits, instead of only inner product predicate. Conse-

quently, two barriers prevent applying the attack into Brakerski *etc.*'s scheme directly. Fortunately, we find and prove two homomorphic properties which conquer above two barriers and make the attack practical.

2) Our second attack is based on the following three observations: The first one is that when running the ciphertexts homomorphic evolution algorithm in [8], the error growth is linear in the corresponding original errors. The second is that when running the GSW homomorphic evaluation algorithm, the error growth is also linear in the corresponding original errors. More importantly, the coefficients in these two linear combination are both public in view of the adversary. The last observation is that by construction of the scheme in [11], the adversary is able to obtain a set of linear equations over all the original errors given a 1-key. Thus, by requesting sufficient 1-keys, the attacker will solve this linear system to recover the errors used in encryption, which lead to recovery of the predicate a.

2 Preliminaries

Notation. Let λ be the security parameter, and let PPT denote probabilistic polynomial time. We use bold uppercase letters to denote matrices $\mathbf{M}$, and bold lowercase letters to denote vectors v. We write $[n]$ to denote the set $\{1, ..., n\}$, and $|t|$ to denote the number of bits in the string t. We denote the i-th bit s by $s[i]$. We say a function $negl(\cdot) : N \to (0, 1)$ is negligible, if for every constant $c \in N$, $negl(n) < n^{-c}$ for sufficiently large n.

2.1 Predicate Encryption

We recall the syntax and security definition of *predicate encryption* (PE) [4,22]. PE can be regarded as a generalization of attribute based encryption. A PE scheme PE with respect to an attribute universe A, predicate universe C and a message universe M consists of four algorithms $\Pi = (Setup, Keygen, Enc, Dec)$:

$Setup(1^\lambda, A, C, M)$: On input the security parameter λ, the setup algorithm outputs public parameters mpk and master secret key msk.

$keygen(msk, C)$: On input the master secret key msk and a predicate $C \in C$, the key generation algorithm outputs a secret key sk_C.

$Enc(mpk, a, \mu)$: On input the public parameter mpk and an attribute/message pair (a, μ), it outputs a ciphertext ct.

$Dec((sk_C, C), ct)$: On input the secret key sk_C and a ciphertext ct, it outputs the corresponding plaintext μ if $C(a) = 1$; otherwise, it outputs $\perp$.

Definition 1 (Correctness)**.** *We say the PE scheme described above is correct, if for any* $(msk, mpk) \leftarrow Setup(1^\lambda)$*, any message* μ*, any predicate* $C \in C$*, and attribute* $a \in A$ *such that* $C(a) = 0$*, we have* $Dec(sk_C, ct) = \mu$*, where* $sk_C \leftarrow Keygen(msk, C)$ *and* $ct \leftarrow Enc(mpk, a, \mu)$*.*

Security. The model $Expt_A^{PE}(1^\lambda)$ for defining the fully attribute-hiding security of PE against adversary A (under chosen plaintext attacks) is given as follows:

1) $Setup$ is run to generate keys mpk and msk, and mpk is given to A.
2) A may adaptively make a polynomial number of key queries for predicate functions, f. In response, A is given the corresponding key $sk_f \xleftarrow{R} Keygen(msk, f)$.
3) A outputs challenge attribute vector $(a^{(0)}, a^{(1)})$ and challenge plaintexts $(\mu^{(0)}, \mu^{(1)})$, subject to the following restrictions:
   - $f(a^{(0)}) \neq 0$ and $f(a^{(1)}) \neq 0$ for all the key queried predicate, f.
   - Two challenge plaintexts are equal, i.e., $\mu^{(0)} = \mu^{(1)}$, and any key query f satisfies $f(a^{(0)}) = f(a^{(1)})$, i.e., one of the following conditions.
     - ⋆ $f(a^{(0)}) = 0$ and $f(a^{(1)}) = 0$;
     - ⋆ $f(a^{(0)}) \neq 0$ and $f(a^{(1)}) \neq 0$,
4) A random bit b is chosen. A is given $ct_{a^{(b)}} \xleftarrow{R} Enc(mpk, \mu^{(b)}, a^{(b)})$.
5) The adversary may continue to issue a polynomial number of key queries for additional predicate, f, subject to the restriction given in Step 3. A is given the corresponding key $sk_f \xleftarrow{R} Keygen(mpk, msk, f)$.
6) A outputs a bit b', and wins if $b' = b$.

The advantage of adversary A in attacking a PE scheme PE is defined as:

$$Advantage_A(1^\lambda) = \left| \Pr[b^* = b'] - \frac{1}{2} \right|,$$

where the probability is over the randomness of the challenger and adversary.

Definition 2 (Fully attribute-hiding)**.** *We say an PE scheme* PE *is fully attribute-hiding against chosen-plaintext attacks in adaptive attribute setting, if for all PPT adversaries* A *engaging in experiment* $Expt_A^{PE}(1^\lambda)$*, we have*

$$Advantage_A(1^\lambda) \leq negl(\lambda).$$

2.2 Gadget Matrix

We now recall the gadget matrix [5,23], and the extended gadget matrix technique appeared in [6], that are important to our construction.

Definition 3. *Let $m = n \cdot \lceil \log q \rceil$, and define the gadget matrix*

$$G_{n,2,m} = g \otimes I_n \in Z_q^{n \times m}$$

where vector $g = (1, 2, 4, ..., 2^{\lfloor \log q \rfloor}) \in Z_q^{\lceil \log q \rceil}$, and $\otimes$ denotes tenser product. We will also refer to this gadget matrix as "powers-of-two" matrix. We define the inverse function $G_{n,2,m}^{-1} : Z_q^{n \times m} \to \{0,1\}^{m \times m}$ which expands each entry $a \in Z_q$ of the input matrix into a column of size $\lceil \log q \rceil$ consisting of the bits of binary representations. We have the property that for any matrix $A \in Z_q^{n \times m}$, it holds that $G_{n,2,m} \cdot G_{n,2,m}^{-1}(A) = A$.

2.3 GSW Homomorphic Encryption Scheme

The GSW scheme [5,18] is parameterized by a dimension n, a modulus q with $l = \lceil \log_2 q \rceil$, and some error distribution χ over Z which we assume to be subGaussian. Formally, we describe the scheme as follows:

- GSW.Gen (choose $\bar{s} \leftarrow \chi^{n-1}$ and output secret key $s = (\bar{s}, 1) \in Z^n$).

- GSW.Enc $(s, \mu \in Z)$: choose $\overline{C} \leftarrow Z_q^{(n-1) \times nl}$ and $e \leftarrow \chi^m$, let $b^T = e^t - \bar{s}^T \overline{C} (\text{mod} q)$, and output the cphertext

$$C = \begin{bmatrix} \overline{C} \\ b^T \end{bmatrix} + \mu G$$

  where G is the gadget matrix. Notice that $s^T C = e^T + \mu \cdot s^T G (\text{mod} q)$.

- GSW.Dec(s, C): Let c be the penultimate column of C, and output $\mu = \lfloor \langle s, c \rangle \rceil_2$.

- GSW.Eval(C_1, C_2):

  - Homomorphic addition: $C_1 \boxplus C_2 = C_1 + C_2$.
  - Homomorphic multiplication: $C_1 \boxdot C_2 \leftarrow C_1 \cdot G^{-1}(C_2)$, and is right associative.

2.4 Lattice Evolution

The following lemma is an abstraction of the evaluation procedure that developed in a long sequence of works [3, 5, 8, 11, 18, 20]. Here we use the formalism as in [11].

Lemma 1. *There exist efficient deterministic algorithms EvalF and EvalFX such that for all $n, q, l \in N$, and for any sequence of matrices $(B_1, \cdots, B_l) \in (Z_q^{n \times n \lceil \log q \rceil})^l$, for any depth-d Boolean circuit $f : \{0,1\}^l \to \{0,1\}$ and for every $x = (x_1, \cdots, x_l) \in \{0,1\}^l$, the following properties hold.*

- *The outputs $H_f = EvalF(f, B_1, \cdots, B_l)$ and $H_{f,x} = EvalFX(f, x, B_1, \cdots, B_l)$ are both matrices in $Z^{(ln \lceil \log q \rceil)} \times n \lceil \log q \rceil$;*

- *It holds that $\|H_f\|_\infty$, $\|H_{f,x}\|_\infty \leqslant (n \log q)^{O(d)}$;*

- *It holds that $[B_1 - x_1 G \| \cdots \| B_l - x_l G] \cdot H_{f,x} = [B_1 \| \cdots \| B_l] \cdot H_f - f(x) G (\text{mod} q)$.*

Construction of algorithms EvalF and EvalFX:

☐ For an addition gate$f(x_1, \cdots, x_k) = x_1 + \cdots + x_k$,

$$EvalF(f, B_1, \cdots, B_k) = \begin{bmatrix} E & \cdots & E \end{bmatrix}^T$$
$$EvalFX(f, x, B_1, \cdots, B_k) = \begin{bmatrix} E & \cdots & E \end{bmatrix}^T$$

where E is the identity matrix.

☐ For a multiplication gate$f(x_1, \cdots, x_k) = x_1 x_2 \cdots x_k$,

$$EvalF(f, B_1, \cdots, B_k) = \begin{bmatrix} O \\ \vdots \\ O \\ G^{-1}(-B_{k-1} G^{-1}(\cdots G^{-1}(-B_2 G^{-1}(-B_1)))) \end{bmatrix}$$

$$EvalFX(f, x, B_1, \cdots, B_k) = \begin{bmatrix} x_2 x_3 \cdots x_k E \\ x_3 x_4 \cdots x_k G^{-1}(-B_1) \\ \vdots \\ G^{-1}(-B_{k-1} G^{-1}(\cdots G^{-1}(-B_2 G^{-1}(-B_1)))) \end{bmatrix}$$

where E is the identity matrix.

☐ For a general circuit f which has l input wires, we construct the required matrices inductively input to output gate-by-gate.

3 Review of the BTVW Predicate Encryption Scheme Using Dual-Use Technique

In this section, we provide a brief overview of the BTVW predicate encryption scheme using Dual-Use technique [11].

We write $\overline{G} \in Z_q^{n \times (n+1) \log q}$ to denote all but the last row of G which is the gadget matrices in $Z_q^{(n+1) \times (n+1) \log q}$. Given a circuit computing a function $f : \{0,1\}^l \to \{0,1\}$, and GSW FHE encryptions $\Psi := (\Psi_1, \cdots, \Psi_l)$ of $x_1, \cdots, x_l$,we write Ψ_f to denote fhe.eval(f, Ψ). Recalling syntax of GSW, Ψ_f is a matrix, and we denote the last row of Ψ_f as $\underline{\Psi}_f$, all but the last row of Ψ_f as $\overline{\Psi}_f$. In addition, we denote the circuit that computes $\Psi \mapsto \overline{\Psi}_f$ as $\hat{f}$, namely it takes as input the bits of Ψ and outputs the matrix $\overline{\Psi}_f$.

We let $e \xleftarrow{\sigma} Z^m$ denote the process of sampling a vector e where each of its entries is drawn independently from the discrete Gaussian with mean 0 and standard deviation σ over Z.

- $Setup(1^\lambda, 1^l, 1^d)$: sample (B, T_B) where $B \in Z_q^{n\times(n+1)\log q}$ and T_B denotes the trapdoor for B. Pick $B_j \xleftarrow{\$} Z_q^{n\times(n+1)\log q}$ and $p \xleftarrow{\$} Z_q^n$. Output

$$mpk := (B, \{B_j\}_{j\in[L]}, p), \qquad msk := (T_B)$$

where $L = l(n+1)^2\log^2 q$.

- $Enc(mpk, x, M \in \{0,1\})$: pick $s \xleftarrow{\$} Z_q^n, e, e_0, e_j \xleftarrow{\sigma} Z^m, e' \xleftarrow{\$} Z, R_i \in \{0,1\}^{(n+1)\log q\times(n+1)\log q}$ and compute

$$\Psi_i := \begin{pmatrix} B \\ s^TB + e^T \end{pmatrix} R_i + x_iG.$$

Parse $\Psi := [\Psi_1|\cdots|\Psi_l]$ as its binary representation $\psi_1, \cdots, \psi_L$. Compute

$$c_{in}^T := s^TB + e_0^T, \qquad c_j^T := s^T[B_j - \psi_j\overline{G}] + e_j^T$$

and $c_{out} := s^Tp + e' + M\cdot\lfloor q/2\rfloor(\mod q)$. Set the PE ciphertext as follows:

$$ct := (\Psi, c_0, \{c_j\}_{j\in[L]}, c_{out}).$$

- $KeyGen(msk, f)$: Let $\hat{f}$ denote the circuit computing $\Psi \mapsto \overline{\Psi}_f$ and

$$H_{\hat{f}} := EvalF(\hat{f}, \{B_j\}_{j\in[L]}), B_{\hat{f}} := [B_1|\cdots|B_L]\cdots H_{\hat{f}}$$

Sample a short sk_f using T_B such that

$$[B|B_{\hat{f}}]\cdot sk_f = p.$$

Output sk_f.

- $Dec((sk_f, f), ct)$: Let $\hat{f}$ denote the circuit computing $\Psi \mapsto \overline{\Psi}_f$ and compute:

$$\Psi_f := \hat{f}(\Psi),$$

$$H_{\hat{f},\Psi} := EvalFX(\hat{f}, \Psi, \{B_j\}_{j\in[L]}),$$

$$c_{\hat{f}}^T := [c_1^T|\cdots|c_L^T]\cdot H_{\hat{f},\Psi} + \underline{\Psi}_f.$$

Output the MSB of $c_{out} - [c_{in}^T|c_{\hat{f}}^T]\cdot sk_f$.

4 Attack #I

In this section,we provide an attack to demonstrate that the predicate encryption scheme reviewed above is insecure against an adversary that requests 1-keys.

Case 1. Say the attacker requests keys for functions f_1 and f_2 such that for the challenge x it holds that:

$$f_1(x) = 0, \qquad f_2(x) = 0.$$

Then, by functionality, the attacker must learn two linear equations in the challenge x but must not learn anything more. Now, by the construction in [11], we can compute matrices B_{f_1} and B_{f_2} from the master public parameter mpk as follows:

$$B_{f_1} = EvalF(B_1, \cdots, B_L, \hat{f}_1),$$

$$B_{f_2} = EvalF(B_1, \cdots, B_L, \hat{f}_2),$$

where $\hat{f}_1$ and $\hat{f}_2$ denote circuits that compute $\Psi \mapsto \overline{\Psi}_{f_1}$ and $\Psi \mapsto \overline{\Psi}_{f_2}$ repectively. Then, we have the following equations:

$$\left[B|B_{f_1}\right]\begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = p(\bmod q),$$

$$\left[B|B_{f_2}\right]\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = p(\bmod q).$$

Hence,

$$\left[B|B_{f_1}|B_{f_2}\right]\begin{bmatrix} r_1 - u_1 \\ r_2 \\ -u_2 \end{bmatrix} = 0(\bmod q).$$

Thus we find a short vector in the lattice $\left[B|B_{f_1}|B_{f_2}\right]$.

Case 2. To obtain more short vectors in the lattice $\left[B|B_{f_1}|B_{f_2}\right]$, the attacker requests a key for small elements k_1f_1 and k_2f_2 for some $k_1, k_2 \in Z_p$. By the construction of GSW [5] and ABE [8], we have the following equations which we will prove a little bit later.

Lemma 2. $B_{k_1f_1} = k_1B_{f_1}, B_{k_2f_2} = k_2B_{f_2}$.

With this lemma, the attacker can get:

$$\left[B|B_{k_1f_1}\right]\begin{bmatrix} r_1' \\ r_2' \end{bmatrix} = p(\bmod q)$$

$$\left[B|B_{k_2f_2}\right]\begin{bmatrix} u_1' \\ u_2' \end{bmatrix} = p(\bmod q)$$

$$\left[B|B_{f_1}\right]\begin{bmatrix} r_1' \\ k_1r_2' \end{bmatrix} = p(\bmod q)$$

$$\left[B|B_{f_2}\right]\begin{bmatrix} u_1' \\ k_2u_2' \end{bmatrix} = p(\bmod q).$$

Hence,

$$\left[B|B_{f_1}|B_{f_2}\right]\begin{bmatrix} r_1' - u_1' \\ k_1r_2' \\ -k_2u_2' \end{bmatrix} = 0(\bmod q).$$

It is easily to see that this results in a new short vector in the same lattice that is independent of result in the first case.

Case 3. More generally, by querying multiple functions $g_i = a_if_1 + b_if_2$ for $i \in [Q]$ where $a_i, b_i \in Z_p$ are small and Q is some polynomial, the attack obtains 1-keys $[v_{1i}, v_{2i}]$ which gives the following equation:

$$\begin{bmatrix}B|B_{g_i}\end{bmatrix}\begin{bmatrix}v_{1i}\\v_{2i}\end{bmatrix} = p(\bmod q).$$

By the construction of GSW [5] and ABE [8], we have the following equations which we will prove a little bit later.

Lemma 3. $B_{g_i} = a_i B_{f_1} + b_i B_{f_2}$ *for all* $i \in [Q]$

With this lemma, we have:

$$\begin{aligned}
&\begin{bmatrix}B|B_{g_i}\end{bmatrix}\begin{bmatrix}v_{1i}\\v_{2i}\end{bmatrix}\\
&= Bv_{1i} + B_{g_i}v_{2i}\\
&= Bv_{1i} + (a_iB_{f_1} + b_iB_{f_2})v_{2i}\\
&= Bv_{1i} + B_{f_1}(a_iv_{2i}) + B_{f_2}(b_iv_{2i})\\
&= \begin{bmatrix}B|B_{f_1}|B_{f_2}\end{bmatrix}\begin{bmatrix}v_{1i}\\a_iv_{2i}\\b_iv_{2i}\end{bmatrix}\\
&= p(\bmod q).
\end{aligned}$$

Therefore, for some $i, j \in [Q]$ we have

$$\begin{bmatrix}B|B_{f_1}|B_{f_2}\end{bmatrix}\begin{bmatrix}v_{1i}\\a_iv_{2i}\\b_iv_{2i}\end{bmatrix} = p(\bmod q)$$

$$\begin{bmatrix}B|B_{f_1}|B_{f_2}\end{bmatrix}\begin{bmatrix}v_{1j}\\a_jv_{2j}\\b_jv_{2j}\end{bmatrix} = p(\bmod q).$$

Hence,

$$\begin{bmatrix}B|B_{f_1}|B_{f_2}\end{bmatrix}\begin{bmatrix}v_{1i}-v_{1j}\\a_iv_{2i}-a_jv_{2j}\\b_iv_{2i}-b_jv_{2j}\end{bmatrix} = 0(\bmod q).$$

Thus, an attacker may get a short basis for the lattice $\begin{bmatrix}B|B_{f_1}|B_{f_2}\end{bmatrix}$. Since $f_1(x) = 0, f_2(x) = 0$, by computing the legitimate decryption equations he/she obtains:

$$\begin{aligned}
&[B^Ts + \eta|B_{f_1}^Ts + \eta_{f_1}|B_{f_2}^Ts + \eta_{f_2}]\\
&= [B|B_{f_1}|B_{f_2}]^T + noise.
\end{aligned}$$

Now, the attacker may use the basis to recover the secret vector s, and hence break the security of the LWE samples that encode the attributes x.

Proof of Lemma 2: By the construction in [11], the computing process of B_f is located in the phase of $KeyGen$. Given a circuit computing a function $f : \{0,1\}^l \to \{0,1\}$, we need to conduct the following two steps in order to get B_f:

- Run the GSW Evaluation algorithm $GSW.Eval(f,\cdot)$ and then make a little change in the output phase to get the circuit corresponding to function $\hat{f} : \Psi \to \overline{\Psi}_f$.
- With the public parameters $B_1, \cdots, B_L$, run the matrices evolution algorithm $EvalF$ to compute $B_f = EvalF(B_1, \cdots, B_L, \hat{f})$.

Therefore, in order to prove the homomorphic relationship in Lemma 4.1, we only need to prove the following two homomorphic properties:

Claim 4.1. $(\hat{kf}) = k(\hat{f})$

Claim 4.2. $B_{kf} = kB_f$

Proof of Claim 4.1: Note that function $\hat{f}$ is computed from f through running the GSW evaluation algorithm $GSW.Eval(f,\cdot)$. Hence, to prove the relationship in Claim 4.1 means to prove that the GSW evaluation algorithm $GSW.Eval(f,\cdot)$ has the following homomorphic property:

$$GSW.Eval((kf),\cdot) = k \times GSW.Eval(f,\cdot).$$

Case 1. when the circuit computing f is only an addition gate, i.e. $f = x_1 + x_2$, for any GSW ciphertexts $C_1 = \begin{bmatrix}B_1\\s^TB_1 + e_1^T\end{bmatrix} + \mu_1G, C_2 = \begin{bmatrix}B_2\\s^TB_2 + e_2^T\end{bmatrix} + \mu_2G$, we have

$$\begin{aligned}
&GSW.Eval(kf, C_1, C_2)\\
&= kC_1 + kC_2\\
&= \begin{bmatrix}kB_1 + kB_2\\(ks^TB_1 + ks^TB_2) + (ke_1^T + ke_2^T)\end{bmatrix} + (k\mu_1G + k\mu_2G)\\
&= k(\begin{bmatrix}B_1 + B_2\\(s^TB_1 + s^TB_2) + (e_1^T + e_2^T)\end{bmatrix}) + k(\mu_1G + \mu_2G)\\
&= k \cdot GSW.Eval(f, C_1, C_2).
\end{aligned}$$

Case 2. when the circuit computing f is only a multiplication gate, i.e. $f = x_1 \cdot x_2$, for any GSW ciphertexts $C_1 = \begin{bmatrix}B_1\\s^TB_1 + e_1^T\end{bmatrix} + \mu_1G, C_2 = \begin{bmatrix}B_2\\s^TB_2 + e_2^T\end{bmatrix} + \mu_2G$, we have

$$\begin{aligned}
&GSW.Eval(kf, C_1, C_2)\\
&= (kC_1) \cdot G^{-1}(C_2)\\
&= (\begin{bmatrix}kB_1\\ks^TB_1 + ke_1^T\end{bmatrix} + k\mu_1G) \cdot G^{-1}(C_2)\\
&= \begin{bmatrix}kB_1G^{-1}(C_2)\\ks^TB_1G^{-1}(C_2) + ke_1^TG^{-1}(C_2)\end{bmatrix} + k\mu_1C_2\\
&= \begin{bmatrix}kB_1G^{-1}(C_2) + k\mu_1B_2\\s^T(kB_1G^{-1}(C_2) + k\mu_1B_2) + ke_1^TG^{-1}(C_2) + k\mu_1e_2^T\end{bmatrix}\\
&\quad + k\mu_1\mu_2G\\
&= k \cdot GSW.Eval(f, C_1, C_2).
\end{aligned}$$

In general, any depth d circuit can be implemented by some addition and multiplication gates, hence this homomorphic property is naturally conserved in the case of general circuits.

Proof of Claim 4.2: Note that matrix B_f is computed from $\hat{f}$ through running the matrices evolution algorithm $EvalF(B_1, \cdots, B_L, \hat{f})$. Hence, to prove the relationship in Claim 4.2 means to prove that the matrices evolution algorithm $EvalF()$ has the following homomorphic property:

$$EvalF(B_1, \cdots, B_L, k\cdot\hat{f}) = k\cdot EvalF(B_1, \cdots, B_L, \hat{f}).$$

Case 1. when the circuit computing $\hat{f}$ is only an addition gate, i.e. $f = x_1 + \cdots + x_L$, for any GSW ciphertexts $B_1, \cdots, B_L$, we have

$$\begin{aligned} & EvalF(B_1, \cdots, B_L, k\cdot\hat{f}) \\ &= [kE, \cdots, kE]^T \\ &= k[E, \cdots, E]^T \\ &= k\cdot EvalF(B_1, \cdots, B_L, \hat{f}). \end{aligned}$$

Case 2. when the circuit computing $\hat{f}$ is only an multiplication gate, i.e. $f = x_1 \times \cdots \times x_L$, for any GSW ciphertexts $B_1, \cdots, B_L$, we have

$$\begin{aligned} & EvalF(B_1, \cdots, B_L, k\cdot\hat{f}) \\ &= [O, \cdots, O, kG^{-1}(\cdots G^{-1}(-B_2G^{-1}(-B_1)))]^T \\ &= k\cdot[O, \cdots, O, G^{-1}(\cdots G^{-1}(-B_2G^{-1}(-B_1)))]^T \\ &= k\cdot EvalF(B_1, \cdots, B_L, \hat{f}). \end{aligned}$$

In general, any depth d circuit can be implemented by some addition and multiplication gates, hence this homomorphic property is naturally conserved in the case of general circuits.

Proof of Lemma 3: Similar to the proof of Lemma 2, here we omit it.

5 Attack #II

In this section,we provide another attack to demonstrate that the predicate encryption scheme reviewed in section 3 is insecure against an adversary that requests 1-keys. This attack exploits two types of linear error growth in the construction of the scheme in [11]. One type of this error growth is from the ciphertexts homomorphic evolution algorithm in [8]; the other one results from the GSW evaluation algorithm in [5]. Concretely, we first recall the correctness of the scheme in [11] as follows:

$$\begin{aligned} & c_{out} - \left[c_{in}^T | c_{\hat{f}}^T\right]\cdot sk_f \\ &= c_{out} - \left[c_{in}^T | \left[c_1^T|\cdots|c_L^T\right]\cdot H_{\hat{f},\Psi} + \underline{\Psi}_f\right]\cdot sk_f \\ &= c_{out} - \left[c_{in}^T | [s^T[B_1 - \psi_1\overline{G}] + e_1^T | \cdots | s^T[B_L - \psi_l\overline{G}] \right. \\ &\quad \left. + e_L^T]\cdot H_{\hat{f},\Psi} + \underline{\Psi}_f\right]\cdot sk_f \\ &= c_{out} - \Big[c_{in}^T | s^T \underbrace{[B_1 - \psi_1\overline{G}|\cdots|B_L - \psi_l\overline{G}]\cdot H_{\hat{f},\Psi}}_{B_f - \overline{\Psi}_f} \\ &\quad + \underbrace{[e_1^T|\cdots|e_L^T]\cdot H_{\hat{f},\Psi}}_{e_{ABE}} + \underline{\Psi}_f\Big]\cdot sk_f \\ &= c_{out} - \left[s^TB + e_0 | s^T[B_f - \overline{\Psi}_f] + \underline{\Psi}_f + e_{ABE}\right]\cdot sk_f \\ &= c_{out} - s^T[B|B_f]\cdot sk_f - [O|(-s^T, 1)\Psi_f]\cdot sk_f \\ &\quad - [e_0|e_{ABE}]\cdot sk_f \\ &= s^T\left[p - [B|B_f]\cdot sk_f\right] - [O|(-s^T, 1)\Psi_f]\cdot sk_f + e' \\ &\quad - [e_0|e_{ABE}]\cdot sk_f + \lfloor\frac{q}{2}\rfloor\cdot\mu \\ &= s^T\left[p - [B|B_f]\cdot sk_f\right] - [O|f(x)\cdot(-s^T, 1)G]\cdot sk_f + e' \\ &\quad - [e_0|e_{GSW} + e_{ABE}]\cdot sk_f + \lfloor\frac{q}{2}\rfloor\cdot\mu \\ &= e' - [e_0|e_{GSW} + e_{ABE}]\cdot sk_f + \lfloor\frac{q}{2}\rfloor\cdot\mu, \end{aligned}$$

where the fourth equality is because of the key relation, and the final equality is because the queries requsted by adversary is 1-keys.

Note that the key sk_f is known by adversary, and by the cipthertext evolution algorithm $EvalFX$, we have

$$e_{ABE} = [e_1^T|\cdots|e_L^T]\cdot H_{\hat{f},\Psi}$$

where $H_{\hat{f},\Psi}$ can also be computed by adversary from f and Ψ through the cipthertext evolution algorithm $EvalFX$. Thus, the term e_{ABE} is linear in these original errors $e_1^T, \cdots, e_L^T$ with public coefficients.

On the other hand, by the construction of the GSW homomorphic evaluation algorithm, the term e_{GSW} is also publicly linear in the errors $e^TR_1, \cdots, e^TR_l$ which are used in the construction of the GSW fresh cipthertext Ψ.

According to the analysis above, it is not difficult to see that a single 1-key (even if it corresponds to a non-linear function) yields a system of m linear equations in the $(l+L+2)m$ variables $e', e_0, e_1, \cdots, e_L, \hat{e}_1, \cdots, \hat{e}_l$ where $\hat{e}_1, \cdots, \hat{e}_l$ denots $R_1^Te, \cdots, R_l^Te$ respectively. By requesting $l + L + 2$ keys totally, the adversary can completely recover the above error terms, which in turn lead to recovery of the main secret s, which then permit to recover all the private attributes completely.

6 Conclusion and Open Problems

In this paper, we propose two practical attacks that demonstrate the predicate encryption scheme proposed by Brakerski *etc.* is insecure under the full attribute-hiding secrity model. The first type of attack mainly exploits two homomorphic properties in construction of the scheme; the other one, however, takes advantage of two types of linear properties in the process of error growth in the construction. This leaves open two possibilities:

1) Optimize the construction of the scheme to resist these two types of attack;

2) Look for new construction of the predicate scheme from lattice based assumptions to bypass those weak properties.

Acknowledgments

This work was supported by the Doctoral Fund of University of Jinan (Granted No. XBS1455), and National Science Foundation of Shandong Province (No. ZR2018LF006).

References

[1] H. Abdalla, X. Hu, A. Wahaballa, A. Abdalla, M. Ramadan, and Z. Qin, "Integrating the functional encryption and proxy re-cryptography to secure drm scheme," *International Journal of Network Security*, vol. 19, no. 1, pp. 27–38, 2017.

[2] S. Agrawal, "Stronger security for reusable garbled circuits, general definitions and attacks," in *Proceedings of the 37th International Cryptology Conference (CRYPTO'17)*, pp. 3–35, Santa Barbara, USA, August 2017.

[3] S. Agrawal, D. Boneh, and X. Boyen, "Lattice basis delegation in xed dimension and shorterciphertext hierarchical IBE," in *Proceedings of the 30th International Cryptology Conference (CRYPTO'10)*, pp. 98–115, Santa Barbara, USA, August 2010.

[4] S. Agrawal, D. M. Freeman, and V. Vaikuntanathan, "Functional encryption for inner product predicates from learning with errors," in *Proceedings of 17th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'11)*, pp. 21–40, Seoul, Korea, December 2011.

[5] J. Alperin-Sheriff and C. Peikert, "Faster bootstrapping with polynomial error," in *Proceedings of the 30th International Cryptology Conference (CRYPTO'14)*, LNCS 8616m pp. 297–314, Springer, Aug. 2014.

[6] D. Apon, X. Fan, and F. H. Liu, "Compact identity based encryption from LWE," in *Cryptology ePrint Archive*, Report 2016/125, 2016. (`http://eprint.iacr.org/2016/125`)

[7] M. Bayat, M. Aref, "An attribute based key agreement protocol resilient to KCI attack," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 10–20, 2015.

[8] D. Boneh, C. Gentry, S. Gorbunov, S. Halevi, V. Nikolaenko, G. Segev, V. Vaikuntanathan, and D. Vinayagamurthy, "Fully key-homomorphic encryption, arithmetic circuit abe and compact garbled circuits," in *Proceedings of 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques(EUROCRYPT'14)*, pp. 533–556, Copenhagen, Denmark, May 2014.

[9] D. Boneh, A. Sahai, and B. Waters, "Functional encryption: Definitions and challenges," in *Proceedings of The 8th Theory of Cryptography Conference (TCC'11)*, pp. 253–273, Rhode Island, USA, March 2011.

[10] D. Boneh and B. Waters, "Functional encryption: Definitions and challenges," in *Proceedings of The 4th Theory of Cryptography Conference (TCC 2007)*, pp. 535–554, Amsterdam, The Netherlands, February 2007.

[11] Z. Brakerski, R. Tsabary, V. Vaikuntanathan, and H. Wee, "Private constrained prfs (and more) from lwe," in *Proceedings of The 15th Theory of Cryptography Conference (TCC'17)*, pp. 264–302, Baltimore, USA, November 2017.

[12] Z. Brakerski and V. Vaikuntanathan, "Latticebased fhe as secure as PKE," in *Proceedings of The 3rd International Conference on Information Technology and Computer Science (ITCS'14)*, pp. 1–12, Saipan, USA, July 2014.

[13] Z. Cao, L. Liu, Z. Guo, "Ruminations on attribute-based encryption," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 9–19, 2018.

[14] Z. Cao, L. Liu, Y. Li, "Ruminations on fully homomorphic encryption in client-server computing scenario," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 32–39, 2018.

[15] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments", *International Journal of Network Security*, vol. 16, no. 1, pp. 1-13, 2014.

[16] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters, "Candidate indistinguishability obfuscation and functional encryption for all circuits," in *Proceedings of The 54th Annual Symposium on Foundations of Computer Science (FOCS'13)*, pp. 40–49, Berkeley, California, October 2013.

[17] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of The 41st Annual ACM Symposium on Theory of Computing (STOC'09)*, pp. 169–178, Bethesda, MD, USA, May/June 2009.

[18] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in *Proceedings of the 33th International Cryptology Conference (CRYPTO'13)*, pp. 75–92, Santa Barbara, USA, August 2013.

[19] S. Gorbunov, V. Vaikuntanathan, and H. Wee, "Attribute-based encryption for circuits," in *Proceedings of The 45st Annual ACM Symposium on Theory of Computing (STOC'13)*, pp. 545–554, Palo Alto, CA, USA, June 2013.

[20] S. Gorbunov, V. Vaikuntanathan, and H. Wee, "Predicate encryption for circuits from lwe," in *Proceedings of the 35th International Cryptology Conference (CRYPTO'15)*, pp. 503–523, Santa Barbara, USA, August 2015.

[21] P. Hu and H. Gao, "A key-policy attribute-based encryption scheme for general circuit from bilinear maps," *International Journal of Network Security*, vol. 19, no. 5, pp. 704–710, 2017.

[22] J. Katz, A. Sahai, and B. Waters, "Predicate encryption supporting disjunctions, polynomial equations, and inner products," in *Proceedings of 27rd Annual International Conference on the Theory and Applications of Cryptographic Techniques(EUROCRYPT'08)*, pp. 146–162, Istanbul, Turkey, April 2008.

[23] D. Micciancio and C. Peikert, "Trapdoors for lattices: Simpler, tighter, faster, smaller," in *Proceedings of 31rd Annual International Conference on the Theory and Applications of Cryptographic Techniques(EUROCRYPT'12)*, pp. 700–718, Cambridge, United Kingdom, April 2012.

[24] Y. Wang, "Lattice ciphertext policy attribute-based encryption in the standard model," *International Journal of Network Security*, vol. 16, no. 6, pp. 444–451, 2014.

[25] L. Zhang and H. Yin, "Recipient anonymous ciphertext-policy attribute-based broadcast encryption," *International Journal of Network Security*, vol. 20, no. 1, pp. 168–176, 2018.

Biography

Chengbo Xu received the B.S. degree in Mathematics from the Liaocheng University, China, in 2002, the M.S. degree in Cryptology from the Hubei University, China, in 2005, and the Ph.D. degree in Computer Science from the Beijing University of Post and Telecommunication, China, in 2014, respectively. Currently, He is a Lecture in the School of Mathimatical Sciences at University of Jinan. His research interests include information security and cryptology. Dr. Xu may be reached at cbqysy@163.com.